Random Batch Method and Emergence of Domain Decomposition

Martín Hernández
joint work with E. Zuazua.
FAU, Department of Mathematics.
martin.hernandez@fau.de

08/2024

Friedrich-Alexander-Universität
Naturwissenschaftliche Fakultät

Friedrich-Alexander-Universität
DYNAMICS, CONTROL, MACHINE LEARNING AND NUMERICS

DAAD
Deutscher Akademischer Austauschdienst
German Academic Exchange Service

Agencia Nacional de Investigación y Desarrollo

TRR 154

Overview

1. RBM for the heat equation
2. Application to PDEs on networks
3. Numerical implementations

RBM for the heat equation

Model

Let $T > 0$. Consider the heat equation

$$\begin{cases} \partial_t y - \partial_{xx} y = f & (x,t) \in (0,1) \times (0,T), \\ y(0,t) = y(1,t) = 0 & t \in (0,T), \\ y(x,0) = y_0(x) & x \in (0,1), \end{cases} \tag{1}$$

with initial condition $y_0 \in L^2(0,1)$ and source term $f \in L^2(0,T;0,1)$.

Model

Let $T > 0$. Consider the heat equation

$$\begin{cases} \partial_t y - \partial_{xx} y = f & (x,t) \in (0,1) \times (0,T), \\ y(0,t) = y(1,t) = 0 & t \in (0,T), \\ y(x,0) = y_0(x) & x \in (0,1), \end{cases} \tag{1}$$

with initial condition $y_0 \in L^2(0,1)$ and source term $f \in L^2(0,T;0,1)$.

Goal: Develop a randomized algorithm for (1) that reduces its computational cost.

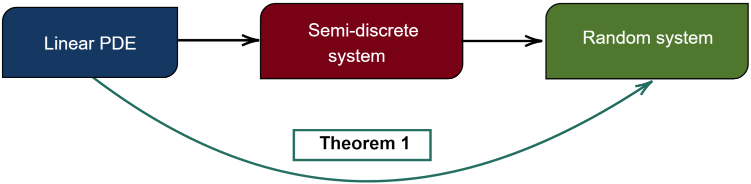
Model

Let $T > 0$. Consider the heat equation

$$\begin{cases} \partial_t y - \partial_{xx} y = f & (x,t) \in (0,1) \times (0,T), \\ y(0,t) = y(1,t) = 0 & t \in (0,T), \\ y(x,0) = y_0(x) & x \in (0,1), \end{cases} \tag{1}$$

with initial condition $y_0 \in L^2(0,1)$ and source term $f \in L^2(0,T;0,1)$.

Goal: Develop a randomized algorithm for (1) that reduces its computational cost.



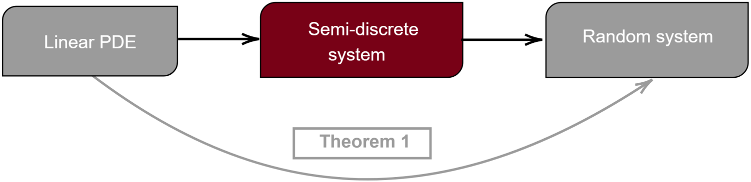
Model

Let $T > 0$. Consider the heat equation

$$\begin{cases} \partial_t y - \partial_{xx} y = f & (x,t) \in (0,1) \times (0,T), \\ y(0,t) = y(1,t) = 0 & t \in (0,T), \\ y(x,0) = y_0(x) & x \in (0,1), \end{cases} \tag{2}$$

with initial condition $y_0 \in L^2(0,1)$ and source term $f \in L^2(0,T;0,1)$.

Goal: Develop a randomized algorithm for (2) that reduces its computational cost.



FEM discretization

Consider the finite-dimensional space V_h (with basis $\{\phi_j\}_{j=1}^{N}$).

FEM discretization

Consider the finite-dimensional space V_h (with basis $\{\phi_j\}_{j=1}^N$).

Semi-discrete problem: Find $y_h \in C^1(0, T; V_h)$ such that

$$\begin{cases} \int_0^L \partial_t y_h(x,t)\phi_j(x)\,dx + \int_0^L \partial_x y_h(x,t)\partial_x\phi_j(x)\,dx = \int_0^L f(x,t)\phi_j(x)\,dx, \\ \int_0^L y_h(x,0)\phi_j(x) = \int_0^L y_h^0(x)\phi_j(x)\,dx, \quad j \in \{1,\ldots,N\}. \end{cases} \tag{3}$$

FEM discretization

Consider the finite-dimensional space V_h (with basis $\{\phi_j\}_{j=1}^N$).

Semi-discrete problem: Find $y_h \in C^1(0,T;V_h)$ such that

$$\begin{cases} \displaystyle\int_0^L \partial_t y_h(x,t)\phi_j(x)\,dx + \int_0^L \partial_x y_h(x,t)\partial_x \phi_j(x)\,dx = \int_0^L f(x,t)\phi_j(x)\,dx, \\ \displaystyle\int_0^L y_h(x,0)\phi_j(x) = \int_0^L y_h^0(x)\phi_j(x)\,dx, \quad j \in \{1,\dots,N\}. \end{cases} \tag{3}$$

Writing $y_h(x,t) = \sum_{j=1}^N y_j(t)\phi_j(x)$ and $y_h^0(x) = \sum_{j=1}^N y_j^0 \phi_j(x)$,

FEM discretization

Consider the finite-dimensional space V_h (with basis $\{\phi_j\}_{j=1}^N$).

Semi-discrete problem: Find $y_h \in C^1(0,T;V_h)$ such that

$$\begin{cases} \int_0^L \partial_t y_h(x,t)\phi_j(x)\,dx + \int_0^L \partial_x y_h(x,t)\partial_x\phi_j(x)\,dx = \int_0^L f(x,t)\phi_j(x)\,dx, \\ \int_0^L y_h(x,0)\phi_j(x) = \int_0^L y_h^0(x)\phi_j(x)\,dx, \quad j\in\{1,\dots,N\}. \end{cases} \tag{3}$$

Writing $y_h(x,t) = \sum_{j=1}^N y_j(t)\phi_j(x)$ and $y_h^0(x) = \sum_{j=1}^N y_j^0\phi_j(x)$,problem (3) can be written as

$$\begin{cases} E_h\partial_t\vec{y}_h + R_h\vec{y}_h = \vec{f}_h, \quad t\in(0,T), \\ \vec{y}_h(0) = \vec{y}_h^0, \end{cases} \tag{4}$$

FEM discretization

Consider the finite-dimensional space V_h (with basis $\{\phi_j\}_{j=1}^N$).

Semi-discrete problem: Find $y_h \in C^1(0,T;V_h)$ such that

$$\begin{cases} \displaystyle\int_0^L \partial_t y_h(x,t)\phi_j(x)\,dx + \int_0^L \partial_x y_h(x,t)\partial_x\phi_j(x)\,dx = \int_0^L f(x,t)\phi_j(x)\,dx, \\ \displaystyle\int_0^L y_h(x,0)\phi_j(x) = \int_0^L y_h^0(x)\phi_j(x)\,dx, \quad j\in\{1,\dots,N\}. \end{cases} \tag{3}$$

Writing $y_h(x,t) = \sum_{j=1}^N y_j(t)\phi_j(x)$ and $y_h^0(x) = \sum_{j=1}^N y_j^0\phi_j(x)$,problem (3) can be written as

$$\begin{cases} E_h\partial_t \vec{y}_h + R_h\vec{y}_h = \vec{f}_h, \quad t\in(0,T), \\ \vec{y}_h(0) = \vec{y}_h^0, \end{cases} \tag{4}$$

where $\vec{y}_h(t) = (y_1(t),\dots,y_N(t))$ and $\vec{f}_h$ is the vector with coefficients $(f,\phi_j)_{L^2}$.

FEM discretization

Consider the finite-dimensional space V_h (with basis $\{\phi_j\}_{j=1}^N$).

Semi-discrete problem: Find $y_h \in C^1(0,T;V_h)$ such that

$$\begin{cases} \int_0^L \partial_t y_h(x,t)\phi_j(x)\,dx + \int_0^L \partial_x y_h(x,t)\partial_x \phi_j(x)\,dx = \int_0^L f(x,t)\phi_j(x)\,dx, \\ \int_0^L y_h(x,0)\phi_j(x) = \int_0^L y_h^0(x)\phi_j(x)\,dx, \quad j \in \{1,\dots,N\}. \end{cases} \tag{3}$$

Writing $y_h(x,t) = \sum_{j=1}^N y_j(t)\phi_j(x)$ and $y_h^0(x) = \sum_{j=1}^N y_j^0 \phi_j(x)$,problem (3) can be written as

$$\begin{cases} E_h \partial_t \vec{y}_h + R_h \vec{y}_h = \vec{f}_h, \quad t \in (0,T), \\ \vec{y}_h(0) = \vec{y}_h^0, \end{cases} \tag{4}$$

where $\vec{y}_h(t) = (y_1(t),\dots,y_N(t))$ and $\vec{f}_h$ is the vector with coefficients $(f,\phi_j)_{L^2}$. Here E_h, $R_h \in \mathbb{R}^{N\times N}$ are the so-called mass and stiffness matrices, respectively.

Since E_h is invertible, we can multiply the system

$$\begin{cases} E_h \partial_t \vec{y}_h + R_h \vec{y}_h = \vec{f}_h, & t \in (0,T), \\ \vec{y}_h(0) = \vec{y}_h^0, \end{cases}$$

by $B_h := E_h^{-1}$. We obtain

$$\begin{cases} \partial_t \vec{y}_h + A_h \vec{y}_h = B_h \vec{f}_h, & t \in (0,T), \\ \vec{y}_h(0) = \vec{y}_h^0, \end{cases} \tag{5}$$

where $A_h = E_h^{-1} R_h$.

FEM discretization

Since E_h is invertible, we can multiply the system

$$\begin{cases} E_h \partial_t \vec{y}_h + R_h \vec{y}_h = \vec{f}_h, & t \in (0,T), \\ \vec{y}_h(0) = \vec{y}_h^0, \end{cases}$$

by $B_h := E_h^{-1}$. We obtain

$$\begin{cases} \partial_t \vec{y}_h + A_h \vec{y}_h = B_h \vec{f}_h, & t \in (0,T), \\ \vec{y}_h(0) = \vec{y}_h^0, \end{cases} \tag{5}$$

where $A_h = E_h^{-1} R_h$. Using a random system, we aim to approximate the system (5).

FEM discretization

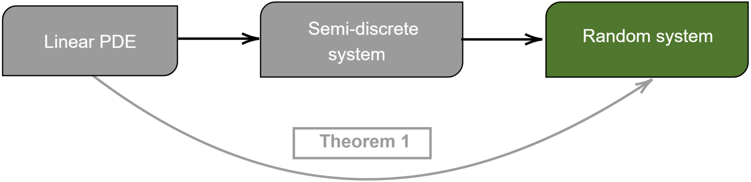
Since E_h is invertible, we can multiply the system

$$\begin{cases} E_h \partial_t \vec{y}_h + R_h \vec{y}_h = \vec{f}_h, & t \in (0,T), \\ \vec{y}_h(0) = \vec{y}_h^0, \end{cases}$$

by $B_h := E_h^{-1}$. We obtain

$$\begin{cases} \partial_t \vec{y}_h + A_h \vec{y}_h = B_h \vec{f}_h, & t \in (0,T), \\ \vec{y}_h(0) = \vec{y}_h^0, \end{cases} \tag{5}$$

where $A_h = E_h^{-1} R_h$. Using a random system, we aim to approximate the system (5).



Random batch method

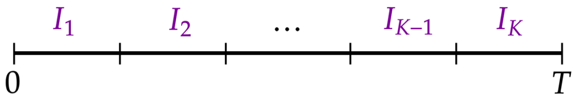
We need three ingredients:

- Let $K \in \mathbb{N}$. Consider $\delta_t = T/(K+1)$ and $I_k = [\delta_t(k-1), \delta_t k]$ for every $k \in \{1, \ldots, K\}$.

Random batch method

We need three ingredients:

- Let $K \in \mathbb{N}$. Consider $\delta_t = T/(K+1)$ and $I_k = [\delta_t(k-1), \delta_t k]$ for every $k \in \{1, \ldots, K\}$.



Random batch method

We need three ingredients

- Let $K \in \mathbb{N}$. Consider $\delta_t = T/(K+1)$ and $I_k = [\delta_t(k-1), \delta_t k]$ for every $k \in \{1, \ldots, K\}$.
- Let $M \in \mathbb{N}$. Consider $S_i \in \mathcal{P}(\{1, \ldots, M\})$ for $i \in \{1, \ldots, 2^M\}$.

Random batch method

We need three ingredients

- Let $K \in \mathbb{N}$. Consider $\delta_t = T/(K+1)$ and $I_k = [\delta_t(k-1), \delta_t k]$ for every $k \in \{1, \dots, K\}$.
- Let $M \in \mathbb{N}$. Consider $S_i \in \mathcal{P}(\{1, \dots, M\})$ for $i \in \{1, \dots, 2^M\}$.
- Consider the family of i.i.d. random variables $\boldsymbol{\omega} = \{\omega_k\}_{k=1}^K$ with values in $\{1, \dots, 2^M\}$.

Random batch method

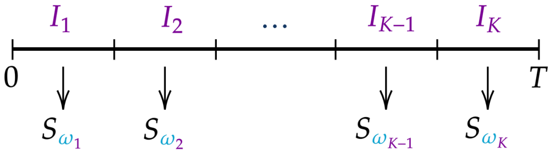
We need three ingredients

- Let $K \in \mathbb{N}$. Consider $\delta_t = T/(K+1)$ and $I_k = [\delta_t(k-1), \delta_t k]$ for every $k \in \{1, \ldots, K\}$.
- Let $M \in \mathbb{N}$. Consider $S_i \in \mathcal{P}(\{1, \ldots, M\})$ for $i \in \{1, \ldots, 2^M\}$.
- Consider the family of i.i.d. random variables $\boldsymbol{\omega} = \{\omega_k\}_{k=1}^K$ with values in $\{1, \ldots, 2^M\}$. Denote by $\mathbb{P}[\omega_k = i] = p_i$ for $i \in \{1, \ldots, 2^M\}$ and $k \in \{1, \ldots, K\}$.

Random batch method

We need three ingredients

- Let $K \in \mathbb{N}$. Consider $\delta_t = T/(K+1)$ and $I_k = [\delta_t(k-1), \delta_t k]$ for every $k \in \{1, \ldots, K\}$.
- Let $M \in \mathbb{N}$. Consider $S_i \in \mathcal{P}(\{1, \ldots, M\})$ for $i \in \{1, \ldots, 2^M\}$.
- Consider the family of i.i.d. random variables $\boldsymbol{\omega} = \{\omega_k\}_{k=1}^K$ with values in $\{1, \ldots, 2^M\}$. Denote by $\mathbb{P}[\omega_k = i] = p_i$ for $i \in \{1, \ldots, 2^M\}$ and $k \in \{1, \ldots, K\}$.



Random system

We consider the matrices $\{A_m^h\}_{m=1}^M \subset \mathbb{R}^{N\times N}$ such that

$$A_h = \sum_{m=1}^{M} A_m^h,$$

Random system

We consider the matrices $\{A_m^h\}_{m=1}^M \subset \mathbb{R}^{N\times N}$ such that

$$A_h = \sum_{m=1}^{M} A_m^h,$$

Thus, we introduce the random time-dependent matrix

$$A_R^h(\omega, t) = \sum_{m \in S_{\omega_k}} \frac{A_m^h}{\pi_m}, \quad t \in I_k.$$

Random system

We consider the matrices $\{A_m^h\}_{m=1}^M \subset \mathbb{R}^{N\times N}$ such that

$$A_h = \sum_{m=1}^{M} A_m^h,$$

Thus, we introduce the random time-dependent matrix

$$A_R^h(\omega, t) = \sum_{m \in S_{\omega_k}} \frac{A_m^h}{\pi_m}, \quad t \in I_k.$$

where π_m is a normalization constant given by

$$\pi_m := \sum_{i \in \{j \in \{1,\dots,2^M\} \,:\, m \in S_j\}} p_i.$$

Random system

We consider the matrices $\{A_m^h\}_{m=1}^M \subset \mathbb{R}^{N\times N}$ such that

$$A_h = \sum_{m=1}^{M} A_m^h,$$

Thus, we introduce the random time-dependent matrix

$$A_R^h(\omega, t) = \sum_{m\in S_{\omega_k}} \frac{A_m^h}{\pi_m}, \quad t \in I_k.$$

where π_m is a normalization constant given by

$$\pi_m := \sum_{i\in\{j\in\{1,\dots,2^M\}\,:\,m\in S_j\}} p_i.$$

In particular, this construction ensures that $\mathbb{E}[A_R^h(t)] = A_h$ for each $t \in (0, T)$.

Random system

Consider the random dynamical system

$$\begin{cases} \partial_t \vec{y}_R(\omega, t) + A_R^h(\omega, t)\vec{y}_R(\omega, t) = B\vec{f}, \\ \vec{y}_R(0) = \vec{y}_h^0. \end{cases} \tag{6}$$

Random system

Consider the random dynamical system

$$\begin{cases} \partial_t \vec{y}_R(\omega, t) + A_R^h(\omega, t)\vec{y}_R(\omega, t) = B\vec{f}, \\ \vec{y}_R(0) = \vec{y}_h^0. \end{cases} \tag{6}$$

In the following, we write

$$y_R(x, t) = \sum_{j=1}^{N} (\vec{y}_R)_j \phi_j(x) \tag{7}$$

Theorem (1)

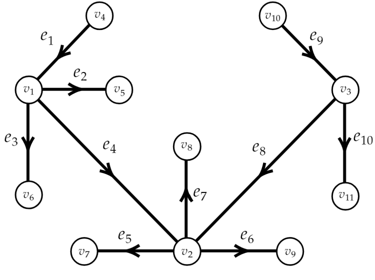
Let y_R be as in (7) and let y be the solution of the heat equation with initial condition $y^0 \in H_0^1(0, L)$ and $f \in L^2(0, T; H_0^1(0, L))$. Then,

$$\mathbb{E}[\|y_R(\cdot, t) - y(\cdot, t)\|_{L^2(0,L)}^2] \leq C\left(h^4 + \frac{\delta_t}{h^7} C(M, \omega)\right), \tag{8}$$

for every $t \in (0, T)$, where $C > 0$ and $C(M, \omega) > 0$ are independent of h and δ_t, with $C(M, \omega)$ depending on the chosen decomposition.

Application to PDEs on networks

Let us consider a graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, bwhere $\mathcal{V}_0$ denotes its interior nodes, and $\mathcal{V}_b$ its boundary nodes.



$$\mathcal{V}_0 = \{v_1, v_2, v_3\}$$
$$\mathcal{V}_b = \{v_4, v_5, \ldots, v_{11}\}$$
$$\mathcal{E} = \{e_1, \ldots, e_{10}\}$$
$$\mathcal{E}(v_1) = \{e_1, e_2, e_3\}.$$

Heat equation on the network

On each e_i with $i \in \{1, \dots, 10\}$ we consider the heat equation

$$\begin{cases} \partial_t y^{e_i} - \partial_{xx} y^{e_i} = f^{e_i}, & (x,t) \in (0,L) \times (0,T), \\ y^{e_i}(x,0) = y^0_{e_i}(x), & x \in (0,L). \end{cases} \tag{9}$$

System (9) is complemented with boundary and coupling conditions

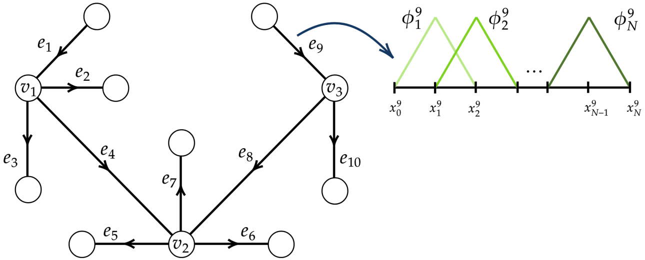
$$\begin{cases} y^e(v,t) = 0, & v \in \mathcal{V}_b,\ e \in \mathcal{E}(v), \\ y^{e_1}(v,t) = y^{e_2}(v,t), & v \in \mathcal{V}_0,\ e_1, e_2 \in \mathcal{E}(v), \\ \displaystyle\sum_{e \in \mathcal{E}(v)} \partial_x y^e(v,t) n_e(v) = 0, & v \in \mathcal{V}_0, \end{cases} \tag{10}$$

over the time interval $(0,T)$.

Let us introduce a semi-discrete system.

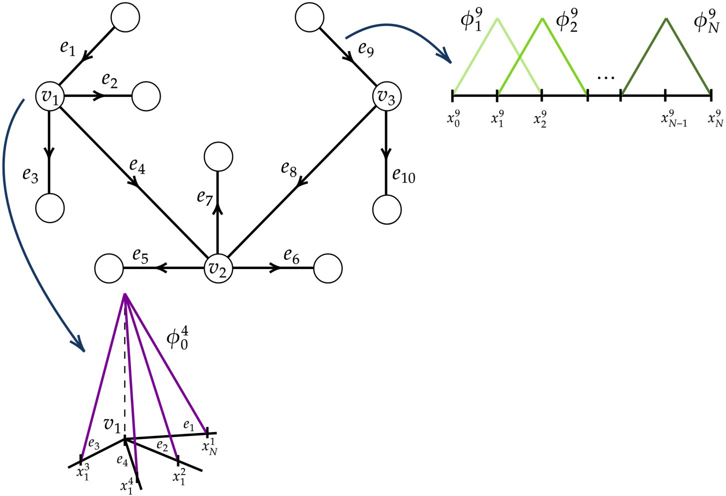
FEM discretization

On each e_i we define the basis functions $\{\phi_j^i\}_{j=1}^N$.



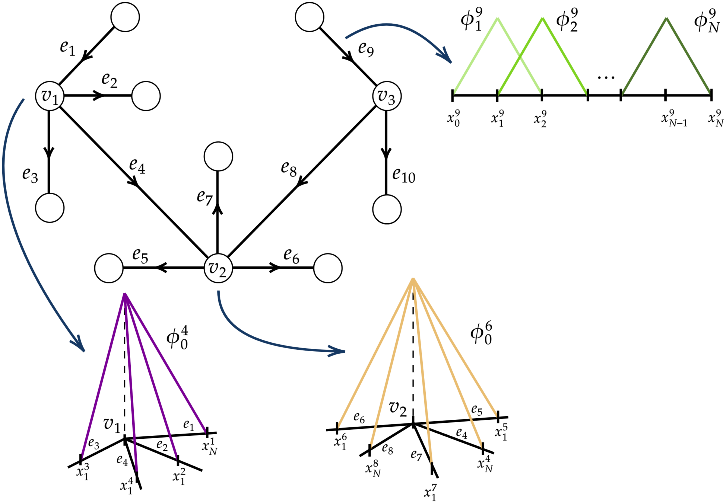
What about the interior vertices?

On each e_i we define the basis functions $\{\phi_j\}_{j=1}^N$.



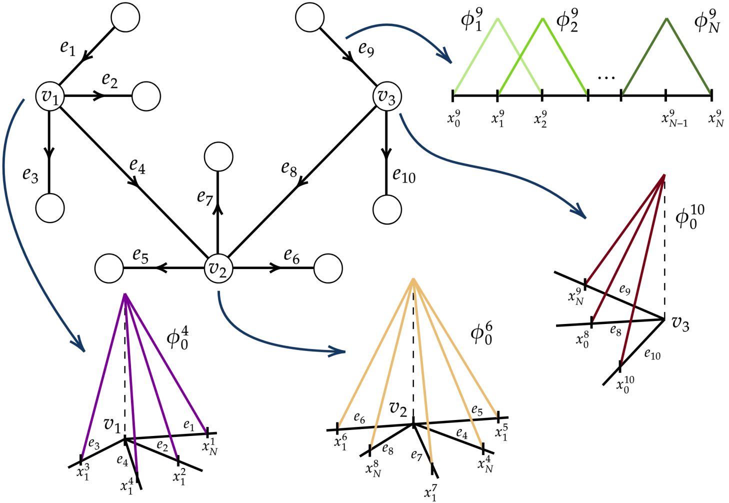
FEM discretization

On each e_i we define the basis functions $\{\phi_j\}_{j=1}^{N}$.



FEM discretization

On each e_i we define the basis functions $\{\phi_j\}_{j=1}^N$.



Semi-discrete equation

We can introduce the space $V_h^{\mathcal{E}}$ as the span of the functions $\left(\bigcup_{i=1}^{10}\bigcup_{j=1}^{N}\{\phi_j^i\}\right) \cup \{\phi_0^4, \phi_0^6, \phi_0^{10}\}$.

Semi-discrete equation

We can introduce the space $V_h^{\mathcal{E}}$ as the span of the functions $\left(\bigcup_{i=1}^{10} \bigcup_{j=1}^{N} \{\phi_j^i\}\right) \cup \{\phi_0^4, \phi_0^6, \phi_0^{10}\}$. Find $y_h \in C^1(0, T; V_h^{\mathcal{E}})$ such that

$$\begin{cases} \displaystyle\sum_{i=1}^{10} \left(\int_0^L \partial_t y_h^{e_i}(x,t)\phi_j^i(x) + \partial_x y_h^{e_i}(x,t)\partial_x \phi_j^i(x)\, dx \right) = \sum_{i=1}^{10} \int_0^L f^{e_i}(x,t)\phi_j^i(x)\, dx \\ \displaystyle\int_0^L y_h^{e_i}(x,0)\phi_j^i(x) = \int_0^L y_{h,e_i}^0(x)\phi_j^i(x)\, dx, \end{cases}$$

We can introduce the space $V_h^{\mathcal{E}}$ as the span of the functions $\left(\bigcup_{i=1}^{10}\bigcup_{j=1}^{N}\{\phi_j^i\}\right) \cup \{\phi_0^4, \phi_0^6, \phi_0^{10}\}$. Find $y_h \in C^1(0,T;V_h^{\mathcal{E}})$ such that

$$\begin{cases} \displaystyle\sum_{i=1}^{10}\left(\int_0^L \partial_t y_h^{e_i}(x,t)\phi_j^i(x) + \partial_x y_h^{e_i}(x,t)\partial_x \phi_j^i(x)\,dx\right) = \sum_{i=1}^{10}\int_0^L f^{e_i}(x,t)\phi_j^i(x)\,dx \\ \displaystyle\int_0^L y_h^{e_i}(x,0)\phi_j^i(x) = \int_0^L y_{h,e_i}^0(x)\phi_j^i(x)\,dx, \end{cases}$$

Since $y_h^{e_i}$ can be written as $y_h^{e_i}(x,t) = \sum_{j=1}^{N} y_j^{e_i}(t)\phi_j^i(x)$, we consider the vector of coefficients $\boldsymbol{Y}_h(t) = \left(\vec{y}_h^{e_1}(t), \dots, \vec{y}_h^{e_{10}}(t)\right)^\top$. Then,

$$\begin{cases} \boldsymbol{E}_h\partial_t\boldsymbol{Y}_h + \boldsymbol{R}_h\boldsymbol{Y}_h = \boldsymbol{F}_h, \quad t \in (0,1), \\ \boldsymbol{Y}_h(0) = \boldsymbol{Y}_h^0. \end{cases} \tag{11}$$

Semi-discrete equation

Here we have that

$$R_h = \begin{pmatrix} R_h \; R_h \; R_h & c_1^4 & 0 & 0 & 0 & 0 & 0 \\ a_1^4 \; a_2^4 \; a_3^4 & 4/h & a_4^4 \; 0 & 0 & 0 & 0 & 0 \\ 0 & c_2^4 & R_h \; R_h & c_1^6 & 0 & 0 & 0 \\ 0 & 0 & a_1^2 \; a_2^2 & 5/h & a_6^6 \; a_7^6 \; a_8^6 \; 0 & 0 & 0 \\ 0 & 0 & 0 & c_2^6 & R_h \; R_h \; R_h \; R_h & c_1^{10} & 0 \\ 0 & 0 & 0 & 0 & 0 \; 0 \; a_8^{10} \; a_9^{10} & 3/h & a_{10}^{10} \\ 0 & 0 & 0 & 0 & 0 & c_2^{10} & R_h \end{pmatrix}$$

where $R_h \in \mathbb{R}^{N \times N}$ is the stiffness matrix of the $1-d$ heat equation.

Semi-discrete equation

Here we have that

$$
\begin{array}{c}
\left[\; e_1 \;\; e_2 \;\; e_3 \qquad v_1 \qquad e_4 \;\; e_5 \qquad v_2 \qquad e_6 \;\; e_7 \;\; e_8 \;\; e_9 \qquad v_{10} \qquad e_{10} \;\right]
\end{array}
$$

$$
R_h = \begin{pmatrix}
R_h \;\; R_h \;\; R_h & c_1^4 & 0 & 0 & 0 & 0 & 0 \\
a_1^4 \;\; a_2^4 \;\; a_3^4 & 4/h & a_4^4 \;\; 0 & 0 & 0 & 0 & 0 \\
0 & c_2^4 & R_h \;\; R_h & c_1^6 & 0 & 0 & 0 \\
0 & 0 & a_1^2 \;\; a_2^2 & 5/h & a_6^6 \;\; a_7^6 \;\; a_8^6 \;\; 0 & 0 & 0 \\
0 & 0 & 0 & c_2^6 & R_h \;\; R_h \;\; R_h \;\; R_h & c_1^{10} & 0 \\
0 & 0 & 0 & 0 & 0 \;\; 0 \;\; a_8^{10} \;\; a_9^{10} & 3/h & a_{10}^{10} \\
0 & 0 & 0 & 0 & 0 & c_2^{10} & R_h
\end{pmatrix}
$$

where $R_h \in \mathbb{R}^{N\times N}$ is the stiffness matrix of the $1-d$ heat equation.

Semi-discrete equation

Here we have that

$$
\begin{array}{ccccccccccccc}
[\,e_1 & e_2 & e_3 & v_1 & e_4 & e_5 & v_2 & e_6 & e_7 & e_8 & e_9 & v_{10} & e_{10}\,]
\end{array}
$$

$$
R_h = \begin{pmatrix}
R_h & R_h & R_h & c_1^4 & 0 & & 0 & & & 0 & & 0 & 0 \\
a_1^4 & a_2^4 & a_3^4 & 4/h & a_4^4 & 0 & 0 & & & 0 & & 0 & 0 \\
 & 0 & & c_2^4 & R_h & R_h & c_1^6 & & & 0 & & 0 & 0 \\
 & 0 & & 0 & a_1^2 & a_2^2 & 5/h & a_6^6 & a_7^6 & a_8^6 & 0 & 0 & 0 \\
 & 0 & & 0 & 0 & & c_2^6 & R_h & R_h & R_h & R_h & c_1^{10} & 0 \\
 & 0 & & 0 & 0 & & 0 & 0 & 0 & a_8^{10} & a_9^{10} & 3/h & a_{10}^{10} \\
 & 0 & & 0 & 0 & & 0 & & & 0 & & c_2^{10} & R_h
\end{pmatrix}
$$

where $R_h \in \mathbb{R}^{N\times N}$ is the stiffness matrix of the $1-d$ heat equation.

Here we have that

$$
\begin{array}{ccccccccccccc}
[\,e_1 & e_2 & e_3 & v_1 & e_4 & e_5 & v_2 & e_6 & e_7 & e_8 & e_9 & v_{10} & e_{10}\,]
\end{array}
$$

$$
R_h = \begin{pmatrix}
R_h & R_h & R_h & c_1^4 & 0 & & 0 & & & 0 & & 0 & 0 \\
a_1^4 & a_2^4 & a_3^4 & 4/h & a_4^4 & 0 & 0 & & & 0 & & 0 & 0 \\
& 0 & & c_2^4 & R_h & R_h & c_1^6 & & & 0 & & 0 & 0 \\
& 0 & & 0 & a_1^2 & a_2^2 & 5/h & a_6^6 & a_7^6 & a_8^6 & 0 & 0 & 0 \\
& 0 & & 0 & 0 & & c_2^6 & R_h & R_h & R_h & R_h & c_1^{10} & 0 \\
& 0 & & 0 & 0 & & 0 & 0 & 0 & a_8^{10} & a_9^{10} & 3/h & a_{10}^{10} \\
& 0 & & 0 & 0 & & 0 & & & 0 & & c_2^{10} & R_h
\end{pmatrix}
$$

where $R_h \in \mathbb{R}^{N \times N}$ is the stiffness matrix of the $1-d$ heat equation.

Matrix $\boldsymbol{E}_h$ has a similar structure.

Here we have that

$$[\, e_1 \;\; e_2 \;\; e_3 \;\; v_1 \;\; e_4 \;\; e_5 \;\; v_2 \;\; e_6 \;\; e_7 \;\; e_8 \;\; e_9 \;\; v_{10} \;\; e_{10} \,]$$

$$R_h = \begin{pmatrix} R_h \;\; R_h \;\; R_h & c_1^4 & 0 & 0 & 0 & 0 & 0 \\ a_1^4 \;\; a_2^4 \;\; a_3^4 & 4/h & a_4^4 \;\; 0 & 0 & 0 & 0 & 0 \\ 0 & c_2^4 & R_h \;\; R_h & c_1^6 & 0 & 0 & 0 \\ 0 & 0 & a_1^2 \;\; a_2^2 & 5/h & a_6^6 \;\; a_7^6 \;\; a_8^6 \;\; 0 & 0 & 0 \\ 0 & 0 & 0 & c_2^6 & R_h \;\; R_h \;\; R_h \;\; R_h & c_1^{10} & 0 \\ 0 & 0 & 0 & 0 & 0 \;\; 0 \;\; a_8^{10} \;\; a_9^{10} & 3/h & a_{10}^{10} \\ 0 & 0 & 0 & 0 & 0 & c_2^{10} & R_h \end{pmatrix}$$

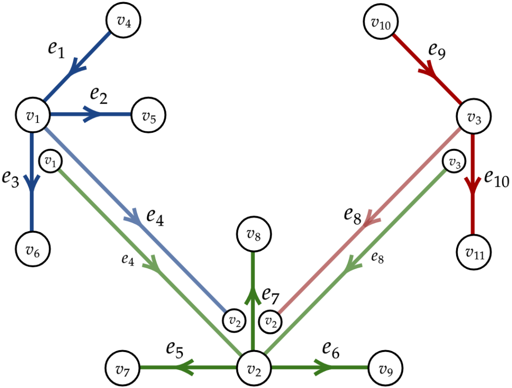
where $R_h \in \mathbb{R}^{N \times N}$ is the stiffness matrix of the $1-d$ heat equation.

Matrix $\boldsymbol{E}_h$ has a similar structure. Observe that

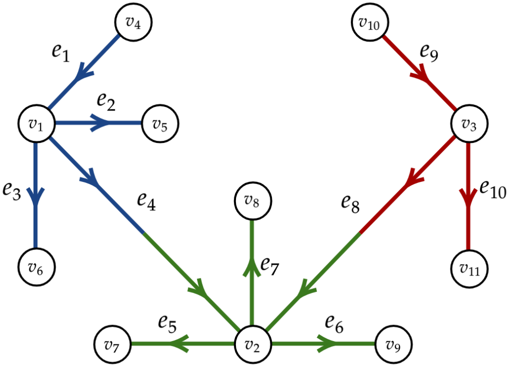
$$\boldsymbol{R}_h = \begin{pmatrix} B_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ 0 & B_2 & 0 \\ 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & B_3 \end{pmatrix} = \boldsymbol{R}_1 + \boldsymbol{R}_2 + \boldsymbol{R}_3$$

A different decomposition of $\boldsymbol{R}_h$ induces a different domain decomposition.

1) Overlapping decomposition:



2) Non-overlapping decomposition: A different election of B_1, B_2 and B_3 can introduce a non-overlapping decomposition.



Numerical implementations

We define

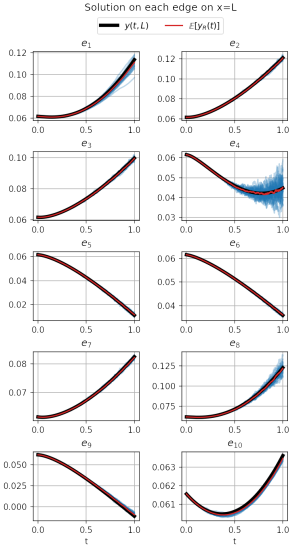
$$\begin{aligned} \boldsymbol{A}_h &= (\boldsymbol{E}_h)^{-1}\boldsymbol{R}_h \\ &= (\boldsymbol{E}_h)^{-1}\boldsymbol{R}_1 + (\boldsymbol{E}_h)^{-1}\boldsymbol{R}_2 + (\boldsymbol{E}_h)^{-1}\boldsymbol{R}_3 \\ &= \boldsymbol{A}_1^h + \boldsymbol{A}_2^h + \boldsymbol{A}_3^h \end{aligned}$$

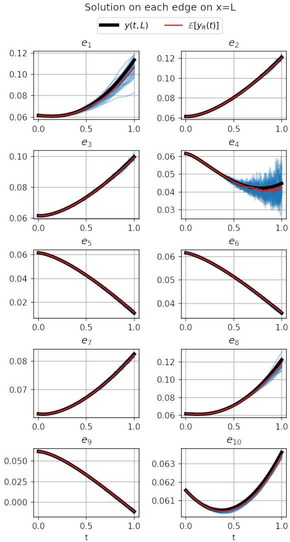
With the previous decomposition we can define a random matrix $\boldsymbol{A}_R^h(\boldsymbol{\omega})$. Then, the random system associated is given by

$$\begin{cases} \partial_t \boldsymbol{Y}_R(\boldsymbol{\omega}, t) + \boldsymbol{A}_R^h(\boldsymbol{\omega}, t)\boldsymbol{Y}_R(\boldsymbol{\omega}, t) = \boldsymbol{F}, \\ \boldsymbol{Y}_R(0) = \boldsymbol{Y}_h^0. \end{cases} \tag{12}$$

Numerical results

Thus, comparing the numerical solution of the heat and random equations, we obtain (left overlapping, right non-overlapping).





Comments and next steps

1. In this work, we have extend this results to the linear quadratic regular for the heat equation.

Comments and next steps

1. In this work, we have extend this results to the linear quadratic regular for the heat equation.

2. Work with A. Dominguez-Corella "Mini-batch descent in semiflows". Application of RBM to gradient flows.

Comments and next steps

1. In this work, we have extend this results to the linear quadratic regular for the heat equation.

2. Work with A. Dominguez-Corella "Mini-batch descent in semiflows". Application of RBM to gradient flows.

3. We are working on introducing a "random" operator splitting scheme of parabolic operators.

Thanks for your attention.

References

- A. Dominguez-corella, M. Hernandez
  Mini-batch descent in semiflows , 2024
- D. Veldman, E. Zuazua
  A framework for randomized time-splitting in linear-quadratic optimal control, 2022
- M. Eisenmann, T. Stillfjord.
  A randomized operator splitting scheme inspired by stochastic optimization methods, 2022
- D. Ko, E. Zuazua
  Model predictive control with random batch methods for a guiding problem. 2020
- D. Veldman, A. Borkowski, E. Zuazua
  Stability and Convergence of a Randomized Model Predictive Control Strategy, 2022

Main theorems used to prove Theorem 1

The proof is based on two crucial theorems.

Theorem 2 (FEM error): Let us assume that $y^0 \in H_0^1(0, L)$ and $f \in L^2(0, T; L^2(0, L))$. Denote by y the solution of the heat equation and y_h the solution of the semi-discrete system. Then, there exists a constant $C > 0$, independent of $h > 0$, such that

$$\|y(t) - y_h(t)\|_{L^2(0,L)}^2 \leq Ch^4(\|y_0\|_{H_0^1(0,L)}^2 + \|f\|_{L^2(0,T;L^2(0,L))}^2), \qquad (13)$$

for every $t \in (0, T)$

Theorem 3 (RBM for ODEs): Denote by $\vec{y}_h$ and $\vec{y}_R$ the solutions of the semi-discrete and random equations, respectively. Then, for every $t \in (0, T)$, we have

$$\mathbb{E}[\|\vec{y}_h(t) - \vec{y}_R(t)\|^2] \leq (\|A_h\|T^2 + 2T)(\|\vec{y}_h^0\| + \|B_h \vec{f}_h\|_{L^1(0,T;\mathbb{R}^N)})^2 \mathrm{Var}[A_R]\Delta t,$$

where

$$\mathrm{Var}[A_R] := \sum_{i=1}^{2^M} \left(\left\| A_h - \sum_{m \in S_i} \frac{A_m^h}{\pi_m} \right\|^2 p_i \right).$$