Space-time domain decomposition of optimal control problems for the wave equation on graphs: An augmented Lagrangian approach

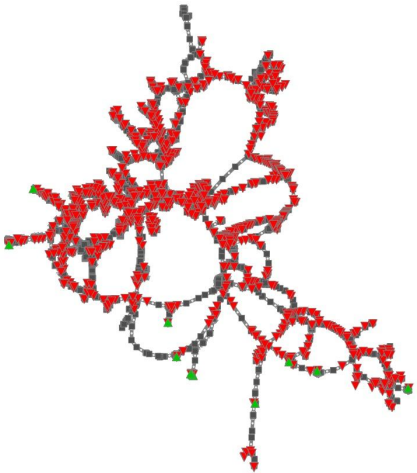
Günter Leugering, FAU Erlangen-Nürnberg[1]

The Mathematics of Scientific Machine Learning and Digital Twins

Erice, November 20–24, 2025

[1] Also: Scientific Director of the Speinshart Science Center for AI and SuperTech (SSC)

Desciption of the overall problem by the example of gas networks



GasLib4197 Gas network (with control instruments) see TRR 154 DFG

We consider the problem of optimal control on a pipe network $G = (V, E)$. Originally, one employs the Euler system on each edge. For subsonic flow, one can derive a wave equation

$$u^e_{tt} - (a_e u^e_x)_x) = f_e(u^e, u^e_t)$$

and for friction domainted flow a doubly nonlinear p-parabolic equation $(\beta(s) = |s|^{p-2}s,\ p = \frac{3}{2}$,

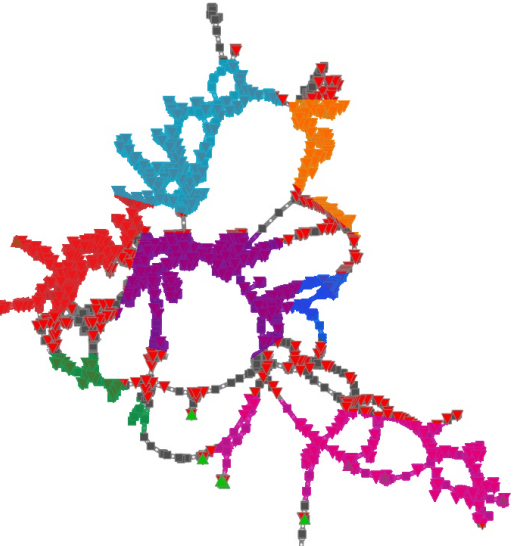
$$\beta(u^e)_t - (\beta(u^e_x))_x = 0$$

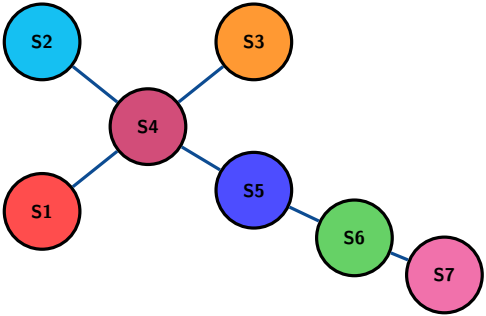
for each edge with tansmission conditions at the joints:

$$u^e(v,t) = u^{e'}(v,t),\ e, e' \in E(v)$$

$$\sum_{e \in E(v)} \epsilon_{e,v} a_e u^e_x(v,t) = 0,$$

Subgraph decomposition and coupling meta-graph

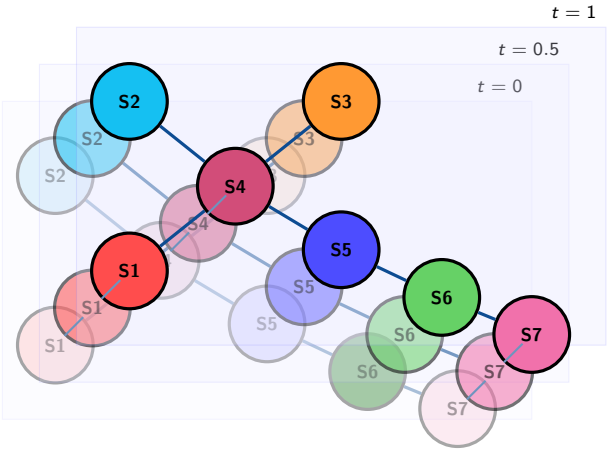




Coupling meta-graph $\mathcal{T}$: each fat node corresponds to a colored subgraph on the left.

Open, but... we may want to collapse a subgraph to a point!

Motivation: formal space-time domain decomposition



models may change between time caps

Formal decomposition

$$G = \bigcup_{p=1}^{P} G^{(p)}, \qquad G^{(p)} = (E^{(p)}, V^{(p)}),$$

$$E^{(p)} \cap E^{(q)} = \emptyset \ (p \neq q),$$

decompose time interval into windows $0 = T_0 < T_1 < \cdots < T_M = T$. For each (p, m) we define the space–time slab and its spatial ports

$$\Omega_m^{(p)} := \bigcup_{e \in E^{(p)}} (0, L_e) \times (T_{m-1}, T_m),$$

The cut Γ defines subdomains for DDM and virtual control coupling.

Wave equation on graphs

On edge e with density ρ_e and stiffness a_e:

$$
\begin{aligned}
&\rho_e u^e_{tt} - (a_e u^e_x)_x = f_e \quad \text{in } (0,\ell_e)\times(0,T),\\
&u^e(v,t) = u^{e'}(v,t),\ e,e' \in \mathcal{E}(v)\\
&\sum_{e\in\mathcal{E}(v)} \sigma_{ve}\, a_e\, u^e_x(v,t) = g_v(t),\ v\in V,\ d(v)\geq 2,\ \sigma_{ve}\in\{\pm 1\}\\
&a_e n_e u^e_x(v,t) = w^e_v(t)\\
&u^e(\cdot,0) = u^e_0,\ u^e_t(\cdot,0) = u^e_1.
\end{aligned}
\tag{1}
$$

Energy $E(t) = \frac{1}{2}\sum_e \int_0^{\ell_e} \left(\rho_e|\partial_t u_e|^2 + a_e|\partial_x u_e|^2\right)\mathrm{d}x$.
For suitable data, $u \in C([0,T];\mathcal{H}^1(\mathcal{G}))\cap C^1([0,T];L^2(\mathcal{G}))$; control at $\mathcal{V}_B$ enters via Dirichlet/Neumann data.

Energy and well-posedness (formal)

We consider the tracking type cost function

$$J(u,w) = \frac{1}{2}\int_0^T \sum_{e\in E} \int_0^{L_e} (u^e - u_d^e)^2 \, dx \, dt \; + \; \frac{\gamma}{2}\int_0^T \sum_{v\in V_b} |w_v(t)|^2 \, dt. \tag{2}$$

Clearly, we can also treat final value tracking and more.
We have the following optimal control problem

$$\inf_{u,w} J(u,w), \qquad \text{s.t. } u \text{ satisfies (1)} \tag{3}$$

Notice: non-overlapping domain decomposition for optimal control problems (e.g. Helmholtz) have been introduced by Benamou/Després 1996 in the sense of P.L. Lions 1989.

Graph decomposition

The global KKT-system reads

$$
\begin{aligned}
&\rho_e u^e_{tt} - (a_e u^e_x)_x = 0,\\
&\rho_e p^e_{tt} - (a_e p^e_x)_x = -(u^e - u^e_d),\\
&u^e(v) = u^{e'}(v),\ p^e(v) = p^{e'}(v),\ e, e' \in E(v)\\
&\sum_{e\in E(v)} a_e n_{e,v} u^e_x(v) = 0,\ \sum_{e\in E(v)} a_e n_{e,v} u^e_x(v) = 0 \quad (v \in V_0),\\
&a_e n_{e,v} u^e_x(v) = w_v = \gamma^{-1} a_e n_{e,v} p^e(v),\ p^e_x(v) = 0 \quad (v \in V_b)\\
&u^e(0,\cdot) = u^e_0,\ u^e_t(0,\cdot) = u^e_1,\\
&p^e(T,\cdot) = 0,\ p^e_t(T,\cdot) = 0.
\end{aligned}
\tag{4}
$$

Our aim is to decompose (4) in space and time by non-overlapping domain decomposition mehtods.

Local KKT-system and virtual controls

To this end, we introduce *virtual controls* both at the interface nodes and the time caps.

$$
\begin{aligned}
&\rho_e u_{tt}^{e,(m,p)} - (a_e u_x^{e,(m,p)})_x = 0, \quad \text{in } G^{(p)} \times I_m \\
&u^{e,(m,p)}(v,t) = u^{e',(m,p)}(v,t),\ e,e' \in E^{(p)},\ v \in V^{(p)} \setminus n, \\
&\sum_{e \in E^{(p)}} a_e n_{e,v} u_x^{e,(m,p)}(v,t) = 0,\ t \in I_m \\
&a_e n_{e,n} u_x^{e,(m,p)}(n,t) = g_{n,e}^{(p)}(t),\ n \in \mathcal{N}^{(p)},\ t \in I_m \\
&a_e n_{e,v} u_x^{e,(m,p)}(v,t) = w_v(t),\ v \in V_b \cap V^{(p)},\ t \in I_m \\
&u^{e,(m,p)}(\cdot, T_{m-1}) = h_{m,m-1}^{e,(p)},\ u_t^{e,(m,p)}(T_{m-1}) = g_{m,m-1}^{e,(p)}, \quad \text{in } (0, L_e).
\end{aligned}
\tag{5}
$$

Notice that at the *interior nodes* (*port nodes*) of $G^{(p)}$, the classical transmission conditions hold. The virtual controls (in red) should be chosen to satisfy the transmisson conditions at the *connecting nodes* of the meta-graph.

The augmented Lagrangian

On each slab $\Omega_m^{(p)}$ with adjoint p^e, define two components of the overall Lagrange function.

$$\begin{aligned}\mathcal{L}_{m;1}^{(p)} =& \frac{1}{2}\int_{T_{m-1}}^{T_m}\sum_{e\in E^{(p)}}\int_0^{L_e}(u^e-u_d^e)^2\,dx\,dt+\frac{\gamma}{2}\int_{T_{m-1}}^{T_m}\sum_{n\in\mathcal{N}^{(p)}\cap V_b}|w_n|^2dt\\ &+\sum_{e\in E^{(p)}}\int_{T_{m-1}}^{T_m}\int_0^{L_e}(u_{tt}^e-(a_e u_x^e)_x)\,p^e\,dx\,dt\\ &\frac{1}{2\beta}\sum_{e\in E^{(p)}}\int_{\Gamma_m^{(p)}}\left[|u^e(T_{m-1})-\xi_{m,m+1}^{e,(p)}|^2+|h_{m,m-1}^{e,(p)}-\xi_{m,m-1}^{e,(p)}|^2\right]d\mu\\ &+\frac{1}{2\beta}\sum_{e\in E^{(p)}}\int_{\Gamma_m^{(p)}}\left[|u_t^e(T_m)-\mu_{m,m+1}^{e,(p)}|^2+|g_{m,m-1}^{e,(p)}-\mu_{m,m-1}^{e,(p)}|^2\right]d\mu.\end{aligned}$$

(green=data). The first part represents the classical Lagrange function for the *physical* control problem on edge e, the second set of sums realize the tracking at the time caps.

Augmented Lagrangian: interface at port nodes

The coupling at the *port nodes* (where two or more subgaphs meet $G^{(p)}$) is realized via an augmented Lagrange ansatz:

$$\begin{aligned}\mathcal{L}^{(p)}_{m;2} = &\sum_n \sum_{e\in\cup_r E^{(r)}(n)} \int_{T_{m-1}}^{T_m} \Big[\lambda^{(m)}_{n,e}(u^e(n) - q^{(m)}_n) + \tfrac{\rho}{2}(u^e(n) - q^{(m)}_n)^2\Big]\,dt \\ &+ \sum_n \sum_{e\in\cup_r E^{(r)}(n)} \int_{T_{m-1}}^{T_m} \Big[\eta^{(m)}_{n,e}(g^{(m)}_{n,e} - z^{(m)}_{n,e}) + \tfrac{\sigma}{2}|g^{(m)}_{n,e} - z^{(m)}_{n,e}|^2\Big]\,dt \\ &+ \sum_n \int_{T_{m-1}}^{T_m} \Big[\nu^{(m)}_n \sum_{e\in\cup_r E^{(r)}(n)} z^{(m)}_{n,e} + \tfrac{\tau}{2}\Big(\sum_{e\in\cup_r E^{(r)}(n)} z^{(m)}_{n,e}\Big)^2\Big]\,dt\end{aligned}$$

(blue= auxiliary variables). The first part realizes the continuity of traces, the second part the block-Kirchhoff constraints.

ALG3 $\frac{1}{2}$-step Uzawa

With $\mathcal{L}_m := \mathcal{L}_{m,1} + \mathcal{L}_{m,2}$, we formulate the saddle point problem

$$\inf_{u,w,g,h_{m,m-1},g_{m,m-1},q,z} \sup_{\lambda,\rho,\tau} \mathcal{L}_m$$

using the Uzawa-type Algorithm 3 of Glowinski and LeTallec (1989), which is equivalent to the Peacment-Rachford scheme for operator splitting. In essence it is a predcitor/corrector algorithm. The short schematic description is:

$$(q^{k-1}, z^{k-1}, \mu^{k-1}, \xi^{k-1}) \xrightarrow[\text{solve}]{\text{optimality}} (u^k, p^k, g^k, h^k_{m,m-1}, g^k_{m,m-1}) \xrightarrow[\frac{1}{2}\text{-step}]{\textit{predictor}} (\lambda^{k+\frac{1}{2}}, \eta^{k+\frac{1}{2}}, \nu^{k+\frac{1}{2}})$$
$$\xrightarrow{\text{primal update}} (q^k, z^k) \xrightarrow[\textit{fullstep}]{\text{corrector}} (\lambda^{k+1}, \eta^{k+1}, \nu^{k+1}, g^{k+1}, h^k_{m,m-1}, g^k_{m,m-1}) \xrightarrow{\text{update}} (u^{k+1}, p^{k+1})$$

The P.L. Lions-type non-overlapping interface conditions

We introduce the Housholder reflection operator S_v at a node v: $S_v(w)_e = w^e - \frac{2}{d(v)} \sum_{e' \in E(v) w^{e'}} w^{e'}$.

With the *balanced* choice $\sigma = \frac{1}{\rho}$ we can derive the set of interface conditions

$$a_e n_{e,v} u_x^{e,(p,m),k+1}(v,t) - \rho p^{e,(p,m),k+1}(v,t) = S_v\Big(\rho p^{\cdot,(m,p),k}(v,t) - a.n.u_x^{\cdot,(m,p,k}(v,t)\Big)_e,$$
$$a_e n_{e,v} p_x^{e,(p,m),k+1}(v,t) + \rho u^{e,(p,m),k+1}(v,t) = -S_v\Big(\rho u^{\cdot,(m,p),k}(v,t) + a.n.p_x^{\cdot,(m,p,k}(v,t)\Big)_e$$

These coupling conditions are precisley the conditions obtained by Lagnese and GL (2000). The coupling conditions at the time interfaces (caps) read

$$\begin{aligned}
&u_t^{e,(m,p),k+1}(\cdot, T_m) - \beta p^{e,(m,p),k+1}(T_m, \cdot) = \mu_{m,m+1}^{e,(p),k} \\
&\beta p_t^{e,(m,p),k+1}(\cdot, T_m) + u^{e,(m,p),k+1}(T_m, \cdot) = \xi_{m,m+1}^{e,(p),k} \\
&u_t^{e,(m,p),k+1}(\cdot, T_{m-1}) + \beta p^{e,(m,p),k+1}(T_{m-1}, \cdot) = \mu_{m,m-1}^{e,(p),k} \\
&-\beta p_t^{e,(m,p),k+1}(\cdot, T_{m-1}) + u^{e,(m,p),k+1}(T_{m-1}, \cdot) = \xi_{m,m-1}^{e,(p),k}
\end{aligned} \tag{6}$$

Update rules

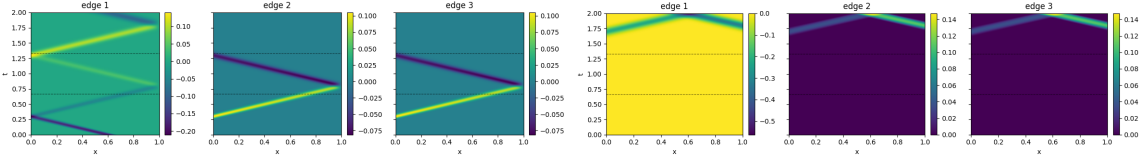
We have the following update rules for time caps coupling the time domain I_m with I_{m-1} and I_{m+1}:

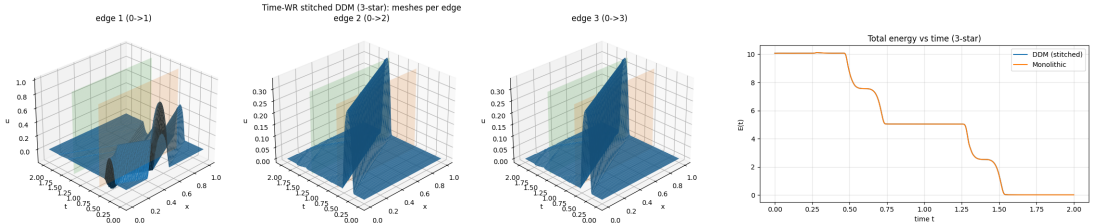
$$\begin{aligned}
\mu^{e,(p),k}_{m,m+1} &= u^{e,(m+1,p),k}_t(\cdot, T_m) - \beta p^{e,(m+1,p),k+1}(\cdot, T_m)\\
\xi^{e,(p),k}_{m,m+1} &= \beta p^{e,(m+1,p),k}_t(\cdot, T_m) + u^{e,(m+1,p),k}(\cdot, T_m)\\
\mu^{e,(p),k}_{m,m-1} &= u^{e,(m-1,p),k}_t(\cdot, T_{m-1}) + \beta p^{e,(m-1,p),k}(\cdot, T_{m-1}),\\
\xi^{e,(p),k}_{m,m-1} &= -\beta p^{e,(m-1,p),k}_t(\cdot, T_{m-1}) + u^{e,(m-1,p),k}(\cdot, T_{m-1}).
\end{aligned} \tag{7}$$

These are prcesiely the transmsission conditions provided by Lagnese and GL (2002).The Robin conditions at the interface (port) nodes together with the time cap conditions (6), (7) provide a full set of initial, final and boundary conditions for the state/adjoint system of wave-equations.

Remark: A similar procedure has been applied to parabolic (and even nonlinear p-parabolic problems)! See Gander and GL (2025)

STDDM: 3-star experiments

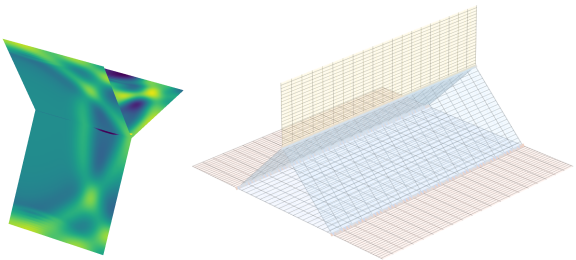




Upper figures: optimal control for the 3-star: left states / right adjoints

Lower figures: simulation with absorbing leaves: left states / right energy (e.g. finite time stabilization)

Outlook: 2-D networks (see J.E.Lagnese & G.L. 2003) for the modelling



We can extend the theory to 2-D networks in $\mathbb{R}^3$.he bilinear form becomes

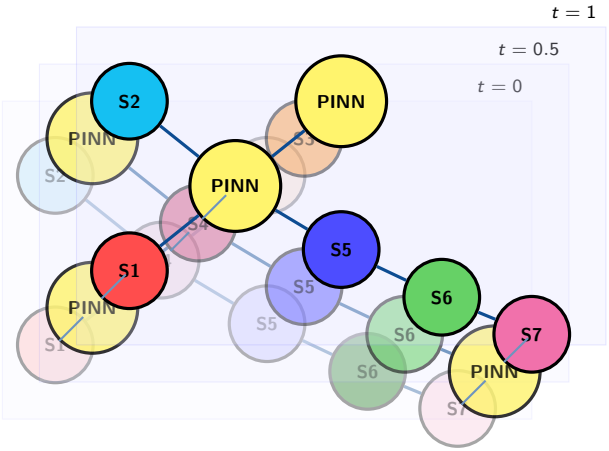
$$a(w,\phi) = \sum_{i\in\mathcal{I}} \int_{P_i} \left[A_i^{\alpha,\beta}(w_{i,\beta})(\phi_{i,\alpha}) + c_i w_i \phi_i \right]$$

The Kichhoff condition becomes:

$$\sum_{i\in\mathcal{I}_e} \nu_{i,e}(A_i \nabla w_i(\Gamma_e)) = 0$$

along with the continuity along Γ_e.
In principle, this could also extend to networks of membranes and Reißner Mindlin plates

Introducing a PINN subgraph: AL–PINN



TDD for **turnpike** handling: Zuazua/Geshkovski 2022

We now consider a sub-graph where we use a PINN-surrogate. We introduce an **augmented Lagrangian** ansatz for the loss. This implies all the benefits from the Hestenes idea to stabilize the Lagrange relaxation by a possibly small penalty term.
Notice: there is no PINN-adjoint in the calssical PINN-approach here we introduce a PINN-adjoint.

c.f. Son, Cho, Hwang 2023, (Burgers, Helmholtz, KLein-Gordon)
Basir, Zenocak 2025 (PECANN)
(1D-wave, 2D-Navier-Stokes)

AL–VPINN for the Wave Equation on a Metric Graph

Neural trial space. Approximate the solution $u_\theta(x,t)$ with a neural network defined on the graph.
Learned space–time test space. $v(x,t) = \phi_i(x)\,\psi_j(t)$, where

- $\phi_i(x)$: FEM-type basis on the metric graph (continuity $\Rightarrow$ Kirchhoff conditions built in),
- $\psi_j(t)$: learned temporal test functions.

Weak VPINN residual.

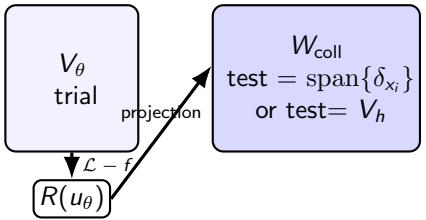
$$\mathcal{R}(u_\theta;\phi_i\psi_j) = \sum_e \int_0^{L_e} \left(u_{\theta,tt}^{(e)}\,\phi_i^{(e)} + c_e^2\, u_{\theta,x}^{(e)}\phi_{i,x}^{(e)} \right) \psi_j(t)\,dx \\ + \text{ boundary \& initial terms.}$$

Augmented Lagrangian for boundary/interface conditions. Let $C(\theta)$ denote the collection of boundary and vertex transmission constraints (Dirichlet values, continuity, Kirchhoff flux balance). Introduce multipliers λ and penalty $\beta > 0$:

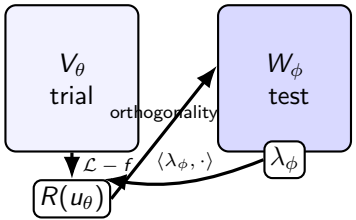
$$\mathcal{L}(\theta,\lambda) = \sum_{i,j} \mathcal{R}(u_\theta;\phi_i\psi_j)^2 \;+\; \lambda^\top C(\theta) \;+\; \frac{\beta}{2}\|C(\theta)\|^2.$$

Comparison: Classical PINN vs. AL–PINN (Petrov–Galerkin view)

Fixed test space ⇒ penalty-like enforcement



Dual network ⇒ adaptive Petrov–Galerkin



Classical PINN	**VPINN**	**AL–PINN**	**AL–VPINN**
$R(u_\theta) \perp W_{\text{coll}} = \text{span}\{\delta_{x_i}\}$	$R(u_\theta) \perp W_h = \text{span}\{\text{poly over } T_h\}$	$R(u_\theta) \perp W_\phi$ (learned test space)	$R(u_\theta) \perp W_\phi$ (weak coll.)
e.g. Karniadakis et al. since 2016		**e.g. Basir et al. 2025**	**in preparation**
		PECANN	

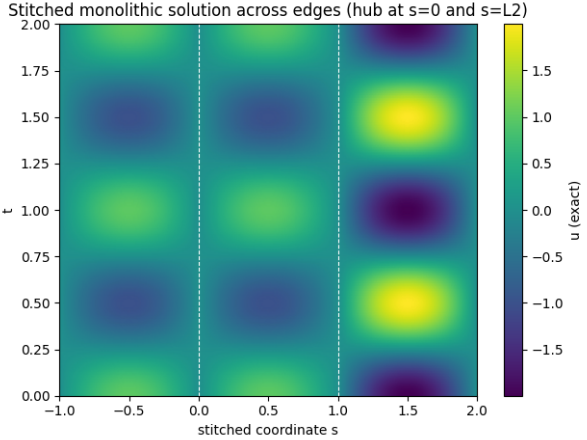
AL–PINN vs. standard PINN

AL–PINN (augmented Lagrangian)

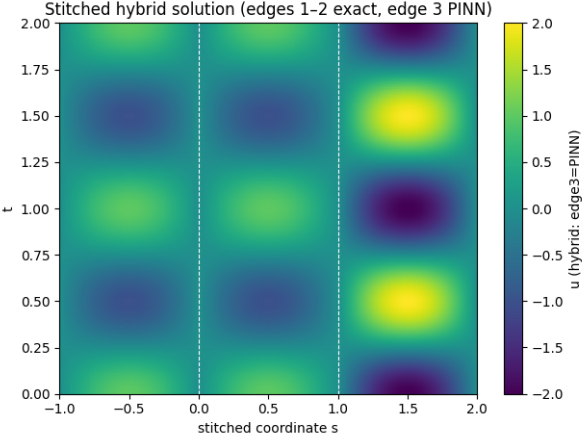
- Works in a **larger test space**: constraints enforced via multipliers and penalties.
- Interfaces handled robustly (continuity & Kirchhoff unified).
- Smoother convergence; residuals may start larger but decay steadily.
- Slightly higher computational cost.
- (DWR) goal oriented error estimates possible. (tbd)

Standard PINN (hard / soft constraints)

- Simpler: no multipliers or penalty parameters.
- Enforces interface conditions directly (smaller early residuals).
- May over-constrain training, underfitting PDE interior.
- Coupling can be stiff and sensitive to loss weights.
- Limited flexibility for mixed / Robin-type interfaces.



Monolithic sollution



Hybrid FEM-PINN solution; see the code of E. Zuazua's group

Outlook: block-stochastic descent in ALG3 - SDD-RB vs. RDD

Setting.

State/adjoint/leaf control on each active subgraph

Interface variables.

Virtual port fluxes $g_{n,e}$ used only for: trace continuity, Kirchhoff balance, AL/ALG3 coupling.

Diagnostics.

Energy, continuity, and Kirchhoff residuals.

Micro-step (NeTI–ALM/ALG3).

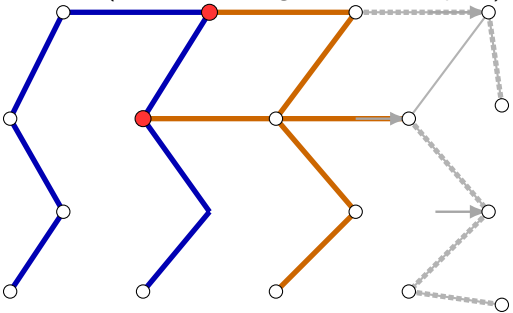
Step 1: Local solves on active subgraphs for (u^k, p^k, w^k) using $(q^{k-1}, z^{k-1}, \lambda^k, \eta^k, \nu^k)$.
Step 2: Half-step multipliers $(\lambda^{k+1/2}, \eta^{k+1/2}, \nu^{k+1/2})$.
Step 3: Update primals (q^k, z^k) with half-step multipliers.
Step 4: Full Uzawa update $(\lambda^{k+1}, \eta^{k+1}, \nu^{k+1})$.
Step 5 (leaves): New control via optimality

Activation policies.

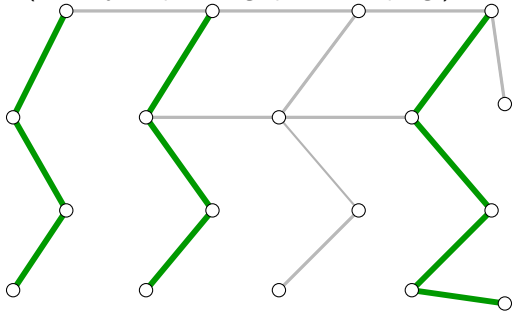
- *SDD:* all subgraphs active; no random selcetion takes places; (see the above)
- *SDD–RB:* random batch $\mathcal{B}_k \subset \{1, \dots, P\}$; we choose randomly from the *given* decomposition
- *RDD:* fresh random partition $G = \bigcup_p G_k^{(p)}$ per step. c.f. Hernández/Zuazua 2025

SDD–RB vs. RDD on same meta-graph

SDD–RB (B_1 blue, B_2 orange active; red = ports)



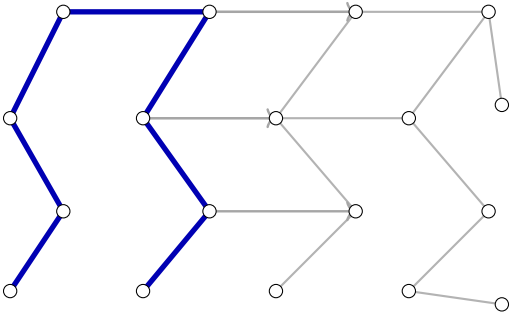
RDD (arbitrary sampled subgraphs; no couplings)



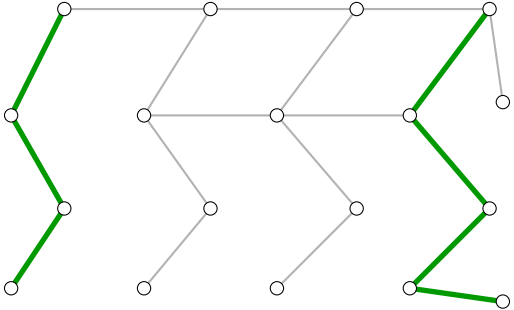
green = arbitrary sampled subgraphs $\mathcal{S}_k$; faint = not sampled; no a

Randomized selections: SDD–RB vs. RDD (step 1/3)

SDD–RB (step 1: only B_1 active)

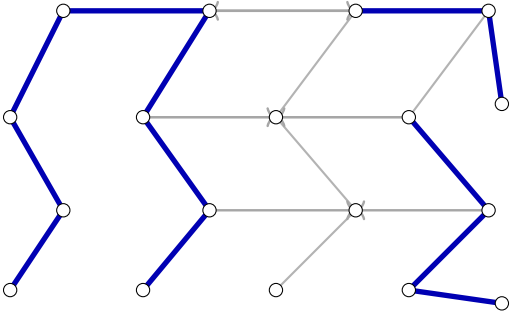


RDD (step 1: one arbitrary sample)

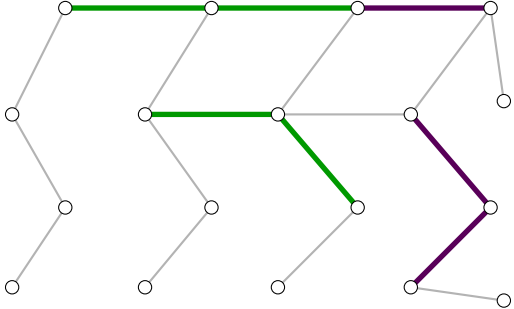


Randomized selections: SDD–RB vs. RDD (step 3/3)

SDD–RB (step 3: B_1 and B_3 active, no common ports)



RDD (step 3: new arbitrary samples, no SDD arrows)



Postlude: Speinshart Science Center for AI and SuperTech (SSC)

https://speinshart.ai

Tante Grazie!