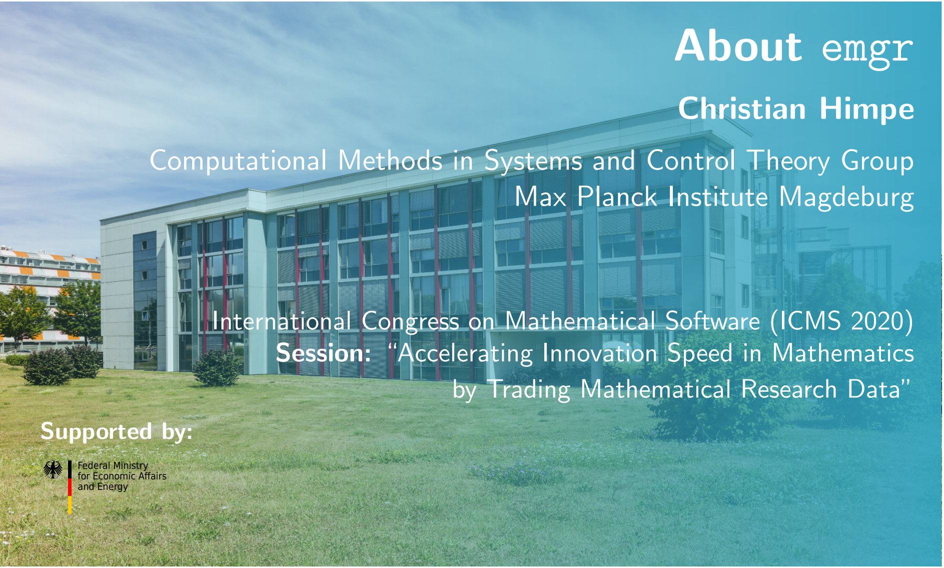
MAX PLANCK INSTITUTE
FOR DYNAMICS OF COMPLEX
TECHNICAL SYSTEMS
MAGDEBURG

CSC COMPUTATIONAL METHODS IN
SYSTEMS AND CONTROL THEORY



About emgr

Christian Himpe

Computational Methods in Systems and Control Theory Group
Max Planck Institute Magdeburg

International Congress on Mathematical Software (ICMS 2020)
Session: "Accelerating Innovation Speed in Mathematics by Trading Mathematical Research Data"

Supported by:

Federal Ministry
for Economic Affairs
and Energy

Outline

1. What is `emgr` (**EM**pirical **GR**amian Framework)?
2. Mathematical software as research data!

Locating emgr

- Applied mathematics

Locating emgr

- Applied mathematics
  - **Mathematical system theory**
    - Modern system theory by Kalman (includes control theory)
    - Study of dynamical input-output systems

Locating emgr

- Applied mathematics
  - **Mathematical system theory**
    - Modern system theory by Kalman (includes control theory)
    - Study of dynamical input-output systems
  - **Numerical mathematics**
    - Numerical linear algebra (matrix decompositions etc.)
    - Numerical analysis (differential equations etc.)

Locating emgr

- Applied mathematics
  - **Mathematical system theory**
    - Modern system theory by KALMAN (includes control theory)
    - Study of dynamical input-output systems
  - **Numerical mathematics**
    - Numerical linear algebra (matrix decompositions etc.)
    - Numerical analysis (differential equations etc.)
  - **Scientific computing**
    - Efficient implementations
    - Parallelization

Locating emgr

- Applied mathematics
  - **Mathematical system theory**
    - Modern system theory by Kalman (includes control theory)
    - Study of dynamical input-output systems
  - **Numerical mathematics**
    - Numerical linear algebra (matrix decompositions etc.)
    - Numerical analysis (differential equations etc.)
  - **Scientific computing**
    - Efficient implementations
    - Parallelization
  - **Computational science and engineering**
    - Modeling and discretization of engineering applications
    - Simulate, estimate, predict real-life behavior

Dynamical Input-Output System

General input-output system:

$$
\begin{aligned}
0 &= f(t, x(t), u(t), \theta, D^\alpha x(t)) \\
y(t) &= g(t, x(t), u(t), \theta)
\end{aligned}
$$

Dynamical Input-Output System

General input-output system:

$$\begin{aligned} 0 &= f(t, x(t), u(t), \theta, D^\alpha x(t)) \\ y(t) &= g(t, x(t), u(t), \theta) \end{aligned}$$

Example: **Linear time-invariant system**

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned}$$

Dynamical Input-Output System

General input-output system:

$$0 = f(t, x(t), u(t), \theta, D^\alpha x(t))$$
$$y(t) = g(t, x(t), u(t), \theta)$$

Example: **Linear time-invariant system**

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

Example: **Gas network model**

$$\begin{pmatrix} E_p & 0 \\ 0 & I \end{pmatrix}\begin{pmatrix} \dot{p} \\ \dot{q} \end{pmatrix} = \begin{pmatrix} 0 & A_{pq} \\ A_{qp} & 0 \end{pmatrix}\begin{pmatrix} p \\ q \end{pmatrix} + \begin{pmatrix} 0 & B_d \\ B_s & 0 \end{pmatrix}\begin{pmatrix} s_p \\ d_q \end{pmatrix} + \begin{pmatrix} 0 \\ f_q(p, q, s_p) \end{pmatrix}$$
$$\begin{pmatrix} s_q \\ d_p \end{pmatrix} = \begin{pmatrix} 0 & C_q \\ C_p & 0 \end{pmatrix}\begin{pmatrix} p \\ q \end{pmatrix}$$

System Gramians

Core input-output system properties:

- Reachability (can the state be driven to any state by finite input?)
- Observability (can the state be decoded from outputs in finite time?)

System Gramians

Core input-output system properties:

- Reachability (can the state be driven to any state by finite input?)
- Observability (can the state be decoded from outputs in finite time?)

Linear time-invariant system:

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

System Gramians

Core input-output system properties:

- Reachability (can the state be driven to any state by finite input?)
- Observability (can the state be decoded from outputs in finite time?)

Linear time-invariant system:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned}$$

Example: **Reachability System Gramian**

$$W_R := \int_0^\infty \mathrm{e}^{At} BB^\intercal \mathrm{e}^{A^\intercal t} \,\mathrm{d}t$$

System Gramians

Core input-output system properties:

- Reachability (can the state be driven to any state by finite input?)
- Observability (can the state be decoded from outputs in finite time?)

Linear time-invariant system:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t)\\ y(t) &= Cx(t)\end{aligned}$$

Example: **Reachability System Gramian**

$$W_R := \int_0^\infty \mathrm{e}^{At}\, BB^\intercal\, \mathrm{e}^{A^\intercal t}\, \mathrm{d}t \quad \Leftrightarrow \quad AW_R + W_R A^\intercal = -BB^\intercal$$

Nonlinear system:

$$\dot{x}(t) = f(x(t), u(t))$$
$$y(t) = g(x(t), u(t))$$

Nonlinear system:

$$\dot{x}(t) = f(x(t), u(t))$$
$$y(t) = g(x(t), u(t))$$

- No A, B, C!

Empirical Gramians

Nonlinear system:

$$\dot{x}(t) = f(x(t), u(t))$$
$$y(t) = g(x(t), u(t))$$

- No A, B, C!

- However: $\dot{x}(t) = Ax(t) + b\delta(t) \Rightarrow x(t) = \mathrm{e}^{At}\, b$:

$$W_R := \int_0^\infty \mathrm{e}^{At}\, bb^\intercal\, \mathrm{e}^{A^\intercal t}\, \mathrm{d}t = \int_0^\infty x(t)x(t)^\intercal\, \mathrm{d}t$$

Nonlinear system:

$$\dot{x}(t) = f(x(t), u(t))$$
$$y(t) = g(x(t), u(t))$$

- No A, B, C!

- However: $\dot{x}(t) = Ax(t) + b\delta(t) \Rightarrow x(t) = \mathrm{e}^{At}\, b$:

$$W_R := \int_0^\infty \mathrm{e}^{At}\, bb^\intercal\, \mathrm{e}^{A^\intercal t}\, \mathrm{d}t = \int_0^\infty x(t)x(t)^\intercal\, \mathrm{d}t$$

- Use state trajectory data from (scaled) impulse responses.
- Impulse responses can also be computed for nonlinear systems!
- Observability can similarly be described by trajectory data.

Application:

Scoping `emgr`

Application:

- Model reduction (aka Model Order Reduction)
  - Parametric Model Order Reduction (pMOR)
  - Nonlinear Model Order Reduction (nMOR)
  - Combined state and parameter reduction (comred)

Scoping emgr

Application:

- Model reduction (aka Model Order Reduction)
  - Parametric Model Order Reduction (pMOR)
  - Nonlinear Model Order Reduction (nMOR)
  - Combined state and parameter reduction (comred)
- Decentralized control

Scoping emgr

Application:

- Model reduction (aka Model Order Reduction)
  - Parametric Model Order Reduction (pMOR)
  - Nonlinear Model Order Reduction (nMOR)
  - Combined state and parameter reduction (comred)
- Decentralized control
- Sensitivity analysis

Scoping emgr

Application:

- Model reduction (aka Model Order Reduction)
  - Parametric Model Order Reduction (pMOR)
  - Nonlinear Model Order Reduction (nMOR)
  - Combined state and parameter reduction (comred)
- Decentralized control
- Sensitivity analysis
- Parameter identification / structural identifiability

Scoping emgr

Application:

- Model reduction (aka Model Order Reduction)
  - Parametric Model Order Reduction (pMOR)
  - Nonlinear Model Order Reduction (nMOR)
  - Combined state and parameter reduction (comred)
- Decentralized control
- Sensitivity analysis
- Parameter identification / structural identifiability
- Optimal actuator placement / optimal sensor placement

Scoping emgr

Application:

- Model reduction (aka Model Order Reduction)
  - Parametric Model Order Reduction (pMOR)
  - Nonlinear Model Order Reduction (nMOR)
  - Combined state and parameter reduction (comred)
- Decentralized control
- Sensitivity analysis
- Parameter identification / structural identifiability
- Optimal actuator placement / optimal sensor placement
- Uncertainty quantification

Scoping emgr

Application:

- Model reduction (aka Model Order Reduction)
  - Parametric Model Order Reduction (pMOR)
  - Nonlinear Model Order Reduction (nMOR)
  - Combined state and parameter reduction (comred)
- Decentralized control
- Sensitivity analysis
- Parameter identification / structural identifiability
- Optimal actuator placement / optimal sensor placement
- Uncertainty quantification
- Nonlinearity quantification

Example: **Linear time-invariant system**

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

Example: **Linear time-invariant system**

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

System dimensions:

- $x : \mathbb{R}_{>0} \to \mathbb{R}^N$
- $u : \mathbb{R}_{>0} \to \mathbb{R}^M$
- $y : \mathbb{R}_{>0} \to \mathbb{R}^Q$
- $A \in \mathbb{R}^{N \times N}$
- $B \in \mathbb{R}^{N \times M}$
- $C \in \mathbb{R}^{Q \times N}$

Example: **Linear time-invariant system**

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

System dimensions:

- $x : \mathbb{R}_{>0} \to \mathbb{R}^N$
- $u : \mathbb{R}_{>0} \to \mathbb{R}^M$
- $y : \mathbb{R}_{>0} \to \mathbb{R}^Q$
- $A \in \mathbb{R}^{N \times N}$
- $B \in \mathbb{R}^{N \times M}$
- $C \in \mathbb{R}^{Q \times N}$
- $N = \dim(x(t)) \gg 1$
- $M = \dim(u(t)) \ll \dim(x(t)) = N$
- $Q = \dim(u(t)) \ll \dim(x(t)) = N$

Model Reduction

Example: **Linear time-invariant system**

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

System dimensions:

- $x : \mathbb{R}_{>0} \to \mathbb{R}^N$
- $u : \mathbb{R}_{>0} \to \mathbb{R}^M$
- $y : \mathbb{R}_{>0} \to \mathbb{R}^Q$
- $A \in \mathbb{R}^{N \times N}$
- $B \in \mathbb{R}^{N \times M}$
- $C \in \mathbb{R}^{Q \times N}$
- $N = \dim(x(t)) \gg 1$
- $M = \dim(u(t)) \ll \dim(x(t)) = N$
- $Q = \dim(u(t)) \ll \dim(x(t)) = N$

Projection-based model order reduction:

- $V : \mathbb{R}^N \to \mathbb{R}^n$
- $x_r := Vx$
- $n \ll N$

Example: **Linear time-invariant system**

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

System dimensions:

- $x : \mathbb{R}_{>0} \to \mathbb{R}^N$
- $u : \mathbb{R}_{>0} \to \mathbb{R}^M$
- $y : \mathbb{R}_{>0} \to \mathbb{R}^Q$
- $A \in \mathbb{R}^{N \times N}$
- $B \in \mathbb{R}^{N \times M}$
- $C \in \mathbb{R}^{Q \times N}$
- $N = \dim(x(t)) \gg 1$
- $M = \dim(u(t)) \ll \dim(x(t)) = N$
- $Q = \dim(u(t)) \ll \dim(x(t)) = N$

Projection-based model order reduction:

- $V : \mathbb{R}^N \to \mathbb{R}^n$
- $U : \mathbb{R}^n \to \mathbb{R}^N$
- $x_r := Vx$
- $x \approx Ux_r$
- $n \ll N$
- $\|y - \tilde{y}\|_? \ll 1$

Example: **Linear time-invariant system**

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

System dimensions:

- $x : \mathbb{R}_{>0} \to \mathbb{R}^N$
- $u : \mathbb{R}_{>0} \to \mathbb{R}^M$
- $y : \mathbb{R}_{>0} \to \mathbb{R}^Q$
- $A \in \mathbb{R}^{N \times N}$
- $B \in \mathbb{R}^{N \times M}$
- $C \in \mathbb{R}^{Q \times N}$
- $N = \dim(x(t)) \gg 1$
- $M = \dim(u(t)) \ll \dim(x(t)) = N$
- $Q = \dim(u(t)) \ll \dim(x(t)) = N$

Projection-based model order reduction:

- $V : \mathbb{R}^N \to \mathbb{R}^n$
- $U : \mathbb{R}^n \to \mathbb{R}^N$
- $x_r := Vx$
- $x \approx Ux_r$
- $n \ll N$
- $\|y - \tilde{y}\|_? \ll 1$

Reduced order model:

$$\dot{x}_r(t) = (V(A(Ux_r(t)))) + (V(Bu(t)))$$
$$\tilde{y}(t) = \quad (C(Ux_r(t)))$$

Model Reduction

Example: **Linear time-invariant system**

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

System dimensions:

- $x : \mathbb{R}_{>0} \to \mathbb{R}^N$
- $u : \mathbb{R}_{>0} \to \mathbb{R}^M$
- $y : \mathbb{R}_{>0} \to \mathbb{R}^Q$
- $A \in \mathbb{R}^{N \times N}$
- $B \in \mathbb{R}^{N \times M}$
- $C \in \mathbb{R}^{Q \times N}$
- $N = \dim(x(t)) \gg 1$
- $M = \dim(u(t)) \ll \dim(x(t)) = N$
- $Q = \dim(u(t)) \ll \dim(x(t)) = N$

Projection-based model order reduction:

- $V : \mathbb{R}^N \to \mathbb{R}^n$
- $U : \mathbb{R}^n \to \mathbb{R}^N$
- $x_r := Vx$
- $x \approx Ux_r$
- $n \ll N$
- $\|y - \tilde{y}\|_? \ll 1$

Reduced order model:

$$\dot{x}_r(t) = (VAU)x_r(t) + (VB)u(t)$$
$$\tilde{y}(t) = \quad (CU)x_r(t)$$

Model Reduction

Example: **Linear time-invariant system**

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

System dimensions:

- $x : \mathbb{R}_{>0} \to \mathbb{R}^N$
- $u : \mathbb{R}_{>0} \to \mathbb{R}^M$
- $y : \mathbb{R}_{>0} \to \mathbb{R}^Q$
- $A \in \mathbb{R}^{N \times N}$
- $B \in \mathbb{R}^{N \times M}$
- $C \in \mathbb{R}^{Q \times N}$
- $N = \dim(x(t)) \gg 1$
- $M = \dim(u(t)) \ll \dim(x(t)) = N$
- $Q = \dim(u(t)) \ll \dim(x(t)) = N$

Projection-based model order reduction:

- $V : \mathbb{R}^N \to \mathbb{R}^n$
- $U : \mathbb{R}^n \to \mathbb{R}^N$
- $x_r := Vx$
- $x \approx Ux_r$
- $n \ll N$
- $\|y - \tilde{y}\|_? \ll 1$

Reduced order model:

$$\dot{x}_r(t) = A_r x_r(t) + B_r u(t)$$
$$\tilde{y}(t) = C_r x_r(t)$$

Empirical Gramians:

- Empirical Reachability Gramian
- Empirical Observability Gramian
- Empirical Linear Cross Gramian
- Empirical Cross Gramian
- Empirical Sensitivity Gramian
- Empirical Identifiability Gramian
- Empirical Joint Gramian

Initial release of emgr (0.9) at: MoRePaS 2 (2012)

Features:

- Open-source **Octave** and **MATLAB** toolbox, with **Python** variant.
- Interfaces for: Solver, inner product kernels & low-rank computation.
- Configurable, vectorized and parallelizable.

More info: https://gramian.de

C.H. emgr – **The Empirical Gramian Framework**. Algorithms 11(7): 91, 2018. doi:c9hj

Through the lens of `emgr`

Classes of Software

Classes of Software

- **Single-use software**
  - Computational proofs
  - Numerical experiments
  - Visualization generators

Classes of Software

- **Single-use software**
  - Computational proofs
  - Numerical experiments
  - Visualization generators
- **Reusable software**
  - Few developers
  - Few users (mostly developers)
  - Continuous development

Classes of Software

- **Single-use software**
  - Computational proofs
  - Numerical experiments
  - Visualization generators
- **Reusable software**
  - Few developers
  - Few users (mostly developers)
  - Continuous development
- **Sustainable software**
  - Many developers
  - Many users (including non-developers)
  - Sustained development

Classes of Software

- **Single-use software**
  - Computational proofs
  - Numerical experiments
  - Visualization generators
- **Reusable software**
  - Few developers
  - Few users (mostly developers)
  - Continuous development
- **Sustainable software**
  - Many developers
  - Many users (including non-developers)
  - Sustained development

Long-Time Archival

Safety in numbers:

- Project website / personal website / institute website

Long-Time Archival

Safety in numbers:

- Project website / personal website / institute website
- *github* project (`https://github.com`)

Long-Time Archival

Safety in numbers:

- Project website / personal website / institute website
- *github* project (`https://github.com`)
- *Software Heritage* archive (`https://softwareheritage.org`)

Long-Time Archival

Safety in numbers:

- Project website / personal website / institute website
- *github* project (`https://github.com`)
- *Software Heritage* archive (`https://softwareheritage.org`)
- *Zenodo* archive (`https://zenodo.org`)

Long-Time Archival

Safety in numbers:

- Project website / personal website / institute website
- *github* project (`https://github.com`)
- *Software Heritage* archive (`https://softwareheritage.org`)
- *Zenodo* archive (`https://zenodo.org`)
- *Internet Archive* (`https://archive.org`)

Persistent Identifiers

Pinpointing state:

- Website URL (Uniform Resource Locator)

Persistent Identifiers

Pinpointing state:

- Website URL (Uniform Resource Locator)

- *github* `git tag` / hash

Persistent Identifiers

Pinpointing state:

- Website URL (Uniform Resource Locator)
- *github* `git tag` / hash
- *Software Heritage* permalink

Persistent Identifiers

Pinpointing state:

- Website URL (Uniform Resource Locator)
- *github* `git tag` / hash
- *Software Heritage* permalink
- *Zenodo* DOI (Digital Object Identifier)

Persistent Identifiers

Pinpointing state:

- Website URL (Uniform Resource Locator)
- *github* `git tag` / hash
- *Software Heritage* permalink
- *Zenodo* DOI (Digital Object Identifier)
- Version (semantic version vs. calendar version)

Centered around ...

Digital Libraries

Centered around ...

- Community (i.e. *MORwiki*, *MoRePaS*)

Centered around ...

- Community (i.e. *MORwiki*, *MoRePaS*)
- Subject (i.e *DSweb*, *Zenodo* communities)

Digital Libraries

Centered around ...

- Community (i.e. *MORwiki*, *MoRePaS*)
- Subject (i.e *DSweb*, *Zenodo* communities)
- Mathematics (*ORMS*, *swMATH*)

Digital Libraries

Centered around ...

- Community (i.e. *MORwiki*, *MoRePaS*)
- Subject (i.e *DSweb*, *Zenodo* communities)
- Mathematics (*ORMS*, *swMATH*)
- Science (*Researchgate*, *archives-ouvertes*)

Digital Libraries

Centered around ...

- Community (i.e. *MORwiki*, *MoRePaS*)
- Subject (i.e *DSweb*, *Zenodo* communities)
- Mathematics (*ORMS*, *swMATH*)
- Science (*Researchgate*, *archives-ouvertes*)
- Language (*MATLAB File Exchange*)

Semantic Enrichment

ALL CAPS:

ALL CAPS:

- `README` (plain text minimal documentation)

ALL CAPS:

- `README` (plain text minimal documentation)
- `RUNME` (working example)

Semantic Enrichment

ALL CAPS:

- README (plain text minimal documentation)
- RUNME (working example)
- CODE (code meta-data, i.e.: https://github.com/gramian/code-ini)

A hard problem:

A hard problem:

- *Check Code* initiative (`https://codecheck.org.uk`)

A hard problem:

- *Check Code* initiative (`https://codecheck.org.uk`)
- *Journal of Open Source Software* (`https://joss.theoj.org`)

Software Peer Review

A hard problem:

- *Check Code* initiative (`https://codecheck.org.uk`)
- *Journal of Open Source Software* (`https://joss.theoj.org`)
- Test review
  - Developer defines and describes a test,
  - based on an analytically known solution,
  - or a solution computed by alternative means.
  - Reviewer evaluates test quality and result.

Conservative values:

Conservative values:

- **Prefer** traditional publication

  → document PDF & supplemental material `zip` archive

- **Optional** experimental technologies

  → Jupyter Notebooks, Code Ocean

Conservative values:

- **Prefer** traditional publication

  → document PDF & supplemental material `zip` archive

- **Optional** experimental technologies

  → Jupyter Notebooks, Code Ocean

- Programming language longevity

  → FORTRAN, LISP, MATLAB

- Programming language readability

  → MATLAB, MAPLE

Summary

Do you want to know more?

- Empirical Gramians: `https://gramian.de`
- Best Practices: `https://doi.org/bsb2`
- Sustainable Software: `https://arxiv.org/abs/1909.09469`
- Mathematical Software: `https://doi.org/10.1016/C2013-0-11363-3`

`https://himpe.science`

Acknowledgment:
Supported by the German Federal Ministry for Economic Affairs and Energy, in the joint project: "**MathEnergy** – Mathematical Key Technologies for Evolving Energy Grids", sub-project: Model Order Reduction (Grant number: 0324019**B**).

Summary

Do you want to know more?

- Empirical Gramians: `https://gramian.de`
- Best Practices: `https://doi.org/bsb2`
- Sustainable Software: `https://arxiv.org/abs/1909.09469`
- Mathematical Software: `https://doi.org/10.1016/C2013-0-11363-3`

`https://himpe.science`

← Get slides

Acknowledgment:
Supported by the German Federal Ministry for Economic Affairs and Energy, in the joint project: "**MathEnergy** – Mathematical Key Technologies for Evolving Energy Grids", sub-project: Model Order Reduction (Grant number: 0324019**B**).