Westfälische Wilhelms-Universität Münster

Applied Mathematics Münster

Comparison of Methods for Parametric Model Order Reduction of Instationary Problems (II)

Ulrike Baur (baur@mpi-magdeburg.mpg.de)
Peter Benner (benner@mpi-magdeburg.mpg.de)
Bernard Haasdonk (haasdonk@mathematik.uni-stuttgart.de)
Christian Himpe (christian.himpe@uni-muenster.de)
Immanuel Martini (immanuel.martini@mathematik.uni-stuttgart.de)
Mario Ohlberger (mario.ohlberger@uni-muenster.de)

EU-MORNET Kick-Off Meeting
18.-19.09.2014

Benchmark Framework

Overview:

- Benchmarks
- Methods
- Tasks
- Taskmaster

About:

- Written in Matlab
- Version 1.0 will be Open-Source
- Extendable with further Benchmarks and Methods
- Reproducible Results

Benchmarks I (About)

- State-space models $\begin{cases} E\dot{x} = (A + A_p)x + Bu \\ y = Cx + Du \end{cases}$
- BYO Integrator
- Listed in the MORwiki (`http://modelreduction.org`)
  - Synthetic ($\dim(x) = 1000$)
  - Microthruster ($\dim(x) = 4257$)
  - Anemometer ($\dim(x) = 29008$)

Benchmarks II (Interface)

- `generate`-method
  - Assembles model
  - Opaque to methods
  - Returns state-space matrices $\{A, A_p, B, C, D, E\}$
- `integrate`-method
  - Computes trajectory for given x_0, u, p
  - Transparent to methods
  - Returns fixed-step-width time series

Benchmarks III (Limitations)

Currently:

- Single-Input-Single Output
- Linear-Time-Invariant
- Single Parameter (affinely dependent)
- Parametrized System Matrix A_p

Future:

- Multiple-Input-Multiple-Output
- Linear-Time-Varying, DDEs, DAEs
- Many Parameters
- Arbitrary Parametrization $\{A_p, B_p, C_p, D_p, E_p\}$
- Source Term F (F_p)

Methods I (About)

- Model order reduction code
- Should not depend on (or be) propriety code (if possible)
- (Should be) listed in the MORwiki

Methods II (Interface)

- No interface requirements (yet)
- Must be able to share data with Matlab

Methods III (On Board)

Currently:

- POD
- POD-Greedy
- Matrix Interpolation
- Transfer Function Interpolation
- Piecewise Tangential $\mathcal{H}_2$ Interpolation
- Multi Parameter Moment Matching
- Empirical (Cross) Gramians

Future:

- (Variant of) Balanced Truncation
- Loewner Framework
- Interpolation on Grassmann Manifold
- Reduced Basis
- ...

Tasks I (About)

- Function Container
- Evaluates a set Benchmark,
- with a set pMOR Method

Tasks II (Interface)

```
{Ar, Apr, Br, Cr, Er, X0r, V, W, Y, Z } =
task_METHOD_BENCHMARK(A,Ap,B,C,D,E,X0,U,P,h,T,r)
```

- Gets full order model (`A,Ap,B,C,D,E,X0`)
- Input (`U`)
- Parameter range (`P`)
- Target reduced order (`r`)
- Returns reduced order model (`Ar,Apr,Br,Cr,Er,X0r`)
- Returns reducing projections (`V,W`)
- Can return extra information (`Y,Z`)
  for use during the online phase

Tasks III (Limitations)

Currently:

- (Benchmark) Input is transparent
- Only target reduced order
- Own or benchmark's integrator

Future:

- Input will be opaque
- Target offline time
- Force own or benchmark's integrator (?)
- ...

Taskmaster I (About)

1. Evaluate Full-Order Model
2. Execute Offline Phase for all Tasks
3. Execute Online Phase for all Tasks
   - Reduced Order Simulations
   - Error Computation
4. Store Results

Taskmaster II (Benchmark Ranges)

Currently:

- Time Interval
- Frequency Range
- Parameter Range
- Impulse Input
- Arbitrary Initial Values
- Fixed Reduced Dimension

Future:

- Arbitrary (RMS) Input
- Uncertainties
- Varying Reduced Dimension
- ...

Taskmaster III (ROM Quality)

Currently:

- Offline Time
- Online Time
- $\mathcal{L}_2$-error (in states)
- $\mathcal{L}_2$-error (in outputs)
- $\mathcal{L}_\infty$-error (in outputs)
- $\mathcal{H}_2$-error (in outputs)
- $\mathcal{H}_\infty$-error (in outputs)
- Scaled $\mathcal{H}_\infty$-error (in outputs)
- ROM Stability Test

Future:

- Machine independent efficiency indicators
- $\mathcal{L}_1$-error (in outputs)
- $\max_\theta \|y(\theta) - y_r(\theta)\|_?$
- ...

Outlook

Content:

- More Benchmarks (MORwiki,Oberwolfach,NICONET,Industry,...)
- More Methods
- Nonlinear Systems (ie Gradient Systems)

Architecture:

- Toolbox Independence (currently: Control System Toolbox)
- Performance (especially $\mathcal{H}_2$ error)
- Parallelization (per Benchmark)

Usage:

- Graphical User Interface
- Results Database in MORwiki (possibly versioned)
- Octave Compatibility (full open-source stack)

Long Term Goals

- Cluster & Local
- Choose & Compare different Benchmarks and Methods
- Extract Classifications from the MORwiki
- Evaluate Errors & Durations
- Ensure Performance for my new Versions
- All Open-Source & Open-Data
- Task is sample implementation
- Fullfill Science-Code-Manifesto[1]

[1] http://sciencecodemanifesto.org

tl;dl

We have:

- Basic Benchmark Framework,
- with Modular Architecture

We need:

- Tuned Methods
- Benchmarks with Integrators

What do you want / need / suggest ?

Thanks!