RDF: Reduced Basis, Domain Decomposition and Finite Elements
A Combined Perspective

Laura Iapichino ▪ Matteo Lesinigo ▪ Alfio Quarteroni ▪ Gianluigi Rozza

Introducing RDF

The benefits of Reduced Basis, Domain Decomposition and Finite Elements
for parametrized PDEs in networks of repetitive blocks with heterogeneous parametrizations

RDF = RB + DD + FE

RDF is a numerical framework to perform rapid and reliable simulations by combining Reduced Basis (RB), Domain Decomposition (DD) and Finite Elements (FE) techniques, for solving parametrized problems (μPDEs) in networks made up by repetitive blocks with heterogeneous parametrizations. The method is applicable to general time-independent linear μPDEs and the continuity of the global solution is assured by a classical domain decomposition approach.

The idea in four steps

1. Consider every block as a parametric deformation of a reference block.
2. Build *offline* local reduced basis spaces by computing once and for every single reference block, some representative solutions for different values of the parameters through FE. The reduced basis must be able to approximate different boundary conditions (a basis for the trace of the solution) on the boundaries corresponding to the internal interfaces Γ of the network.
3. Partition the mesh nodes in the subsets Ω_{RB}, where the solution will be obtained by RB, and Ω_{FE}, where the solution will be obtained by FE.
4. Recover *online* the global solution by a Galerkin projection on the reduced basis spaces in Ω_{RB} and on the finite element spaces in Ω_{FE}.

Where does the error come from?

u: Exact solution u_{FE}: FE solution u_{RDF}: RDF solution
u_{BC}: Auxiliary FE function that, among the ones whose trace on Γ is a linear combination of the local BC nodal values, minimizes $||u_{FE} - u_{BC}||_V$

$$||u - u_{RDF}||_V \le ||u - u_{FE}||_V + ||u_{FE} - u_{BC}||_V + ||u_{BC} - u_{RDF}||_V$$

- Can be reduced by improving the local FE discretization of the reference blocks by decreasing the mesh size h.
- Can be reduced by enriching the local RB space to account for more possible boundary conditions.
- Can be reduced by increasing the number of local reduced basis functions as it is proportional to the RB greedy error.

The Ingredients

Finite Element Method

The Finite Element method consists in solving a PDE problem by variational approach, performing a Galerkin projection on a given discrete space V_h. Given a regular triangulation τ_h of the domain Ω, the FE method corresponds to choosing V_h as the space of the functions that are globally continuous on Ω and that, restricted to an element $K \in \tau_h$ are polynomial.

Strong Formulation

$$Lu = f \ \text{in} \ \Omega, \qquad Bu = g \ \text{on} \ \partial\Omega$$

Variational Formulation

$$u_h \in V_h \quad a(u_h, v_h) = F(v_h) \quad \forall v_h \in V_h$$

Space for the discrete solution

$$V_h \equiv \{v_h \in C^0(\bar\Omega) : v_h|_K \in \mathbb{P}_k \ \forall K \in \mathcal{T}_h\}$$

Algebraic Formulation

$$A^{FE}\mathbf{u} = \mathbf{f}^{FE}$$

Domain Decomposition

The Domain Decomposition Method consists in partitioning the domain Ω into the union of smaller subdomains Ω_i and in solving smaller local problem. In order to assure that the local solutions are the restriction of the global solution, suitable coupling conditions must be provided on the fictitious internal interfaces, Γ_{ij}, introduced by the domain decomposition.

Strong Formulation

$$\begin{aligned} Lu_i &= f && \text{in} \ \Omega_i \\ Bu_i &= g && \text{on} \ \partial\Omega_i \setminus \Gamma_{ij} \\ Bu_i &= Bu_j && \text{on} \ \Gamma_{ij} \end{aligned}$$

Algebraic Formulation

$$\begin{bmatrix} A_1 & 0 & 0 & A_{1\Gamma} \\ 0 & \ddots & 0 & \vdots \\ 0 & 0 & A_{N_i} & A_{N_i\Gamma} \\ A_{\Gamma 1} & \dots & A_{\Gamma N_i} & A_\Gamma \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_{N_i} \\ \mathbf{u}_\Gamma \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_{N_i} \\ \mathbf{f}_\Gamma \end{bmatrix}$$

Reduced Basis Method

The Reduced Basis method consists in solving a parametrized PDE problem by variational approach, performing a Galerkin projection on a particular discrete space V_N. Particularly, a set of solutions corresponding to different values of the parameters is precomputed during an *offline* stage by FEM (greedy). In a many query context, each new evaluation is then performed in an *online* step by choosing V_h as the space generated by the precomputed solutions.

Strong Formulation

$$L_\mu u = f_\mu \ \text{in} \ \Omega_\mu, \qquad B_\mu u = g_\mu \ \text{on} \ \partial\Omega_\mu$$

Variational Formulation

$$u_N \in V_N \quad a_\mu(u_N, v_N) = F_\mu(v_N) \quad \forall v_N \in V_N$$

Space for the discrete solution

$$V_N = \text{span}\{u_h^{FEM}(\mu_i), i = 1, \dots, N_{sample}\}$$

Algebraic Formulation

$$\Psi^T A^{FE}_\mu \Psi \mathbf{y} = \Psi^T \mathbf{f}^{FE}_\mu$$

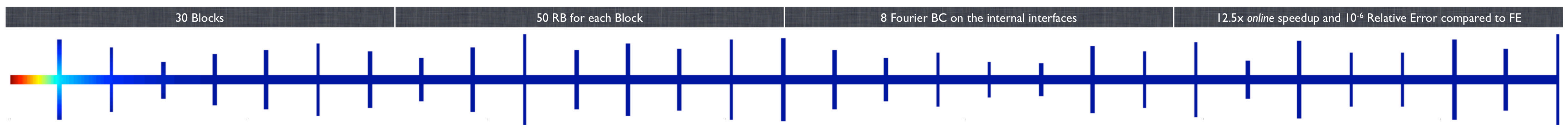
RDF in Action

All you need to know, for the thermal fin problem

The problem

Consider the problem of the heat conduction in a thermal dissipator made of a certain number R of blocks with fins of variable length. This problem may be described as a parametric Laplace problem where the parameters vector μ includes the length of each fin μ_θ and the heat conductivity of each block μ_c.

30 Blocks | 50 RB for each Block | 8 Fourier BC on the internal interfaces | 12.5x online speedup and 10^{-6} Relative Error compared to FE



Strong Formulation

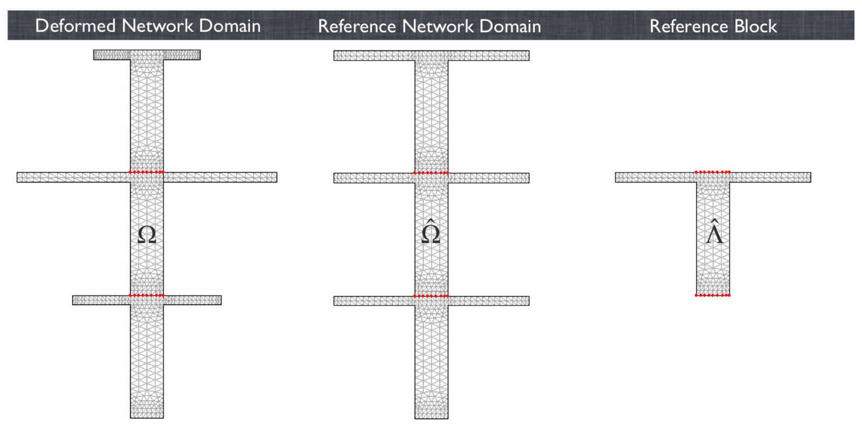
$$\begin{cases} \mu_c \Delta u = 0 & \text{in} \ \Omega_{\mu_\theta} \\ u = 1 & \text{on} \ \Gamma_{In} \\ u = 0 & \text{on} \ \Gamma_{Out} \\ \mu_c \frac{\partial u}{\partial n} = 0 & \text{on} \ \Gamma_{Wall} \end{cases}$$

Variational Formulation

$$a_\mu(u, v) = F_\mu(v)$$
$$a_\mu(u, v) = \sum_{r=1}^{R} \int_{\Omega_{\mu_\theta}^r} \mu_c^r \nabla u \cdot \nabla v \, dx$$
$$F_\mu(v) = -\sum_{r=1}^{R} \int_{\Omega_{\mu_\theta}^r} \mu_c^r \nabla R_g \cdot \nabla v \, dx$$

1 From the deformed configuration to the reference blocks

Deformed Network Domain | Reference Network Domain | Reference Block

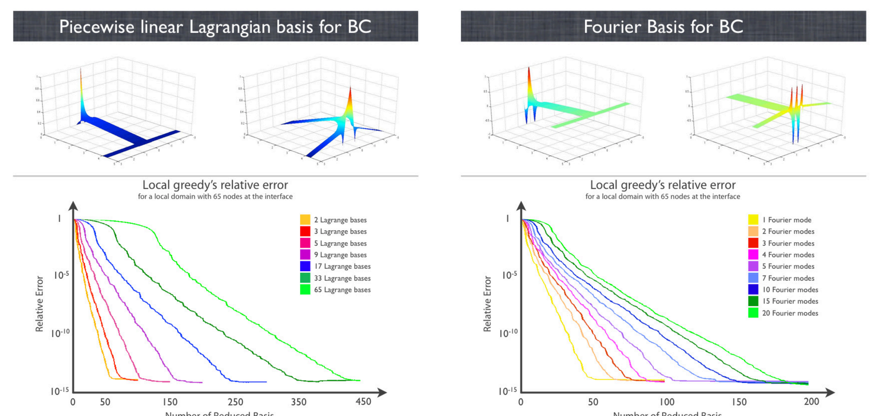


RDF method deals with three different geometries. The *deformed network domain*, where we are interested in obtaining the solution, the *reference network domain*, where the variational problem is formulated and the solution is obtained in the *online* phase and the *reference blocks*, where the local reduced basis spaces are built in the *offline* phase. Affine maps are used to easily pass from one geometry into another and let us incorporate the geometrical parameters in the variational formulation:

$$\begin{aligned} a_\mu(u, v) &= \sum_{r=1}^{R} \int_{\Omega_{\mu_\theta}^r} \mu_c^r \nabla u \cdot \nabla v \, dx = \sum_{r=1}^{R} \int_{\hat\Omega} \mu_c^r \, C_{\mu_\theta^r}^{-T} \nabla \hat u \cdot C_{\mu_\theta^r}^{-T} \nabla \hat v \, |\det C_{\mu_\theta^r}| \, d\hat x \\ &= \sum_{r=1}^{R} \sum_{i,j} \underbrace{\mu_c^r \left[(C_{\mu_\theta^r}^{-1} C_{\mu_\theta^r}^{-T})_{ij} \right] |\det C_{\mu_\theta^r}|}_{\Theta^r_{ij}(\mu)} \underbrace{\int_{\hat\Omega} \frac{\partial \hat u}{\partial \hat x_i} \frac{\partial \hat v}{\partial \hat x_j} \, d\hat x}_{a^r_{ij}(\hat u, \hat v)} = \sum_{r=1}^{R} \sum_{i,j} \Theta^r_{ij}(\mu) \, a^r_{ij}(\hat u, \hat v) \end{aligned}$$

2 Local greedy algorithms & the missing boundary conditions

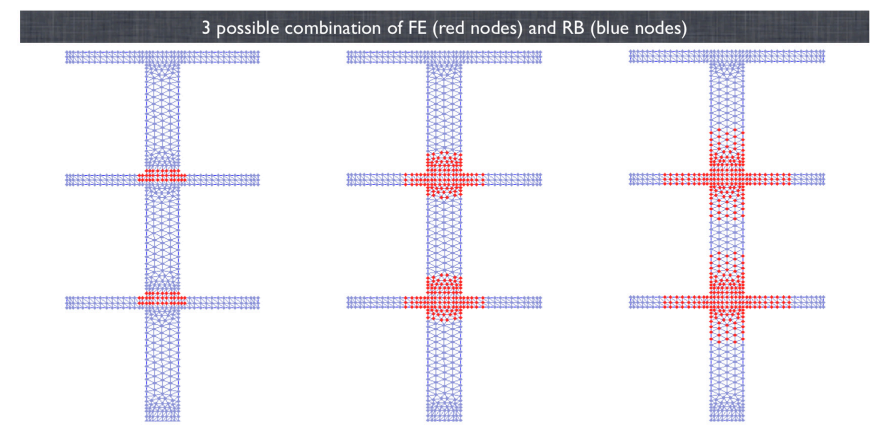
Piecewise linear Lagrangian basis for BC | Fourier Basis for BC



Classical greedy techniques are used on the reference blocks to build the local RB spaces. The fin lenght μ_θ is the only parameter to take into account as the heat conductivity does not modify the local solution for this specific problem.
A dicotomy appears at this stage: the global solution is unknown and so is its trace on the local boundaries mapped into internal interface of the global network. However, some boundary conditions (BC) must be provided for building the RB spaces. The idea is to perform the greedy by choosing the solutions in a set where both μ_θ and the boundary condition vary. Ideally the boundary conditions should constitute a basis for the trace of the FE solution on the internal interfaces. We used either a piecewise linear Lagrangian basis (of different amplitude) or a Fourier basis (with different modes).

3 A little bit of Reduced Basis & a little bit of Finite Elements

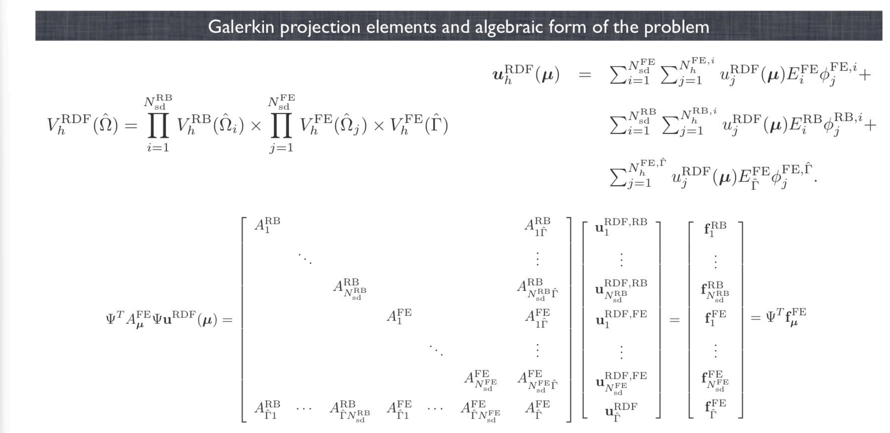
3 possible combination of FE (red nodes) and RB (blue nodes)



RDF method allows to arbitrarily mixing reduced basis and finite elements on each block. Once the local reduced bases have been obtained, we select the nodes to be treated by the FE method and the ones to be treated by the RB method.
Usually, zones where the solution does not depend smoothly on the parameters or where the RB error is concentrated are the ones where the usage of the FE method is indicated. On the other side, where the solution depends smoothly on the parameters, RB method can be used effectively. Once the division is made, we drop the component corresponding to the FE nodes from the local reduced basis and we orthonormalize the set of basis functions to improve the properties of the local reduced basis.

4 Galerkin projection put everything together

Galerkin projection elements and algebraic form of the problem

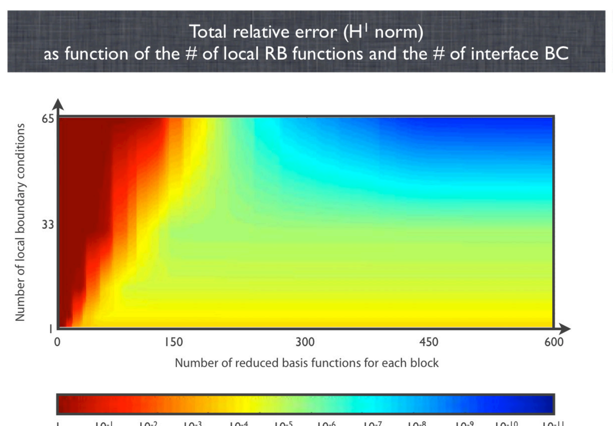


The final step of our method consists in performing a Galerkin projection on the discrete RDF space which is obtained by direct product of the local discrete FE and RB spaces. Once proper extension operator E_i are introduced that prolongate globally to zero the local functions, deducing the algebraic form of the problem is straightforward. Particularly we can reorganize the terms to isolate the single subdomains and the single FE and RB contribution. The extra-diagonal blocks provide the correct coupling conditions. Besides the RDF matrix can be obtained by pre and post multiplying the global FE matrix by a proper block diagonal matrix Ψ that contains identity blocks in correspondence of the FE nodes and the local RB functions blocks in correspondence to the RB nodes.

RDF Performances & Errors

Averaged results on 100 casual parametric samples of a 3 blocks network configuration with a mesh of 37505 nodes and 65 local interface nodes

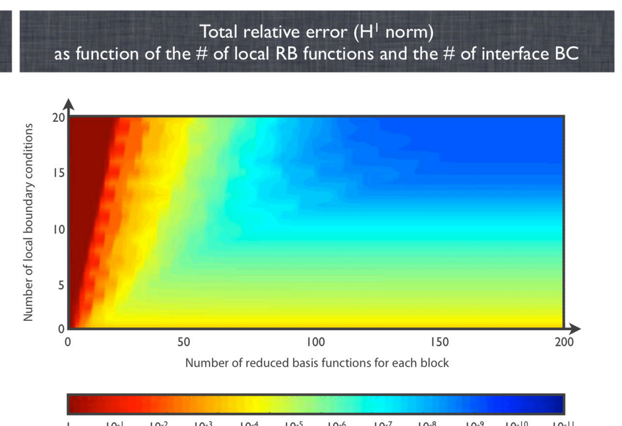
Total relative error (H^1 norm) as function of the # of local RB functions and the # of interface BC



Type of local BC: Piecewise Linear Lagrangian
Number of FE Layers: 0

Relative error (in H^1 norm) between the RDF solution and the FE solution. For a fixed number of local BC, the error decreases by augmenting the number of local RB functions and reaches a plateau that depends on the number of BC considered. For a fixed number of local RB functions the error diminishes by increasing the number of local BC, however, if the number of local RB functions is too small, the error increases as the RB is not large enough to accurately approximate solutions corresponding to all the different local BC. FE are used only for the nodal value on the internal interfaces.

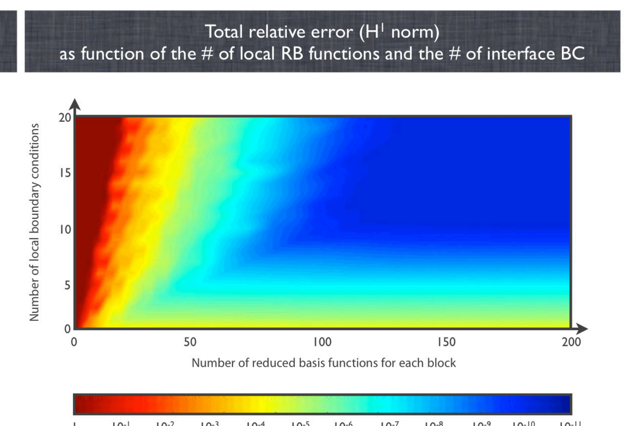
Total relative error (H^1 norm) as function of the # of local RB functions and the # of interface BC



Type of local BC: Fourier
Number of FE Layers: 0

Relative error (in H^1 norm) between the RDF solution and the FE solution. For a fixed number of local BC, the error decreases by augmenting the number of local RB functions and reaches a plateau that depends on the number of BC considered. For a fixed number of local RB functions the error diminishes by increasing the number of local BC, however, if the number of local RB functions is too small, the error increases as the RB is not large enough to accurately approximate solutions corresponding to all the different local BC. FE are used only for the nodal value on the internal interfaces.

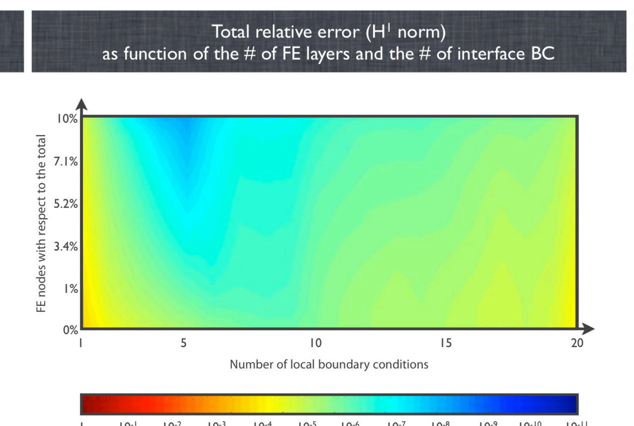
Total relative error (H^1 norm) as function of the # of local RB functions and the # of interface BC



Type of local BC: Fourier
Number of FE Layers: 5 (FE nodes are 4.3% of the total)

Relative error (in H^1 norm) between the RDF solution and the FE solution. For a fixed number of local BC, the error decreases by augmenting the number of local RB functions and reaches a plateau that depends on the number of BC considered. For a fixed number of local RB functions the error diminishes by increasing the number of local BC, however, if the number of local RB functions is too small, the error increases as the RB is not large enough to accurately approximate solutions corresponding to all the different local BC. In this case the FE zone at the interface has been expanded locally by adding 5 additional FE layers.

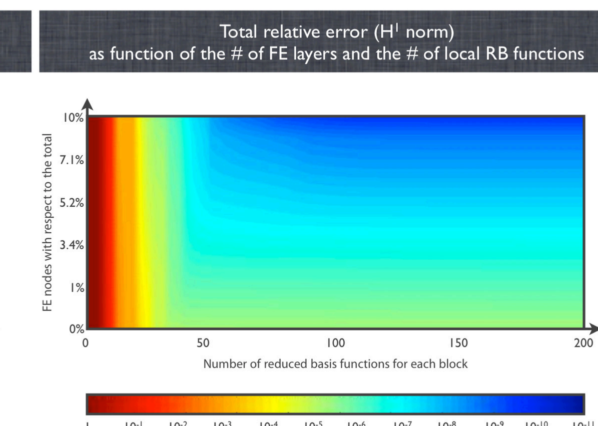
Total relative error (H^1 norm) as function of the # of FE layers and the # of interface BC



Type of local BC: Fourier
Number of local RB: 50

Relative error (in H^1 norm) between the RDF solution and the FE solution. The error diminishes by increasing the number of FE layers (percentage of FE nodes with respect to the total) and it is non monotonic w.r.t. the number of local interface BC. This is due to the fact that, at fixed number of local RB functions, too many different local BC cannot be properly approximated with the RB method. This result shows that the expansion of FE nodes by adding layers proximal to the internal interfaces at fixed number of local RB functions is particularly effective for a specific range of the number of local BC.

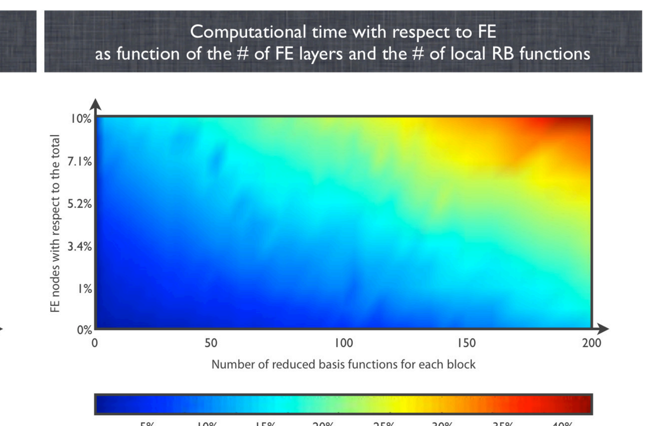
Total relative error (H^1 norm) as function of the # of FE layers and the # of local RB functions



Type of local BC: Fourier
Number of local BC: 5

Relative error (in H^1 norm) between the RDF solution and the FE solution. The error diminishes both by increasing both the number of local RB functions and the number of FE layers (percentage of FE nodes with respect to the total). Starting from the internal interfaces, each layer adds to the FE domain the nodes of the triangles contiguous to the previous layer. Therefore, increasing the percentage of FE nodes is a viable strategy to decrease the error. When FE nodes are 100% of the total (limit case), RDF correspond to the FE method on the global domain. The graphs relative to different number of local BC present similar patterns.

Computational time with respect to FE as function of the # of FE layers and the # of local RB functions



Type of local BC: Fourier
Number of local BC: Result independent of the number of local BC

The graph shows the computational time of the RDF method as percentage of the FE method applied to the global problem. The computational time increases both augmenting the number of local RB functions and the number of FE layers (percentage of FE nodes with respect to the total). Starting from the internal interfaces, each layer adds to the FE domain the nodes of the triangles contiguous to the previous layer. Selecting a reference time gain, it is possible to use isolines to search the best combination between the number of FE layers and that of local RB functions that minimizes the error.

A little more about RDF

Tuning RDF, flexibility in your hands

RDF method is very flexible. In fact, to achieve a desired accuracy for the RDF solution in a competitive time, RDF let you modify the number of local RB functions, the number (and the type) of local BC used for building the RB spaces and the ratio between the FE nodes and the total number of nodes. In general, by increasing the value of these parameters one improves the accuracy of the numerical RDF solution, however, it may also increase the *online* or *offline* computational times.

	Number of local RB functions		Number of local BC for the interface		Number of FE layers	
↑	Error Decreases	Slower online RDF solution Bigger full and smaller sparse local submatrices	Error Decreases	Slower offline greedies More local RB needed for the same greedy's tolerance	Error Decreases	Slower online RDF solution Bigger sparse and smaller full local submatrices
↓	Faster online RDF solution Bigger sparse and smaller full local submatrices	Error Increases	Faster offline greedies Less local RB needed to reach the same greedy's tolerance	Error Increases	Faster online RDF solution Bigger full and smaller sparse local submatrices	Error Increases

Increasing the Blocks

Averaged results for casual parametric network configurations made of R blocks with 40 local RB functions, 3 Fourier BC functions and 0 FE layers (FE nodes only on Γ)

	R=20	R=40	R=60	R=80	R=100
Relative Error	3.4e-5	5.8e-6	1.2e-5	5.6e-6	1.2e-5
FE Time Ratio	2.6%	3.8%	5.5%	7.3%	11.9%

RDF method retains its efficiency when increasing the number of blocks *R*. The relative error remains almost constant depending on the local RB spaces while the computational gain slightly decreases. However DD preconditioning strategies may be adopted to improve the efficiency of the method for large number of blocks.

Some References

- **L. Iapichino, M. Lesinigo, A. Quarteroni and G. Rozza**
  Reduced basis, finite element method and domain decomposition: a combined perspective.
  (In preparation, 2012).
- **L. Iapichino, A. Quarteroni, and G. Rozza**
  A reduced basis hybrid method for the coupling of parametrized domains represented by fluidic networks.
  Comput. Methods Appl. Mech. Engrg., 221-222:63-82, 2012.
- **Y. Maday and E. M. Rönquist**
  The reduced basis element method: Application to a thermal fin problem.
  SIAM J. Sci. Comput., 26(1):240ñ258, 2004.
- **G. Rozza, D.B.P. Huynh, and A.T. Patera**
  Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations.
  Arch. Comput. Methods Engrg., 15:229–275, 2008.
- **G. Rozza**
  Reduced-basis methods for elliptic equations in sub-domains with a posteriori error bounds and adaptivity.
  Appl. Numer. Math., 55(4):403-424, 2005.
- **A. Quarteroni and A. Valli**
  Domain Decomposition Methods for Partial Differential Equations.
  Oxford University Press, Oxford, 1999.

Conclusions & Remarks

RDF:

- ☆ Is a new computational framework, effectively combining RB, FE and DD
- ☆ Is particularly suitable for μPDEs in networks made up by repetitive geometries
- ☆ Is applicable to general time-independent linear μPDEs
  (already implemented and tested on Laplace and Stokes problems)
- ☆ Drastically reduces the *offline* time for large networks
  (w.r.t. the RB method applied on the whole domain, as the greedy is made only on the local reference blocks)
- ☆ Allows a flexible balance of accuracy, *offline* and *online* time
  (by changing the number of local RB, the kind and number of local interface BC and the percentage of FE nodes)
- ☆ Is suitable for applying classical algebraic DD preconditioning strategies
  (Dirichlet-Neumann, Neumann-Neumann and Robin-Robin algebraic preconditioners have already been implemented and tested for a 2 domain configuration on the Laplace problem)
- ☆ Theoretical analysis and extensions are currently under investigation
  (with focus on A-priori error estimates, improving the local BC, extending the method to nonlinear problems)

laura.iapichino@epfl.ch

CMCS - Modeling and Scientific Computing Chair - École Polytechnique Fédérale de Lausanne - Lausanne, Switzerland