

# On Transforming Tight Coupling Into Loose Coupling for a Class of Multi-Scale Models

**K. C. Park and Carlos A. Felippa**  
Center for Aerospace Structures,  
University of Colorado,  
Boulder, Colorado, USA.

**Denis Caillerie**  
Laboratoire 3S, Domaine Universitaire  
Grenoble, France.

**Roger Ohayon**  
Structural Mechanics and Coupled Systems Laboratory,  
Conservatoire National des Arts et Metiers (CNAM),  
Paris, France.

## Definitions of **Tight Coupling** and **Loose Coupling**

***Tight Coupling:*** two or more solution vectors occupy the same spatial domain, and/or one or more field-variables are embedded as parameters into the governing equations of another solution vectors (often called constitutive coupling).

***Loose Coupling:*** the solution vector of one-field governing equations is coupled with those of other field equations through the interface (**and localized**) Lagrange multipliers.

## Background for Present (for now idea-only) Work

- Most existing treatments of **tight coupling** employ specific boundary-condition impositions (MD-continuum coupling, bridging scales), the use of periodicity (homogenization), scale separation (turbulence, concurrent coupling), etc.
- From the classical discretization viewpoint of finite difference(differential) vs. finite element(variational) methods, a majority of the tight coupling handling schemes are akin to finite difference approach rather than variational discretization approach.
- The idea proposed herein is an attempt to bypass tailored finite difference-like treatment and employ variational and “**kinematical**” approach.

## Background for Present (for now idea-only) Work - cont'd

- From the outset, we claim that (**localized**) Lagrange multipliers are the most accurate and elegant way of effecting loose coupling. In principle, Lagrange multipliers can treat both smooth and discontinuous interfaces.
- Also, conventional notion that Lagrange multipliers require challenges in the solution of the resulting loosely coupled systems has by and large been alleviated thanks to the advances in equation solvers.

## Model Problem:

$$J(u) = \frac{1}{2} \int_V \sum_{i,j} a_{i,j}^\epsilon(x, \mathbf{u}, x/\epsilon) \frac{\partial \mathbf{u}}{\partial x_i} \frac{\partial \mathbf{u}}{\partial x_j} dV - \int_S \mathbf{u}^T \mathbf{f}(x, t) dS$$

## Two-scale approximation:

$$\mathbf{u} = \sum_{j=1}^m \mathbf{N}_0(\mathbf{x}_j) \mathbf{u}_{0j} + \sum_{j=1}^n \mathbf{N}_1(\mathbf{x}_j) \mathbf{u}_{1j}$$

## Resulting tightly coupled matrix equation

$$\begin{bmatrix} \mathbf{A}_{00} & \mathbf{A}_{01} \\ \mathbf{A}_{10} & \mathbf{A}_{11} \end{bmatrix} \begin{Bmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_0 \\ \mathbf{f}_1 \end{Bmatrix}$$

## Loosening tight coupling via interpolation

**Basic idea:** decompose the slowly varying components,  $\mathbf{u}_0$ , into intrinsic component ( $\mathbf{v}_0$ ) and extrinsic component,  $\mathcal{L}\mathbf{v}_1$ , where  $\mathbf{v}_1$  is intrinsically fast-varying components.

$$\mathbf{u}_0 = \mathbf{v}_0 + \mathcal{L}\mathbf{v}_1, \quad \mathbf{u}_1 = \mathbf{v}_1$$

The resulting transformed equation:

$$\begin{bmatrix} \mathbf{A}_{00} & (\mathbf{A}_{00}\mathcal{L} + \mathbf{A}_{01}) \\ \text{sym.} & (\mathbf{A}_{11} + \mathcal{L}^T\mathbf{A}_{01} + \mathbf{A}_{10}\mathcal{L} + \mathcal{L}^T\mathbf{A}_{00}\mathcal{L}) \end{bmatrix} \begin{Bmatrix} \mathbf{v}_0 \\ \mathbf{v}_1 \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_0 \\ \mathbf{f}_1 + \mathcal{L}^T\mathbf{f}_0 \end{Bmatrix}$$

**Remark:** If  $\mathcal{L}$  is a projection operator, it is akin to a bridging scale method; and, if  $\mathcal{L}$  is an averaging operator, it is akin to a multi-grid method.

## Loosening tight coupling via interpolation - cont'd

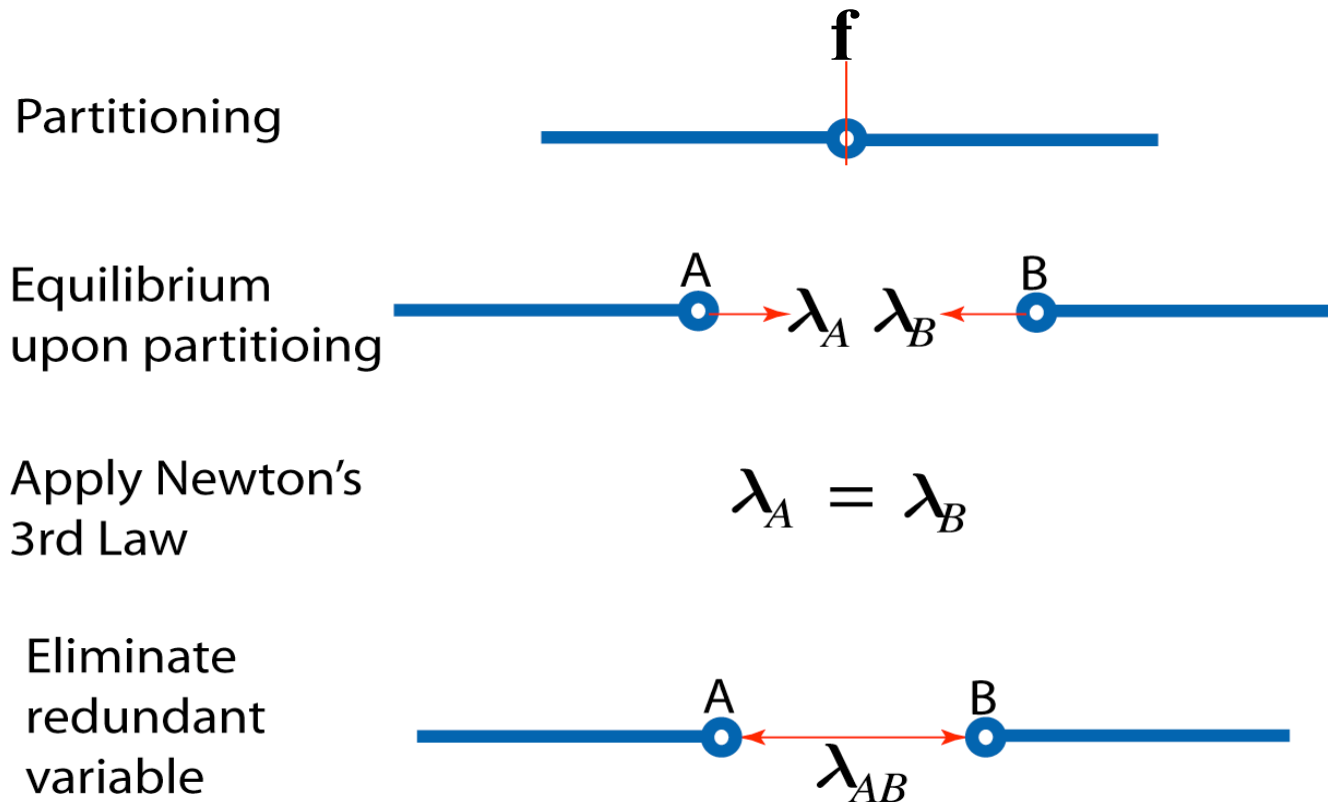
Second, introduce a frame variable ( $\mathbf{u}_f$ ) to **localize**  $\mathbf{v}_0$  from ( $\mathbf{v}_1$ ):

$$\mathbf{B}^T \mathbf{v} - \mathbf{L} \mathbf{u}_f = 0, \quad \mathbf{B} = \begin{Bmatrix} \mathbf{B}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_1 \end{Bmatrix}, \quad \mathbf{L} = \begin{Bmatrix} \mathbf{L}_0 \\ \mathbf{L}_1 \end{Bmatrix}, \quad \mathbf{v} = \begin{Bmatrix} \mathbf{v}_0 \\ \mathbf{v}_1 \end{Bmatrix}$$

**Remark:** if  $\mathbf{v}$  is a kinematic variable (viz., displacement), then the localized Lagrange multipliers to enforce the localized constraint are physical interface forces; on the other hand, if  $\mathbf{v}$  represents a jump condition or tractions, then the localized Lagrange multipliers are gradient (viz., strains) or first-order kinematical tensor components.

# A quick excursion to localized Lagrange multipliers

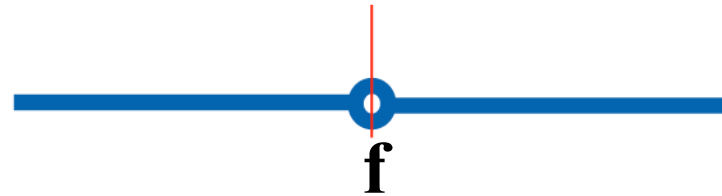
## Classical Notion of Lagrange Multipliers



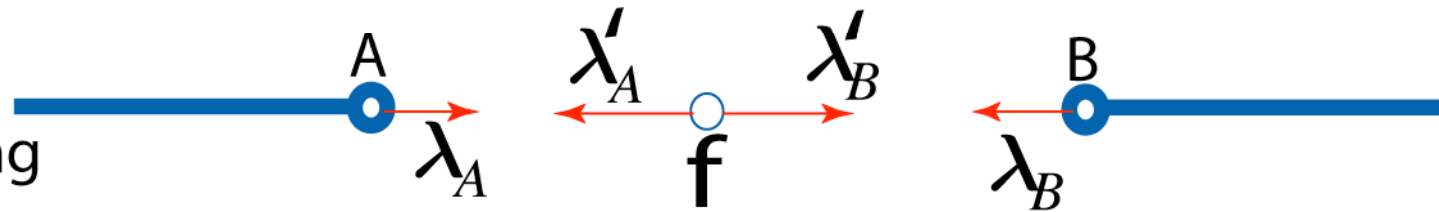
**Partitioning node,  $f$ , is lost!**

# New Interpretation of Lagrange Multipliers

Partitioning



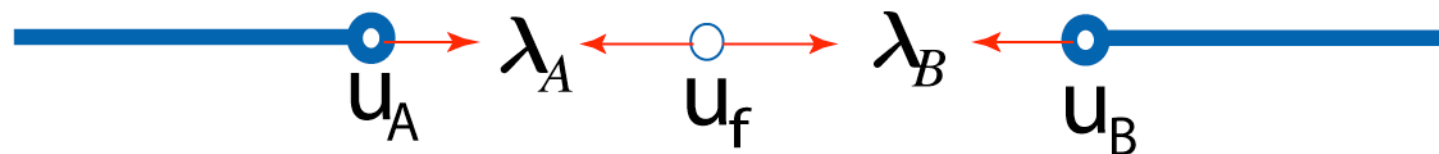
Equilibrium upon partitioning



Apply Newton's 3rd Law

$$\lambda_A = \lambda'_A \quad \text{and} \quad \lambda_B = \lambda'_B$$

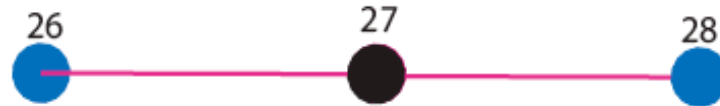
Eliminate redundant variable



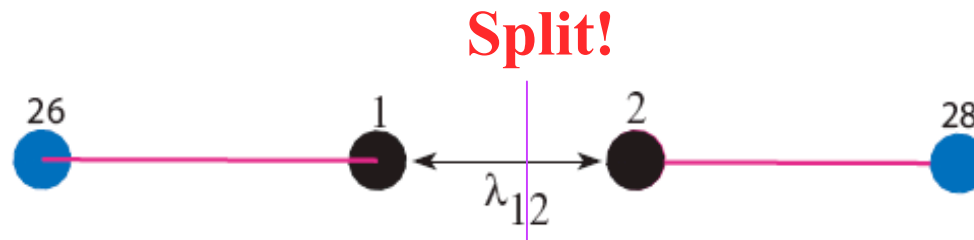
**Partitioning node,  $f$ , is preserved!**

# Localization of Classical Lagrange Multipliers

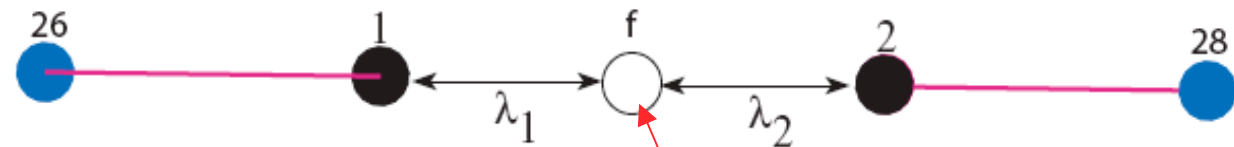
Assembled



Classical Partitioning

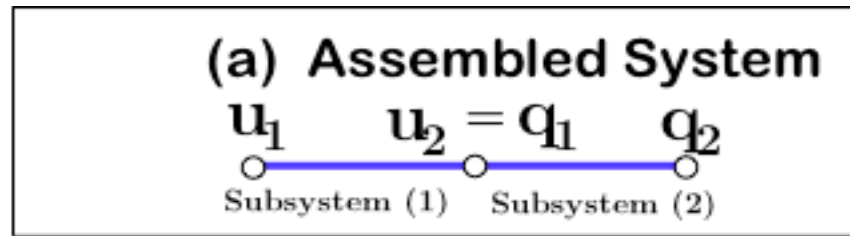


Localized Partitioning

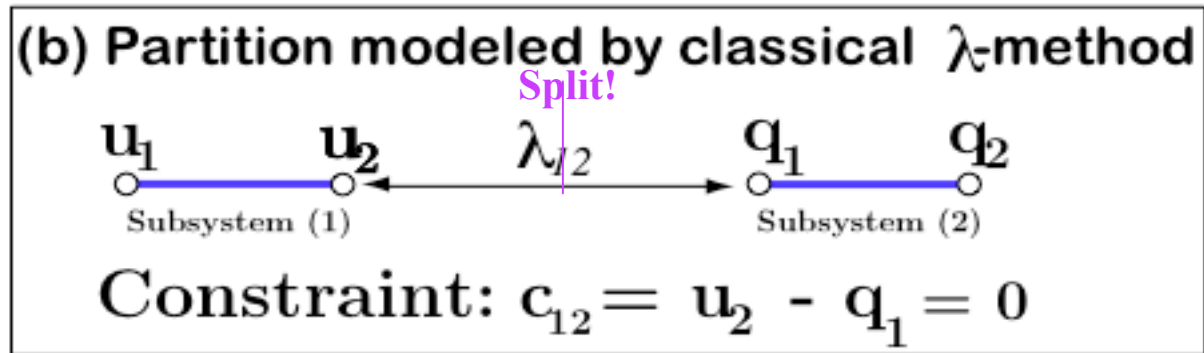


*Localization is achieved by introducing a frame node,  $f$*

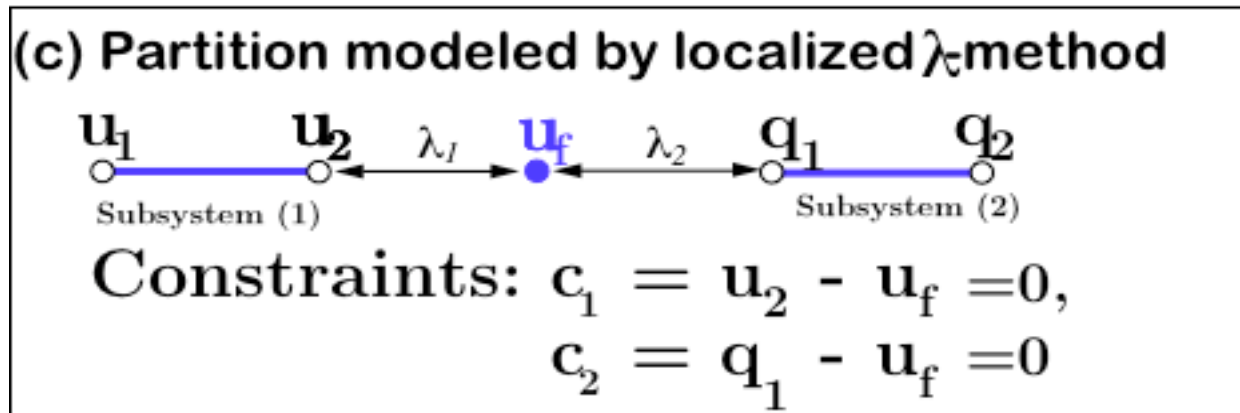
# Localization Process



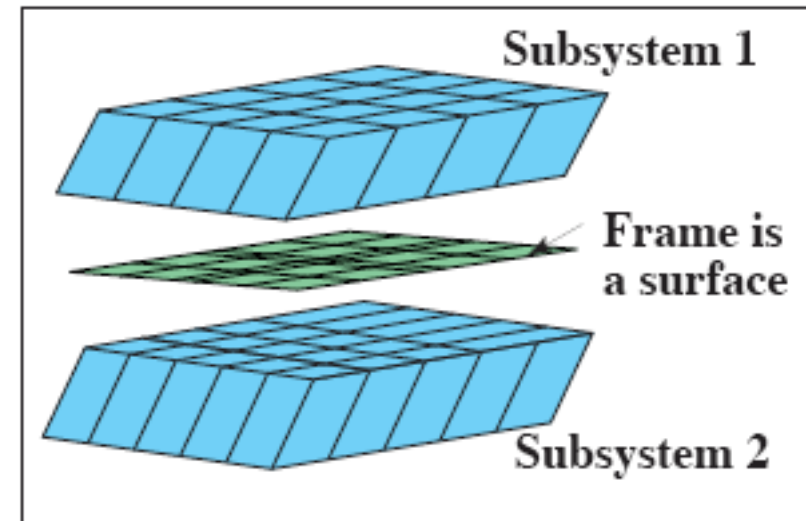
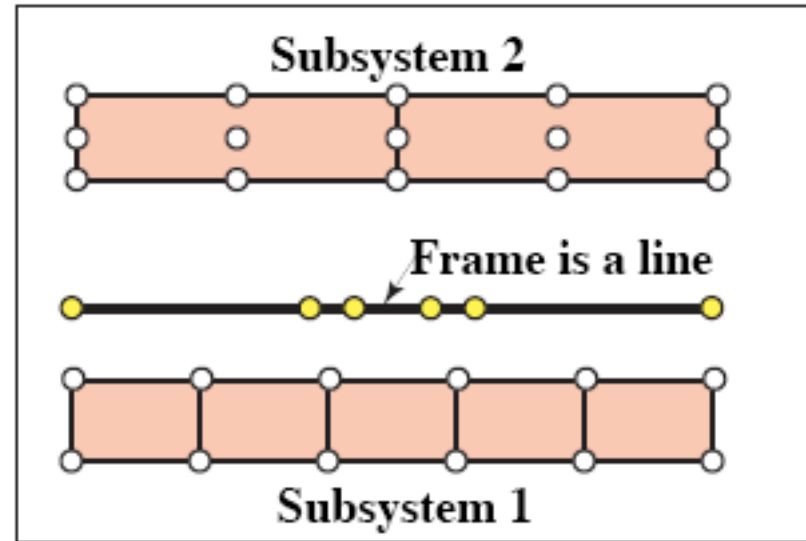
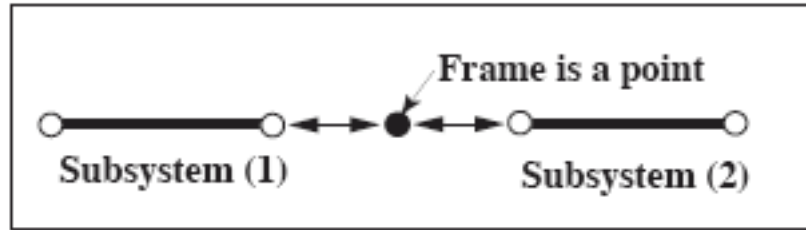
Partitioning ↓



Further Partitioning ↓



# Localization Frame Features



# *Why increase the number of Lagrange multipliers?*

Excerpts from what **William Hamilton** read at Royal Society of London in 1834:

*“While science is advancing in one direction by the improvement of physical laws, it may advance in another direction also by the invention of mathematical methods.” ...*

*“This difficulty is therefore at least transferred from the integration of many equations of one class to the integration of two of another.”*

## *Why increase the number of Lagrange multipliers? -- cont'd*

*“Even if it should be thought that no practical facility is gained, yet an intellectual pleasure may result from ...”*

**Here, Hamilton refers to the canonical transformation of  $N$  equations of motion to  $2N$  equations:**

**Euler-Lagrange Equation:**

$$\mathbf{m} \ddot{\mathbf{x}}(t) + \mathbf{k} \mathbf{x}(t) = \mathbf{f}(t)$$

**Hamilton's Equation:  $\dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u}$**

$$\begin{Bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{p}}(t) \end{Bmatrix} + \begin{bmatrix} 0 & -\mathbf{m}^{-1} \\ \mathbf{m}^{-1} \mathbf{k} & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{x}(t) \\ \mathbf{p}(t) \end{Bmatrix} = \begin{Bmatrix} 0 \\ \mathbf{m}^{-1} \mathbf{f}(t) \end{Bmatrix}$$

# Localized Lagrange Multipliers

The localized modeling, as in Hamilton's canonical equations, also increases the interface unknowns from  $N$  to  $2N$  multipliers for two interfaces.

Now the question is:

Is the localized modeling just an intellectual pleasure(exercise) or does it offer also a practical utility?

## Loosening tight coupling via interpolation - cont'd

Third, enforce the frame interface condition by Lagrange multipliers to yield the following loosely coupled system:

$$\begin{bmatrix} \mathbf{A}_{00}^{(0)} & 0 & \mathbf{B}_0 & 0 & 0 \\ 0 & \mathbf{A}_{11} & 0 & \mathbf{B}_1 & 0 \\ \mathbf{B}_0^T & 0 & 0 & 0 & -\mathbf{L}_0 \\ 0 & \mathbf{B}_1^T & 0 & 0 & -\mathbf{L}_1 \\ 0 & 0 & -\mathbf{L}_0^T & -\mathbf{L}_1^T & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{v}_0 \\ \mathbf{v}_1 \\ \lambda_0 \\ \lambda_1 \\ \mathbf{u}_f \end{Bmatrix} = \begin{Bmatrix} \mathbf{b}_0 \\ \mathbf{b}_1 \\ 0 \\ 0 \\ 0 \end{Bmatrix}$$
$$\mathbf{A}_{00}^{(0)} = \begin{bmatrix} \mathbf{A}_{00} & \mathbf{A}_{00}\mathcal{L} + \mathbf{A}_{01} \\ \text{sym.} & \mathcal{L}^T \mathbf{A}_{01} + \mathbf{A}_{10}\mathcal{L} + \mathcal{L}^T \mathbf{A}_{00}\mathcal{L} \end{bmatrix}$$

## Properties of Interpolator (Low-Pass Filter), $\mathcal{L}$

If both the fast-varying components,  $\mathbf{v}_1$ , and slowly varying component,  $\mathbf{u}_0$ , are associated with sufficient discrete nodal points and/or evaluation points (e.g., meshless or SPH), then the interpolator,  $\mathcal{L}$ , can be constructed as a geometrical filter.

For such a case, the zero-th order consistency of the frame discretization provides the interpolator.

## Loosening tight coupling via extrapolator

**Basic idea:** decompose the fast-varying components,  $\mathbf{u}_1$ , into intrinsic component ( $\hat{\mathbf{v}}_1$ ) and extrinsic component,  $\mathcal{H}\mathbf{u}_0$ , where  $\mathbf{u}_0$  is intrinsically slowly varying components.

$$\mathbf{u}_0 = \hat{\mathbf{v}}_0, \quad \mathbf{u}_1 = \hat{\mathbf{v}}_1 + \mathcal{H}\hat{\mathbf{v}}_0$$

The resulting transformed equation:

$$\begin{bmatrix} (\mathbf{A}_{00} + \mathcal{H}^T \mathbf{A}_{01} + \mathbf{A}_{10} \mathcal{H} + \mathcal{H}^T \mathbf{A}_{11} \mathcal{H}) & (\mathcal{H}^T \mathbf{A}_{11} + \mathbf{A}_{01}) \\ \text{sym.} & \mathbf{A}_{11} \end{bmatrix} \begin{Bmatrix} \hat{\mathbf{v}}_0 \\ \hat{\mathbf{v}}_1 \end{Bmatrix} \\ = \begin{Bmatrix} \mathbf{f}_0 + \mathcal{H}^T \mathbf{f}_1 \\ \mathbf{f}_1 \end{Bmatrix}$$

## Loosening tight coupling via extrapolation - cont'd

Second, introduce a frame variable ( $\mathbf{u}_f$ ) to separate  $\hat{\mathbf{v}}_0$  from  $\hat{\mathbf{v}}_1$ :

$$\mathbf{B}^T \hat{\mathbf{v}} - \mathbf{L} \hat{\mathbf{u}}_f = 0$$

$$\mathbf{B} = \begin{Bmatrix} \mathbf{B}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_1 \end{Bmatrix}, \quad \mathbf{L} = \begin{Bmatrix} \mathbf{L}_0 \\ \mathbf{L}_1 \end{Bmatrix}, \quad \hat{\mathbf{v}} = \begin{Bmatrix} \hat{\mathbf{v}}_0 \\ \hat{\mathbf{v}}_1 \end{Bmatrix}$$

Third, enforce the frame interface condition by Lagrange multipliers to yield the following loosely coupled system:

$$\begin{bmatrix} \mathbf{A}_{00} & \mathbf{0} & \mathbf{B}_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{A}}_{11} & \mathbf{0} & \mathbf{B}_1 & \mathbf{0} \\ \mathbf{B}_0^T & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{L}_0 \\ \mathbf{0} & \mathbf{B}_1^T & \mathbf{0} & \mathbf{0} & -\mathbf{L}_1 \\ \mathbf{0} & \mathbf{0} & -\mathbf{L}_0^T & -\mathbf{L}_1^T & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \hat{\mathbf{v}}_0 \\ \hat{\mathbf{v}}_1 \\ \hat{\lambda}_0 \\ \hat{\lambda}_1 \\ \hat{\mathbf{u}}_f \end{Bmatrix} = \begin{Bmatrix} \hat{\mathbf{b}}_0 \\ \hat{\mathbf{b}}_1 \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{Bmatrix}$$

$$\hat{\mathbf{A}}_{11} = \begin{bmatrix} \mathcal{H}^T \mathbf{A}_{01} + \mathbf{A}_{10} \mathcal{H} + \mathcal{H}^T \mathbf{A}_{11} \mathcal{H} & \mathcal{H}^T \mathbf{A}_{11} + \mathbf{A}_{01} \\ \text{sym.} & \mathbf{A}_{11} \end{bmatrix}$$

## *Construction of interpolation operator, $\mathcal{L}$*

**Obtain the mapping operators,  $\mathbf{L}_0$  and  $\mathbf{L}_1$  from the slowly and fast varying components onto the frame,  $\mathbf{u}_f$ :**

$$\bar{\mathbf{v}}_0 = \mathbf{B}_0 \mathbf{v}_0 = \mathbf{L}_0 \mathbf{u}_f$$

$$\bar{\mathbf{v}}_1 = \mathbf{B}_1 \mathbf{v}_1 = \mathbf{L}_1 \mathbf{u}_f$$

**Obtain the relation from the fast varying component to the frame component:**

$$\mathbf{u}_f = (\mathbf{L}_1^T \mathbf{L}_1)^{-1} \mathbf{L}_1^T \bar{\mathbf{v}}_1$$

**Finally, project onto the slowly varying frame to obtain:**

$$\bar{\mathbf{v}}_0 = \mathcal{L} \bar{\mathbf{v}}_1, \quad \mathcal{L} = \mathbf{L}_0 (\mathbf{L}_1^T \mathbf{L}_1)^{-1} \mathbf{L}_1^T$$

## *Construction of extrapolation operator, $\mathcal{H}$*

**Obtain the mapping operators,  $\mathbf{L}_0$  and  $\mathbf{L}_1$  from the slowly and fast varying components onto the frame,  $\mathbf{u}_f$ :**

$$\bar{\mathbf{v}}_0 = \mathbf{B}_0 \mathbf{v}_0 = \mathbf{L}_0 \mathbf{u}_f$$

$$\bar{\mathbf{v}}_1 = \mathbf{B}_1 \mathbf{v}_1 = \mathbf{L}_1 \mathbf{u}_f$$

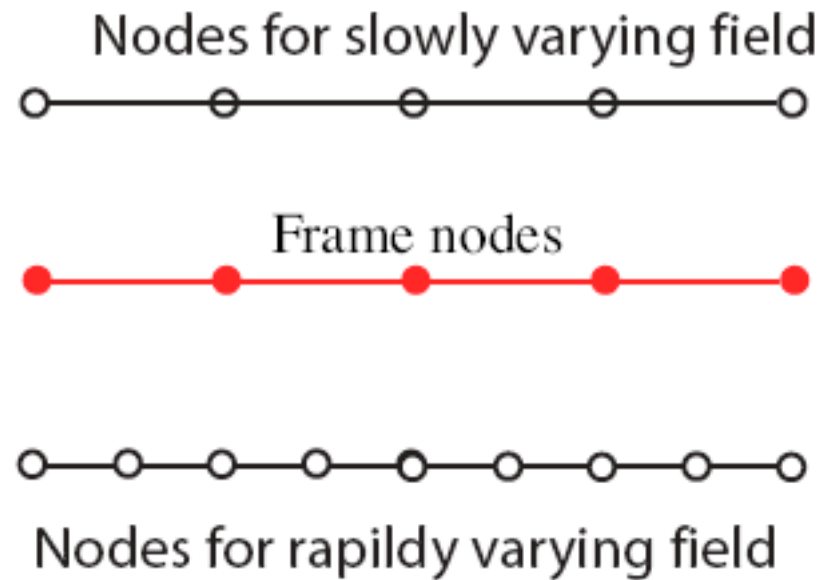
**Obtain the relation from the slowly varying component to the frame component:**

$$\mathbf{u}_f = (\mathbf{L}_0^T \mathbf{L}_0)^{-1} \mathbf{L}_0^T \bar{\mathbf{v}}_0$$

**Finally, project onto the fast varying frame to obtain:**

$$\bar{\mathbf{v}}_1 = \mathcal{H} \bar{\mathbf{v}}_0, \quad \mathcal{H} = \mathbf{L}_1 (\mathbf{L}_0^T \mathbf{L}_0)^{-1} \mathbf{L}_0^T$$

*Illustrative examples of  $\mathcal{L}$  and  $\mathcal{H}$  constructions*



(a) Matching nodes for every two nodes

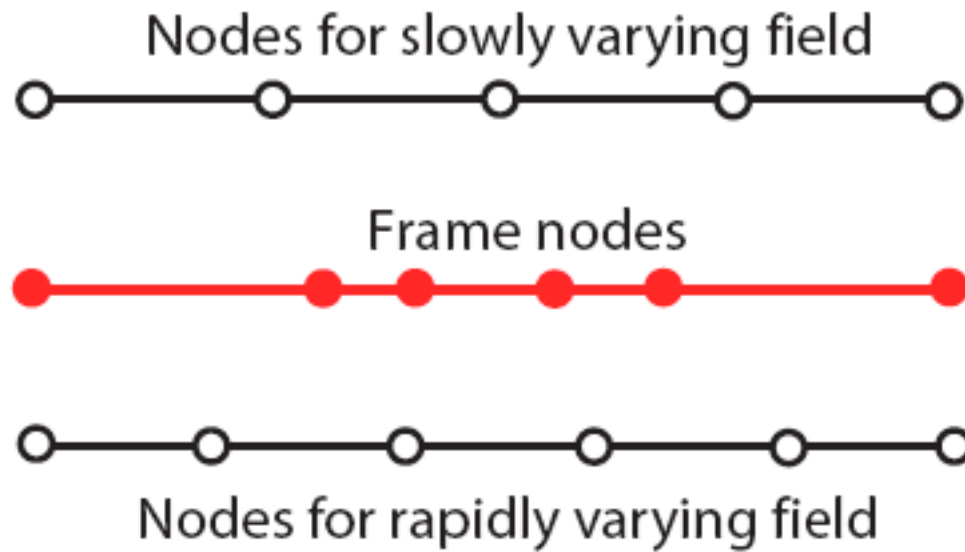
## Case of matching node for every two fast varying nodes

$$\begin{aligned}\mathcal{L} &= \mathbf{L}_0(\mathbf{L}_1^T \mathbf{L}_1)^{-1} \mathbf{L}_1^T \\ \mathcal{H} &= \mathbf{L}_1\end{aligned}\quad \mathcal{L} \mathcal{H} = \mathbf{I}$$

where  $\mathbf{L}_0$  and  $\mathbf{L}_1$  are given by

$$\mathbf{L}_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{L}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

## Case of non-matching nodes



(b) Nonmatching nodes for all the nodes

## Case of non-matching nodes - cont'd

Interpolator (Low-pass filter) :  $\mathcal{L} = \mathbf{L}_0(\mathbf{L}_1^T \mathbf{L}_1)^{-1} \mathbf{L}_1^T$

Extrapolator (High-pass filter):  $\mathcal{H} = \mathbf{L}_1(\mathbf{L}_0^T \mathbf{L}_0)^{-1} \mathbf{L}_0^T$

$$\mathbf{L}_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1/6 & 5/6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5/6 & 1/6 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathcal{L}\mathcal{H} \neq \mathbf{I}$$

$$\mathbf{L}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 2/3 & 0 & 0 & 0 & 0 \\ 0 & 3/8 & 5/8 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5/8 & 3/8 & 0 \\ 0 & 0 & 0 & 0 & 2/3 & 1/3 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

**Which is better for loose coupling,  
interpolation (low-pass geometrical filter)  
or extrapolation (high-pass filter)?**

**1 cent-worth answer: Intuitively the interpolation would be preferred. However, this awaits numerical experiments.**

**In multigrid methods, often one constructs**

$$\mathcal{L} = \mathcal{H}^T$$

**However, that is not the case for the present procedure. Whether there is any advantage or disadvantage of the present procedure, while one can always modify the present filters to possess the above property, remains to be verified.**

# Discussions

## (1) Introducing the Definitions of **Tight Coupling** and **Loose Coupling** as:

**Tight Coupling:** two or more solution vectors occupy the same spatial domain, and/or one or more field-variables are embedded as parameters into the governing equations of another solution vectors (often called constitutive coupling), and

**Loose Coupling:** the solution vector of one-field governing equations is coupled with those of other field equations through the interface (**and localized**) Lagrange multipliers.

We have presented a general framework for effecting loose coupling and introduced a special case that employs kinematical loose-coupling realization.

## Discussions - cont'd

- (2) The idea proposed herein is an attempt to bypass case-by-case finite difference-like treatments and employ variational and **kinematical** approach.
- (3) An immediate task is to **“show”** that some of the existing tailored coupling treatments can be **“reformulated”** within the framework of the proposed approach.

**A few courageous volunteers?**