ENABLING HIGH-FIDELITY DIGITAL TWINS OF CRITICAL ASSETS VIA REDUCED ORDER MODELING

Dr. David Knezevic
Akselos CTO
1. Digital Twin Context
2. Examples
3. More Details on RB-FEA
1. Digital Twin Context
2. Examples
3. More Details on RB-FEA
Akselos’s Take on Digital Twins

At Akselos we focus on Digital Twins of large and critical engineering systems, e.g:

- Offshore structures for oil & gas (FPSOs, tankers, semisubmersibles, platforms)
- Rotating machinery (compressors, turbines, mills)
- Wind farms
- Mining machinery
- Civil infrastructure (bridges, buildings)
Akselos’s Take on Digital Twins

Operators face a number of significant challenges with systems of these types:

• Systems are large and complex, deployed for decades, often operated beyond original design life
• Failure leads to safety and environmental risks
• Downtime is extremely costly in terms of lost revenue and repairs
• Reliability is driven by structural integrity issues such as structural fatigue, buckling, and cracking
• Inspection and maintenance is often based on ad hoc rules with significant uncertainty and which require engineering judgement and interpretation
To meet these major challenges, heavy industry is increasingly looking toward Digital Twins to provide **quantitative** guidance for the inspection, maintenance, and operation of critical assets.

**Akselos Digital Twin**: Physics-based model of an entire asset which incorporates the current condition (e.g. based on inspection data), and enables fast high-fidelity structural integrity analysis, optionally linked to sensor data (IIoT).
Physics-based Digital Twins

Standard approach for physics-based structural integrity simulation is finite element analysis (FEA)

However, FEA does not meet the needs of Digital Twins for heavy industry:

• Generally not possible to provide detailed model of full systems: computational load/memory requirements are not feasible

• Not fast enough to assist with operational decision-making, or to link to sensor data

To overcome these limitations, Akselos’s platform is based on reduced order modeling

FEA workflows generally involve coarse global models since detailed global models with FEA are not feasible
What are Reduced Order Models?

Reduced order models are a family of numerical methods for reducing the computational cost associated with evaluating high-fidelity models such as FEA, CFD, finite differences.

Very active research topic in academia for at least 2 decades, with a range of different methods, e.g.:

• Proper Orthogonal Decomposition (POD)
• Reduced Basis (RB) methods
• Balanced Truncation
• Proper Generalized Decomposition (PGD)
• Response surfaces

Yields surrogate models that are well-suited in real-time or many query contexts.
RB-FEA: Enabler for Digital Twins

At Akselos, our reduced order modeling approach is called RB-FEA (Reduced Basis FEA)

RB-FEA is a component-based version of the Reduced Basis (RB) method, and is a model reduction method for large-scale parametrized partial differential equations (PDEs)

RB method: Actively developed in academia (MIT, Brown, EPFL, Paris VI, NTNU, etc.), over 100 papers published since 2000 in peer reviewed journals, e.g. see ARCME review article by Patera et al., 2007

We now provide a brief overview of RB, and then of RB-FEA
Parametrized Partial Differential Equations

Let $F$ be a PDE operator, models some physical system (e.g. structures, fluids, thermal, acoustics, electromagnetics, …)

$F$ depends on a parameter vector $\mu \in \mathcal{D} \subseteq \mathbb{R}^P$, $\mu$ characterizes system (e.g. material properties, section thickness, model geometry, loads, boundary conditions, …)

Given $\mu$ find solution $u(\mu)$ such that $F(u(\mu); \mu) = 0$

Change $\mu$ and re-solve to model different “what if” scenarios, account for uncertainty, optimize a design, incorporate measurement data, etc.
Parametrized PDE: Example

Los Montes Simple-Span Bridge

\[ -\nabla \cdot \sigma = f, \quad \text{on } \Omega, \]

\[ F: \quad \sigma \cdot n = t, \quad \text{on } \partial \Omega_t, \]

\[ u = u_d, \quad \text{on } \partial \Omega_d. \]
Parametrized Partial Differential Equations

Typical approach is to use FEA to solve for each new $\mu$.

This requires a solve time $T^{FEA}$ for each $\mu$.

If mesh is large then response time is slow, and it becomes \textit{computationally unfeasible} if many $\mu$ are required.
Parametrized Partial Differential Equations

Reduced Basis (RB) method was developed to resolve this issue via the following features:

• Offline/Online decomposition to separate model training from fast model evaluation

• Greedy algorithm based on residual-driven error estimators to adaptively select parameters during Offline stage, FEA solve at each selected parameter

• Very fast evaluation in Online stage with $T^{RB} \ll T^{FEA}$
The Reduced Basis Method

RB and FEA are based on exactly the same parametrized PDE, and both use the same Galerkin projection approach

\[ a(u_n(\mu), \nu; \mu) = f(\nu; \mu) \quad \forall \nu \in X_n \]  
\[ a(u^N(\mu), \nu; \mu) = f(\nu; \mu) \quad \forall \nu \in X^N \]

The key difference is that the RB space \( X_n \) is highly targeted at the parametric PDE, whereas FEA space \( X^N \) is very high-dimensional with no “knowledge” of the PDE
Key insight of RB: We can directly approximate the nonlinear “solution manifold,” where each point $u(\mu)$ on the manifold satisfies $F(u(\mu); \mu) = 0$.
The Reduced Basis Method: Greedy Algorithm

RB Greedy Algorithm samples the manifold by solving FEA at adaptively selected parameters to construct RB model with $n$ basis functions, where generally $n \ll N$.
The Reduced Basis Method: Greedy Algorithm

1: specify $\Xi \subset D$ of size $n^{\text{train}}$ and tolerance $\epsilon$,
   set $n = 0$, $err = \infty$
2: while $err > \epsilon$ do
3:   set $\mu_n^* = \arg \max_{\mu \in \Xi} \Delta_n^s(\mu)$; $err = \Delta_n^s(\mu_n^*)$;
4:   $\xi_{n+1} = (I - \text{proj}_{X_n})u^N(\mu_n^*)$ (normalized);
5:   compute and store Online data (blue terms);
6:   $n \leftarrow n + 1$
7: end while
The Reduced Basis Method: Greedy Algorithm

Step 4

Step 5

Greedy Algorithm converges rapidly (exponential convergence under reasonable assumptions, e.g. see Binev et al., 2010), hence we typically reach tolerance with small $n$. 
The Reduced Basis Method: Example

RB basis functions from “greedily selected parameters” – Greedy Algorithm converges with $n = 28$ for $\epsilon = 10^{-4}$ in this case
The Reduced Basis Method

The RB space from the Greedy Algorithm is highly targeted at solving $F$ for $\mu \in \mathcal{D}$ since the RB basis functions are from sampling $\mathcal{M}^N$ (i.e. from FEA solves at parameter samples).

In contrast the FEA approximation space is completely untargeted, e.g. it can represent arbitrary data irrelevant to solving $F$, such as “random noise”: 

![Diagram](image-url)
The Reduced Basis Method

FEA is computationally expensive since it must find the good solution within a huge space of mostly irrelevant data – requires solving large systems of equations.

RB approach is to first construct a targeted space during the Offline stage, then perform fast and accurate solves (which depend only on $n$, not $N$) during the Online stage.

Note that assembly of $n \times n$ RB system in Online stage is independent of $N$ since parameter-independent quantities are computed and stored during Offline.

E.g. we can evaluate RB models on a smartphone in real time, see Knezevic et al., High-Fidelity Real-Time Simulation on Deployed Platforms", Computers & Fluids, 2010.
The Reduced Basis Method: Example

Footbridge
Set RB dimension
Online N = 28
Set parameters
t_d
55.70
E_s
5400
rho_d
3070
rho_s
Solve μ Sweep

Akselos
The Reduced Basis Method
The Reduced Basis Method
Moving to Components

The single-domain RB method is very powerful for parametrized PDEs, but it has some limitations:

• The Greedy Algorithm requires an FEA solution at every selected $\mu$, but for very large-scale problems it may be unfeasible to do even a single FEA solve

• The method is restricted to relatively few parameters since otherwise we can require an impractically large $n$ to “cover” the parameter domain $\mathcal{D}$

• Geometric parametrizations must be continuous, e.g. one cannot introduce topological changes like removing a part or creating a hole
Moving to Components

To address these limitations, the component-based formulation of the RB method was developed (see Knezevic et al., M2AN, 2013), which we refer to as RB-FEA

The key idea of RB-FEA is to create RB models for components, and connect components together to form large parametrized models

(The idea of the name “RB-FEA” is that the method is built upon RB and uses components in a manner similar to how FEA uses elements)

RB-FEA uses single-domain RB method + substructuring, with a further model reduction on component interfaces via “optimal modes” (see Patera & Smetana, 2016)
Parametrized Component-Based Modeling
Parametrized Component-Based Modeling
Moving to Components

RB-FEA resolves the key limitations of single-domain RB:

• **Large models:** Component training involves individual components or small groups of components only, hence never need to solve full model with FEA during Offline stage

• **Many parameters:** Each component can have several parameters, and we may have thousands of components in a model, hence easily supports >1000 parameters

• **Topology changes:** Simply add/remove/replace components and re-solve

With RB-FEA we also retain the speed of RB: Typically observe **1000x speedup** or more compared to FEA for large-scale models
RB-FEA: Substructuring + RB

Core idea: Consider a system with two components, $\Omega_1$ and $\Omega_2$, connected on interface $P$

$$
\begin{bmatrix}
A_{P,P} & A_{P,\Omega_1} & A_{P,\Omega_2} \\
A_{P,\Omega_1}^T & A_{\Omega_1,\Omega_1} & 0 \\
A_{P,\Omega_2}^T & 0 & A_{\Omega_2,\Omega_2}
\end{bmatrix}
\begin{bmatrix}
U \\
u_{\Omega_1} \\
u_{\Omega_2}
\end{bmatrix}
=
\begin{bmatrix}
f_P \\
f_{\Omega_1} \\
f_{\Omega_2}
\end{bmatrix}
$$

Solve for the non-interface DOFs via single-domain RB method:

$$
A_{\Omega_i,\Omega_i} u_{\Omega_i} = f_{\Omega_i} - A_{P,\Omega_i}^T U
$$

Substitute into the interface DOF rows to get a system that involves interface DOFs only:

$$
A(\mu) U(\mu) = F(\mu)
$$
RB-FEA: Enabler for Digital Twins

**FEA Workflow**

1. CAD MODEL
2. MESH
3. SET UP SIMULATION
4. FEA SOLVE

Very long re-analysis process

5. SET UP SIMULATION
6. RB-FEA SOLVE

Very fast re-analysis process

**RB-FEA Workflow**

1. CAD MODEL
2. DIVIDE MODEL INTO COMPONENTS
3. MESH COMPONENTS
4. PARAMETERIZE COMPONENTS AND PRE-COMPUTE REDUCED ORDER MODELS
RB-FEA vs. FEA Example

FEA: 155 seconds

RB-FEA (Online): 0.06 seconds
RB-FEA vs. FEA Example

FEA: 155 seconds

RB-FEA (Online): 0.06 seconds
Some Key Features of Akselos’s Digital Twin Platform

• RB-FEA algorithms + Akselos Cloud platform scales efficiently to very large models – Enables detailed models of full assets, cloud brings scalability and collaboration

• RB-FEA is fast, e.g. 1000x faster than FEA – “Keep up” with decision-making and sensor data, or perform many solves within big data, optimization, ML, or UQ loops

• Models are easily modifiable via parameters or component replacement – Incorporate updates from inspections or sensors

• RB-FEA, FEA, Hybrid Solver provide the full range of linear and nonlinear analysis – Perform whichever type of structural analysis is relevant (stress, fatigue, buckling, elastoplasticity, contact, friction, thermal expansion, etc.)
Akselos Cloud

Digital Twin Monitoring
Web-based asset monitoring shows current status of each asset based on its Digital Twin

Cloud-based digital twins for live assets from any office around the world
1. Digital Twin Context
2. Examples
3. More Details on RB-FEA
Floating Structures
Floating Structures

Stage 1: Create CAD model based on drawings
Floating Structures

Stage 2: Create and train RB-FEA components
Floating Structures

Stage 3: Assemble global model by connecting components (868 components in this case), set parameters and load cases
Floating Structures

Stage 4: Perform fast, detailed, global RB-FEA solves

Zoom-in of the global model
Floating Structures

In contrast see conventional workflow, e.g. DNV-RP-C206
Floating Structures: Pre- and Post-processing

- Hydrodynamic loading: Interface to WAMIT to impose wave loads

- Standards-based checks (DNV-GL, LR, ABS, etc.) for buckling and fatigue based on post-processing stresses for many load cases

\[
\frac{\tau_d}{\tau_c} \leq \frac{1}{\lambda_1} \\
\frac{\sigma_d}{\sigma_{d(i)}} \leq \frac{1}{\lambda_2} \\
\frac{\sigma_d}{\sigma_{d(w)}} \leq \frac{1}{\lambda_3} \\
\sigma_{c(i)} > \sigma_{c(l)} \\
\sigma_{c(w)} > \sigma_{c(l)} \\
\sigma_{c(l)} > \sigma_{c(l)} \\
\left(1 + 0.6A_R \right) \left(\frac{\sigma_{dY}}{\sigma_Y} \right) + \left(\frac{\tau_d}{\tau_c} \right)^2 \leq 1 \\
\frac{\sigma_d}{\sigma_c} \leq 1,0 \frac{1}{\lambda_4} \\
\frac{\tau_d}{\tau_c} \leq \frac{1}{\lambda_5} \\
\frac{\sigma_{dk}}{\sigma_{ex}} + \frac{\sigma_{dy}}{\sigma_{cy}} \leq G \\
0,625 \left( 1 + \frac{0.6}{A_R} \right) \left( \frac{\sigma_{dY}}{\sigma_{cy}} \right) + \left( \frac{\tau_d}{\tau_c} \right)^2 \leq 1 \\
1 - 0,625 \left( \frac{\sigma_{dx}}{\sigma_{cx}} \right) + 1 - \left( \frac{\sigma_{dx}}{\sigma_{cx}} \right) \leq 1
\]
Floating Structures: Pre- and Post-processing

- Update thicknesses based on inspection results and re-run wave-loading and code-checks: Fully automated and fast

- Vary thickness parameters to examine many corrosion scenarios (e.g. “Monte Carlo” analysis) and identify high risk regions to focus inspections

Thickness scaled by 0.9
Floating Structures: Pre- and Post-processing

- J-integral based crack-propagation within the global Digital Twin

- Nonlinear analysis in critical regions (e.g. elastoplasticity, contact, post-buckling) to use the Digital Twin to go beyond the standards
Floating Structures

Semi-submersible model developed with DNV-GL
Fixed Structures
Fixed Offshore Structures

Fixed offshore structures are used in shallow water.
Fixed Offshore Structures

Fast solves with beams-only (traditional approach), beams+RB-FEA, or all RB-FEA
Fixed Offshore Structures
Fixed Offshore Structures

Requires soil/pile modeling (nonlinear springs)

Also, pre/post-processing for fixed structures is quite different from floating structures:

- Wave loading on tubular members based on various wave theories (Airy, Stokes, Dean Stream, etc.) with Morison equation

- Specific standards for buckling and fatigue (ISO 19902, API, AISC, etc.)
Shell Joins Digital Twin Project For Offshore Structures Led By Akselos

Operators investing in technology to monitor critical assets and hence reduce the number of personnel that need to be sent offshore for inspection, maintenance, repair (safety risks, very expensive)
Digital Twins Project: Demonstrate Sensor-Enabled Models

Link sensor data from offshore platform (fixed platform in this case) to RB-FEA Digital Twin
Sensor-Connected Digital Twins Project

Key points regarding sensors:

- Typically only up to 10 structural sensors are used per asset, hence RB-FEA model has the critical role of giving insight into the rest (the “unmeasured” parts)
- Reconstruct displacement history of entire structure from accelerometer data, enables fatigue, risk, and asset life-extension analysis based on true conditions
- Wave buoys measure true ocean conditions which can be imposed on the Digital Twin (instead of using statistical metocean data)
- Feed output from sensor-integrated RB-FEA model into ML to identify and classify anomalies in the measurements for further investigation by engineers
Port infrastructure & heavy equipment
Port Infrastructure
Pressure Vessels
Compressors
Bearings
1. Digital Twin Context
2. Examples
3. More Details on RB-FEA
Classical substructuring typically uses “all” DOFs on an interface, which can be very slow (comparable computational cost to a full FEA solve).

We advocate truncated port representations, similar to CMS and Craig-Bampton, using a new formulation called “Optimal Modes”.

This approach was developed by Patera & Smetana (e.g. Patera & Smetana, SIAM J. Sci. Comput., 2016).

General idea is similar to RB in the sense that we reduce the interface space in a targeted manner during the Offline stage.
Optimal Modes: Motivating Example

Consider

$$-\Delta u = 0 \quad \text{in} \quad \Omega = (-L, L) \times (0, 1)$$

- homogeneous Neumann b.c. at $y = 0$ and $y = 1$
- arbitrary Dirichlet data on $\Gamma_1$ and $\Gamma_2$

Separation of variables:

All harmonic functions on $\Omega$ have the form

$$u(x, y) = a_0 + b_0x + \sum_{n=1}^{\infty} \cos(n\pi y) \left( a_n \cosh(n\pi x) + b_n \sinh(n\pi x) \right)$$

Rapid exponential decay of $\cosh$ for higher $n$ in the interior of $\Omega$

Harmonic extensions of basis $\{ \cos(n\pi y) : n = 0, \ldots, \infty \}$ on $\Gamma_1, \Gamma_2$ are almost always close to zero on the interface $\Gamma_{12}$
Optimal Modes: Transfer Eigenproblem

In general, for a harmonic function $u$ the map from the Dirichlet boundary to the interface can be described by a transfer operator $P(u|_{\Gamma_1 \cup \Gamma_2}) := u|_{\Gamma_{12}}$

The best approximation space on the interface can be determined by solving the (hermitian) transfer eigenproblem

$$P^* P u|_{\Gamma_1 \cup \Gamma_2} = \lambda u|_{\Gamma_{12}}$$

for the dominant eigenmodes.

If the eigenvalues decay quickly, a small approximation space on the interface provides a good approximation.

Parameter dependency: The transfer eigenproblem is instantiated for a (small) number of samples. The final (parameter-independent) port space is obtained as the dominant subspace of the individual port spaces.
Optimal Modes: Example

Linear Elasticity: Eigenvalue decay for two connected I-Beams with and without a crack
RB-FEA: Incorporating Nonlinear Analysis

The single-domain RB method applies to nonlinear analysis, but substructuring is linear-only.

Hence our approach for nonlinear analysis is to provide coupled RB-FEA/FEA models (Hybrid Solver) with continuity constraints on the interface.

RB-FEA in “linear regions”, FEA in “nonlinear regions”: Enables full range of nonlinear analysis to be incorporated within large-scale Digital Twins.

Full details of formulation in Full details in Knezevic et al., CMAME, 2017.
Hybrid Solver

**Core idea:** Formulate as a coupled nonlinear system

\[ G(U) = 0, \quad U \in \mathbb{R}^{N_{SCRBE} + N_{FE}} \]

Use constraint matrix to rewrite residual and Jacobian in terms of (linear) RB-FEA and (nonlinear) FEA terms:

\[
G(U) = C^T \begin{bmatrix} F(\mu) - A(\mu)U(\mu) & G_{FE}(U) \\ \end{bmatrix}
\]

\[
J_G(U) = C^T \begin{bmatrix} -A(\mu) & 0 \\ 0 & J_{G_{FE}}(U) \\ \end{bmatrix} C
\]
Hybrid Solver

Apply Newton’s method (with line search) to the coupled system:

\[ J_G(U^k) \Delta U^k = -G(U^k) \]

\[ U^{k+1} = U^k + \Delta U^k \]

Key points:
- Arbitrary nonlinearities in FEA region (e.g. contact, elastoplasticity, finite strain)
- RB-FEA accelerates the linear region, very fast for “localized nonlinearities”
- Formulation is fully conforming, numerically robust
Hybrid Solver

Elastoplasticity

Contact