Math 673

# Multigrid Methods: A Mostly Matrix-Based Approach

## Chapter 02: The Two-Grid Algorithm

Abner J. Salgado and Steven M. Wise

asalgad1@utk.edu swise1@.utk.edu
University of Tennessee

Fall 2024

# Chapter 02

# The Two-Grid Algorithm

# Overview

In this chapter, we give the basics of the two-grid algorithm. Multigrid is basically a recursive form of the two-grid algorithm. Thus, understanding the two-grid algorithm is almost essential for understanding multigrid methods.

Our approach is based on matrices rather than operators. This helps the reader focus on implementation issues and dimensional considerations. This chapter will present the material in an otherwise generic framework. There is no connection to elliptic PDE's yet. But, be patient, and we will get there.

We will build the two-grid algorithm upon the GLIS framework that we discussed in the last chapter. Though it will not be obvious at first glance, we will show that, remarkably, the two-grid algorithm is a more complicated example of a particular GLIS.

# Components of the Two-Grid Algorithm

## Definition (Two-Grid Algorithm Components)

Suppose $n_0, n_1 \in \mathbb{N}$ and $n_1 \geq n_0$. Suppose $\mathsf{A}_1 \in \mathbb{R}^{n_1 \times n_1}$ is SPD. $\mathsf{A}_1$ is called the **fine grid stiffness matrix**. Choose a matrix $\mathsf{R}_0 \in \mathbb{R}^{n_0 \times n_1}$, such that $\mathsf{R}_0$ has full rank, that is, $\operatorname{rank}(\mathsf{R}_0) = n_0$. $\mathsf{R}_0$ is called the **restriction matrix**. Define

$$\mathsf{P}_0 := \mathsf{R}_0^\top \in \mathbb{R}^{n_1 \times n_0}.$$

$\mathsf{P}_0$ is called the **prolongation matrix**. Choose a matrix $\mathsf{A}_0 \in \mathbb{R}^{n_0 \times n_0}$ that is SPD. $\mathsf{A}_0$ is called the **coarse grid stiffness matrix**. We say that the coarse-grid stiffness matrix, $\mathsf{A}_0$, **satisfies the Galerkin condition** iff it is defined via

$$\mathsf{A}_0 := \underbrace{\mathsf{R}_0}_{n_0 \times n_1} \underbrace{\mathsf{A}_1}_{n_1 \times n_1} \underbrace{\mathsf{R}_0^\top}_{n_1 \times n_0} \in \mathbb{R}^{n_0 \times n_0}. \tag{1}$$

# Visualization of the Galerkin Condition

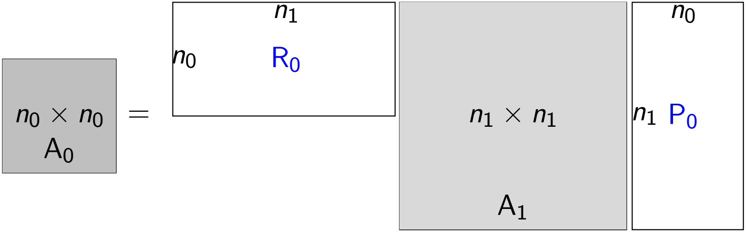


Figure: The creation of the coarse grid matrix via the Galerkin condition.

## Proposition

*If* $\mathsf{A}_1 \in \mathbb{R}^{n_1 \times n_1}$ *is SPD and* $\mathsf{A}_0$ *satisfies the Galerkin condition, then* $\mathsf{A}_0 \in \mathbb{R}^{n_0 \times n_0}$ *is also SPD.*

## Proof.

(Symmetry-ness): Since $\mathsf{A}_0 = \mathsf{R}_0 \mathsf{A}_1 \mathsf{R}_0^\top$ and $\mathsf{A}_1$ is SPD, then

$$\mathsf{A}_0^\top = \mathsf{R}_0 \mathsf{A}_1^\top \mathsf{R}_0^\top = \mathsf{R}_0 \mathsf{A}_1 \mathsf{R}_0^\top = \mathsf{A}_0.$$

(Positive definite-ness): Let $\boldsymbol{w} \in \mathbb{R}^{n_0}$ an arbitrary vector and $\boldsymbol{w} \neq \mathbf{0}$. Since $\mathsf{R}_0$ has full rank, then $\mathsf{R}_0^\top \boldsymbol{w} \neq \mathbf{0}$. Moreover, since $\mathsf{A}_1$ is SPD,

$$\boldsymbol{w}^\top \mathsf{A}_0 \boldsymbol{w} = \boldsymbol{w}^\top \mathsf{R}_0 \mathsf{A}_1 \mathsf{R}_0^\top \boldsymbol{w} = \left(\mathsf{R}_0^\top \boldsymbol{w}\right)^\top \mathsf{A}_1 \left(\mathsf{R}_0^\top \boldsymbol{w}\right) > 0.$$

$\square$

## Remark

*Observe the actions of* $\mathsf{R}_0$ *and* $\mathsf{P}_0$*:*

$$\mathsf{R}_0 \in \mathbb{R}^{n_0 \times n_1}, \quad \mathsf{R}_0 \boldsymbol{v}_1 \in \mathbb{R}^{n_0}, \quad \forall\, \boldsymbol{v}_1 \in \mathbb{R}^{n_1},$$

*and*

$$\mathsf{P}_0 \in \mathbb{R}^{n_1 \times n_0}, \quad \mathsf{P}_0 \boldsymbol{v}_0 \in \mathbb{R}^{n_1}, \quad \forall\, \boldsymbol{v}_0 \in \mathbb{R}^{n_0}.$$

*In other words,* $\mathsf{R}_0$ *sends a vector from a higher-dimensional space into a lower-dimensional space.* $\mathsf{R}_0$ *destroys information.* $\mathsf{P}_0$ *sends a vector from a lower-dimensional space into a higher-dimensional space.* $\mathsf{P}_0$ *creates information.* *See figure on the next page.*

# Information Lost in Restriction

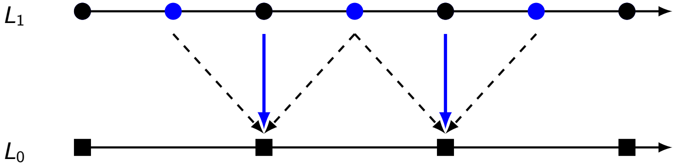


Figure: Restriction from a 1D fine grid (level 1, or $L_1$) to a 1D coarse grid (level 0, or $L_0$). Information is lost through restriction, since there are fewer degrees of freedom left in the coarse grid. Restriction can be thought of as a kind of data compression.

# Information Created in Prolongation

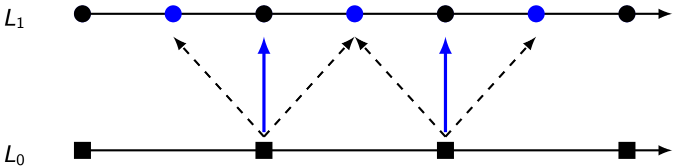


Figure: Prolongation from a 1D coarse grid (level 0, or $L_0$) to a 1D fine grid (level 1, or $L_1$). Information is created through prolongation, usually through some process of interpolation.

We wish to solve the following: given $\boldsymbol{f}_1 \in \mathbb{R}^{n_1}$ and $\mathsf{A}_1 \in \mathbb{R}^{n_1 \times n_1}$ (SPD), find $\boldsymbol{u}_1^{\mathrm{E}} \in \mathbb{R}^{n_1}$ such that

$$\mathsf{A}_1 \boldsymbol{u}_1^{\mathrm{E}} = \boldsymbol{f}_1.$$

To do this, we define a sequence of approximations

$$\boldsymbol{u}_1^{k+1} := \mathrm{TG}\left(\boldsymbol{f}_1, \boldsymbol{u}_1^{k}\right)$$

using the two-grid operator, which we now define.

## Definition (Two-Grid Operator)

Suppose that $\boldsymbol{g}_1, \boldsymbol{u}_1^{(0)} \in \mathbb{R}^{n_1}$ are given. The vector $\boldsymbol{u}_1^{(3)} \in \mathbb{R}^{n_1}$ is computed via the **two-grid operator**

$$\boldsymbol{u}_1^{(3)} := \mathrm{TG}\left(\boldsymbol{g}_1, \boldsymbol{u}_1^{(0)}\right),$$

as follows:

1. Pre-smoothing:
   - $\boldsymbol{u}_1^{(1,0)} := \boldsymbol{u}_1^{(0)}$;
   - $\boldsymbol{u}_1^{(1,\sigma+1)} := \boldsymbol{u}_1^{(1,\sigma)} + \mathsf{S}_1\left(\boldsymbol{g}_1 - \mathsf{A}_1\boldsymbol{u}_1^{(1,\sigma)}\right), \quad 1 \leq \sigma \leq m_1 - 1$;
   - $\boldsymbol{u}_1^{(1)} := \boldsymbol{u}_1^{(1,m_1)}$.

## Definition (Two-Grid Operator, Cont.)

Suppose that $\boldsymbol{g}_1, \boldsymbol{u}_1^{(0)} \in \mathbb{R}^{n_1}$ are given. The vector $\boldsymbol{u}_1^{(3)} \in \mathbb{R}^{n_1}$ is computed via the **two-grid operator**

$$\boldsymbol{u}_1^{(3)} := \mathrm{TG}\left(\boldsymbol{g}_1, \boldsymbol{u}_1^{(0)}\right),$$

as follows:

2. Coarse grid correction:
   - $\boldsymbol{r}_1^{(1)} := \boldsymbol{g}_1 - \mathsf{A}_1 \boldsymbol{u}_1^{(1)}$;
   - $\boldsymbol{r}_0^{(1)} := \mathsf{R}_0 \boldsymbol{r}_1^{(1)}$;
   - $\boldsymbol{q}_0^{(1,\mathrm{E})} := \mathsf{A}_0^{-1} \boldsymbol{r}_0^{(1)}$;
   - $\boldsymbol{q}_0^{(1)} = \boldsymbol{q}_0^{(1,\mathrm{E})}$;
   - $\boldsymbol{q}_1^{(1)} := \mathsf{P}_0 \boldsymbol{q}_0^{(1)}$;
   - $\boldsymbol{u}_1^{(2)} := \boldsymbol{u}_1^{(1)} + \boldsymbol{q}_1^{(1)}$.

## Definition (Two-Grid Operator, Cont.)

Suppose that $\boldsymbol{g}_1, \boldsymbol{u}_1^{(0)} \in \mathbb{R}^{n_1}$ are given. The vector $\boldsymbol{u}_1^{(3)} \in \mathbb{R}^{n_1}$ is computed via the **two-grid operator**

$$\boldsymbol{u}_1^{(3)} := \mathrm{TG}\left(\boldsymbol{g}_1, \boldsymbol{u}_1^{(0)}\right),$$

as follows:

3. Post-smoothing:
   - $\boldsymbol{u}_1^{(3,0)} := \boldsymbol{u}_1^{(2)}$;
   - $\boldsymbol{u}_1^{(3,\sigma+1)} := \boldsymbol{u}_1^{(3,\sigma)} + \mathsf{S}_1^\top(\boldsymbol{g}_1 - \mathsf{A}_1\boldsymbol{u}_1^{(3,\sigma)}), \quad 0 \le \sigma \le m_2 - 1$;
   - $\boldsymbol{u}_1^{(3)} := \boldsymbol{u}_1^{(3,m_2)}$.

The element $\boldsymbol{q}_0^{(1)} \in \mathbb{R}^{n_0}$ is called the **two-grid coarse grid correction**. The element $\boldsymbol{q}_0^{(1,\mathrm{E})} \in \mathbb{R}^{n_0}$ is called the **exact coarse grid correction**. The matrix $\mathsf{S}_1$ is called the **smoothing operator** or **smoothing matrix**.

## Remark

*In the two-grid setting,* $\boldsymbol{q}_0^{(1)} = \boldsymbol{q}_0^{(1,\mathrm{E})}$*. This is a very important feature of the two-grid algorithm. In the multigrid algorithm, these two vectors will not be the same, and much of our analysis will focus on estimating their difference.*

## Remark

*The element* $\boldsymbol{q}_1^{(1)} \in \mathbb{R}^{n_1}$ *is also often referred to as the* **coarse grid correction***, even though it lives in level 1 (the fine level). Perhaps* $\boldsymbol{q}_1^{(1)}$ *would be more appropriately called the* **fine grid correction***, but this name has not stuck. In the finite element context,* $\boldsymbol{q}_0^{(1)} \in \mathbb{R}^{n_0}$ *and* $\boldsymbol{q}_1^{(1)} \in \mathbb{R}^{n_1}$ *are typically different basis coordinates for the same function, the finite element coarse grid correction function.*

## Remark

*Notice that we have pre-symmetrized the smoothing operations by using the iterator* $\mathsf{S}_1^\top$ *in the post smoothing step. These smoothing operations are classical GLIS's that we introduced in the last chapter.*

## Definition

Let $m_1$ and $m_2$ be nonnegative integers. Suppose that $\boldsymbol{u}_1^k \in \mathbb{R}^{n_1}$ is given. Then the iteration process

$$\boldsymbol{u}_1^{k+1} := \mathrm{TG}\left(\boldsymbol{f}_1, \boldsymbol{u}_1^k\right)$$

defines the **generic two-grid algorithm** for solving

$$\mathsf{A}_1 \boldsymbol{u}_1^{\mathrm{E}} = \boldsymbol{f}_1.$$

The two-grid algorithm is called **one-sided** iff $m_1 \geq 1$ and $m_2 = 0$. The algorithm is called **symmetric** iff $m_1 = m_2$.

## Remark

*Let us examine the coarse-grid correction in a very special case. Suppose that* $\mathsf{A}_0$ *satisfies the Galerkin condition,* $n_0 = n_1$*, and*

$$\mathsf{R}_0 = \mathsf{I}_1, \quad (n_1 \times n_1 \textit{ identity matrix}).$$

*Then*

$$\mathsf{A}_0 = \mathsf{R}_0 \mathsf{A}_1 \mathsf{P}_0 = \mathsf{R}_0 \mathsf{A}_1 \mathsf{R}_0^\top = \mathsf{A}_1.$$

*Let us compute the coarse grid correction:*

$$\begin{aligned} \boldsymbol{r}_1^{(1)} &:= \boldsymbol{f}_1 - \mathsf{A}_1 \boldsymbol{u}_1^{(1)}; \\ \boldsymbol{r}_0^{(1)} &:= \mathsf{R}_0 \boldsymbol{r}_1^{(1)} = \boldsymbol{r}_1^{(1)}; \\ \boldsymbol{q}_0^{(1)} &:= \mathsf{A}_0^{-1} \boldsymbol{r}_0^{(1)} = \mathsf{A}_1^{-1} \boldsymbol{r}_1^{(1)} = \boldsymbol{e}_1^{(1)}. \end{aligned}$$

## Remark (Cont.)

*Then,*

$$\begin{aligned}\boldsymbol{u}_1^{(2)} &:= \boldsymbol{u}_1^{(1)} + \mathsf{P}_0 \boldsymbol{q}_0^{(1)} \\ &= \boldsymbol{u}_1^{(1)} + \mathsf{P}_0 \boldsymbol{e}_1^{(1)} \\ &= \boldsymbol{u}_1^{(1)} + \boldsymbol{e}_1^{(1)} \\ &= \boldsymbol{u}_1^{(1)} + \boldsymbol{u}_1^{\mathrm{E}} - \boldsymbol{u}_1^{(1)} \\ &= \boldsymbol{u}_1^{\mathrm{E}},\end{aligned}$$

*where*

$$\mathsf{A}_1 \boldsymbol{u}_1^{\mathrm{E}} = \boldsymbol{f}_1.$$

## Remark (Cont.)

*Of course, the two-grid algorithm should terminate at this stage, because we have the exact solution. Now, in general*

$$\boldsymbol{q}_0^{(1)} \neq \boldsymbol{e}_1^{(1)}.$$

*But, the approximation*

$$\mathsf{P}_0 \boldsymbol{q}_0^{(1)} \approx \boldsymbol{e}_1^{(1)}.$$

*can be a quite good, in some sense. We will revisit this issue later.*

# The Coarse-Grid Ritz Projection

Now, let us show that the two-grid algorithm is a GLIS. We need only derive the iterator matrix, or, equivalently, the error propogation matrix. We need the following definitions first.

## Definition (Coarse-Grid Ritz Projection)

Referring to a previous definition, we define the matrix

$$\tilde{\Pi}_1 = \mathsf{R}_0^\top \Pi_0 \in \mathbb{R}^{n_1 \times n_1},$$

where

$$\Pi_0 = \underbrace{\mathsf{A}_0^{-1}}_{n_0 \times n_0} \underbrace{\mathsf{R}_0}_{n_0 \times n_1} \underbrace{\mathsf{A}_1}_{n_1 \times n_1} \in \mathbb{R}^{n_0 \times n_1}.$$

$\tilde{\Pi}_1$ is called the **coarse-grid Ritz projection matrix**.

## Proposition

*The matrix $\tilde{\Pi}_1$, as defined previously, is a bona fide projection matrix, that is,*

$$\tilde{\Pi}_1 \tilde{\Pi}_1 = \tilde{\Pi}_1,$$

*provided $A_0$ satisfies the Galerkin condition.*

## Proof.

Using the Galerkin condition (1),

$$\begin{aligned}
\tilde{\Pi}_1 \tilde{\Pi}_1 &= R_0^\top \Pi_0 R_0^\top \Pi_0 \\
&= R_0^\top A_0^{-1} \underbrace{R_0 A_1 R_0^\top}_{A_0} A_0^{-1} R_0 A_1 \\
&= R_0^\top A_0^{-1} R_0 A_1 \\
&= R_0^\top \Pi_0 \\
&= \tilde{\Pi}_1.
\end{aligned}$$

$\square$

## Corollary

*If the Galerkin Condition holds,* $\mathsf{I}_1 - \tilde{\Pi}_1$ *is also a projection matrix, that is,*

$$\left(\mathsf{I}_1 - \tilde{\Pi}_1\right)\left(\mathsf{I}_1 - \tilde{\Pi}_1\right) = \mathsf{I}_1 - \tilde{\Pi}_1,$$

The smoothing operation is just the application of a GLIS a certain number of times. We define the error transfer for this process in the expected way.

## Definition (Smoothing Error Transfer Matrix)

Define the **pre-smoothing error transfer matrix** as

$$\mathsf{K}_1 := \mathsf{I}_1 - \mathsf{S}_1\mathsf{A}_1.$$

## Definition (Adjoint)

Let $\mathsf{M} \in \mathbb{R}^{n_1 \times n_1}$. Define the **level-1 inner product** via

$$(\boldsymbol{u}_1, \boldsymbol{v}_1)_1 := \boldsymbol{u}_1^\top \boldsymbol{v}_1, \quad \forall\, \boldsymbol{u}_1, \boldsymbol{v}_1 \in \mathbb{R}^{n_1}.$$

The **level-1 adjoint of** $\mathsf{M}$ is the unique matrix $\mathsf{M}^\top$ that satisfies

$$(\mathsf{M}\boldsymbol{u}_1, \boldsymbol{v}_1)_1 = \left(\boldsymbol{u}_1, \mathsf{M}^\top \boldsymbol{v}_1\right)_1, \quad \forall\, \boldsymbol{u}_1, \boldsymbol{v}_1 \in \mathbb{R}^{n_1}.$$

(Of course, $\mathsf{M}^\top$ is the usual transpose of $\mathsf{M}$.) We say that $\mathsf{M}$ is **level-1 self-adjoint**, or **level-1 symmetric**, iff $\mathsf{M} = \mathsf{M}^\top$. When the context is clear, we will drop the '**level-1**' part of the terms above.

## Definition (Adjoint, Cont.)

Define the $A_1$ **inner product** via

$$(\boldsymbol{u}_1, \boldsymbol{v}_1)_{A_1} := (A_1 \boldsymbol{u}_1, \boldsymbol{v}_1)_1, \quad \forall \, \boldsymbol{u}_1, \boldsymbol{v}_1 \in \mathbb{R}^{n_1}.$$

The $A_1$ **adjoint of** M is the unique matrix $M^*$ satisfying

$$(M\boldsymbol{u}_1, \boldsymbol{v}_1)_{A_1} = (\boldsymbol{u}_1, M^* \boldsymbol{v}_1)_{A_1}, \quad \forall \, \boldsymbol{u}_1, \boldsymbol{v}_1 \in \mathbb{R}^{n_1}.$$

We say that M is $A_1$ **self-adjoint**, or $A_1$ **symmetric**, iff $M = M^*$. We say that M is $A_1$ **symmetric positive definite** iff it is $A_1$ symmetric and

$$(M\boldsymbol{u}_1, \boldsymbol{u}_1)_{A_1} > 0, \quad \forall \, \boldsymbol{u}_1 \in \mathbb{R}^{n_1}_{\star}.$$

## Proposition

*Suppose that* $\mathsf{K}_1$ *is the pre-smoothing error transfer matrix. Then,*

$$\mathsf{K}_1^* = \mathsf{I}_1 - \mathsf{S}_1^\top \mathsf{A}_1.$$

## Proof.

To see this, let $\boldsymbol{u}_1, \boldsymbol{v}_1 \in \mathbb{R}^{n_1}$ be arbitrary. Then

$$
\begin{aligned}
((\mathsf{I}_1 - \mathsf{S}_1\mathsf{A}_1)\,\boldsymbol{u}_1, \boldsymbol{v}_1)_{\mathsf{A}_1} &= (\mathsf{A}_1\,(\mathsf{I}_1 - \mathsf{S}_1\mathsf{A}_1)\,\boldsymbol{u}_1, \boldsymbol{v}_1)_1 \\
&= \left(\boldsymbol{u}_1, (\mathsf{I}_1 - \mathsf{S}_1\mathsf{A}_1)^\top \mathsf{A}_1^\top \boldsymbol{v}_1\right)_1 \\
&= \left(\boldsymbol{u}_1, \left(\mathsf{I}_1 - \mathsf{A}_1^\top \mathsf{S}_1^\top\right) \mathsf{A}_1 \boldsymbol{v}_1\right)_1 \\
&= \left(\boldsymbol{u}_1, \left(\mathsf{I}_1 - \mathsf{A}_1 \mathsf{S}_1^\top\right) \mathsf{A}_1 \boldsymbol{v}_1\right)_1 \\
&= \left(\boldsymbol{u}_1, \left(\mathsf{A}_1 - \mathsf{A}_1 \mathsf{S}_1^\top \mathsf{A}_1\right) \boldsymbol{v}_1\right)_1 \\
&= \left(\boldsymbol{u}_1, \mathsf{A}_1\left(\mathsf{I}_1 - \mathsf{S}_1^\top \mathsf{A}_1\right) \boldsymbol{v}_1\right)_1 \\
&= \left(\mathsf{A}_1\boldsymbol{u}_1, \left(\mathsf{I}_1 - \mathsf{S}_1^\top \mathsf{A}_1\right) \boldsymbol{v}_1\right)_1 \\
&= \left(\boldsymbol{u}_1, \left(\mathsf{I}_1 - \mathsf{S}_1^\top \mathsf{A}_1\right) \boldsymbol{v}_1\right)_{\mathsf{A}_1} .
\end{aligned}
$$

□

# The Two-Grid Error Transfer Matrix

We now have the tools to identify the structure of the error transfer matrix for the two-grid algorithm.

## Theorem (Two-Grid Error Transfer Matrix)

*For the two-grid algorithm defined earlier, we have*

$$\boldsymbol{e}_1^{k+1} = \mathsf{E}_1 \boldsymbol{e}_1^k,$$

*where*

$$\mathsf{E}_1 = (\mathsf{K}_1^*)^{m_2} \left( \mathsf{I}_1 - \tilde{\Pi}_1 \right) (\mathsf{K}_1)^{m_1},$$

*and*

$$\boldsymbol{e}_1^k := \boldsymbol{u}_1^{\mathrm{E}} - \boldsymbol{u}_1^k$$

*where $\boldsymbol{u}_1^{\mathrm{E}}$ is the exact solution, that is,*

$$\mathsf{A}_1 \boldsymbol{u}_1^{\mathrm{E}} = \boldsymbol{f}_1.$$

*Furthermore, the two-grid algorithm is a GLIS, that is, there is some $\mathsf{B}_1 \in \mathbb{R}^{n_1 \times n_1}$ such that*

$$\mathsf{E}_1 = \mathsf{I}_1 - \mathsf{B}_1 \mathsf{A}_1.$$

## Proof.

Set

$$\boldsymbol{e}_1^{(i)} := \boldsymbol{u}_1^{\mathrm{E}} - \boldsymbol{u}_1^{(i)}, \quad i = 1, 2, 3.$$

Then, the error after the pre-smoothing step is

$$\boldsymbol{e}_1^{(1)} := (\mathsf{I}_1 - \mathsf{S}_1\mathsf{A}_1)^{m_1}\,\boldsymbol{e}_1^k = \mathsf{K}_1^{m_1}\boldsymbol{e}_1^k.$$

Next we have the coarse-grid correction step.

$$\begin{aligned}
\boldsymbol{e}_1^{(2)} &= \boldsymbol{u}_1^{\mathrm{E}} - \boldsymbol{u}_1^{(2)} \\
&= \boldsymbol{u}_1^{\mathrm{E}} - \boldsymbol{u}_1^{(1)} - \mathsf{R}_0^{\top}\boldsymbol{q}_0^{(1)} \\
&= \boldsymbol{e}_1^{(1)} - \mathsf{R}_0^{\top}\mathsf{A}_0^{-1}\boldsymbol{r}_0^{(1)} \\
&= \boldsymbol{e}_1^{(1)} - \mathsf{R}_0^{\top}\mathsf{A}_0^{-1}\mathsf{R}_0\boldsymbol{r}_1^{(1)} \\
&= \boldsymbol{e}_1^{(1)} - \mathsf{R}_0^{\top}\mathsf{A}_0^{-1}\mathsf{R}_0\mathsf{A}_1\boldsymbol{e}_1^{(1)} \\
&= \left(\mathsf{I}_1 - \tilde{\Pi}_1\right)\boldsymbol{e}_1^{(1)}.
\end{aligned}$$

## Proof (Cont.)

So

$$\boldsymbol{e}_1^{(2)} = \left(\mathsf{I}_1 - \tilde{\Pi}_1\right) \mathsf{K}_1^{m_1} \boldsymbol{e}_1^{k}.$$

Finally, after the post-smoothing step,

$$\boldsymbol{e}_1^{k+1} = \boldsymbol{e}_1^{(3)} = \left(\mathsf{I}_1 - \mathsf{S}_1^{\top} \mathsf{A}_1\right)^{m_2} \boldsymbol{e}_1^{(2)}.$$

So

$$\boldsymbol{e}_1^{k+1} = (\mathsf{K}_1^{*})^{m_2} \left(\mathsf{I}_1 - \tilde{\Pi}_1\right) \mathsf{K}_1^{m_1} \boldsymbol{e}_1^{k},$$

as desired.

We leave the second part as an exercise. □

## Remark

*In the case that $m_1 = m_2 = 1$, we have the following simple form for $\mathsf{B}_1$:*

$$\mathsf{B}_1 = \underbrace{\mathsf{S}_1^\top \left(\mathsf{S}_1^{-\top} + \mathsf{S}_1^{-1} - \mathsf{A}_1\right) \mathsf{S}_1}_{\text{symmetric smoothing}} + \underbrace{\mathsf{K}_1^* \mathsf{R}_0^\top \mathsf{A}_0^{-1} \mathsf{R}_0 \mathsf{K}_1^{*\top}}_{\text{coarse grid correction}},$$

*which is, of course, symmetric. In general, we can show that $\mathsf{B}_1$ is symmetric in the Euclidean sense iff $m_1 = m_2 = m$.*

## Theorem (Symmetry and Positivity of $\mathsf{E}_1$)

*The error propagation matrix for the two-grid algorithm, $\mathsf{E}_1$, is symmetric with respect to $(\cdot,\cdot)_{\mathsf{A}_1}$ iff $m_1 = m_2 = m$. Futhermore, if $m_1 = m_2 = m$ and, if the Galerkin condition holds for $\mathsf{A}_0$, then $\mathsf{E}_1$ is symmetric positive semi-definite (SPSD) with respect to $(\cdot,\cdot)_{\mathsf{A}_1}$.*

## Proof.

(Symmetry): Let $\boldsymbol{u}_1, \boldsymbol{v}_1 \in \mathbb{R}^{n_1}$ be arbitrary. Then we have

$$\begin{aligned}
(\mathsf{E}_1\boldsymbol{u}_1, \boldsymbol{v}_1)_{\mathsf{A}_1} &= \left((\mathsf{K}_1^*)^{m_2}\left(\mathsf{I}_1 - \tilde{\Pi}_1\right)\mathsf{K}_1^{m_1}\boldsymbol{u}_1, \boldsymbol{v}_1\right)_{\mathsf{A}_1} \\
&= \left(\left(\mathsf{I}_1 - \tilde{\Pi}_1\right)\mathsf{K}_1^{m_1}\boldsymbol{u}_1, \mathsf{K}_1^{m_2}\boldsymbol{v}_1\right)_{\mathsf{A}_1} \\
&= \left(\mathsf{K}_1^{m_1}\boldsymbol{u}_1, \left(\mathsf{I}_1 - \tilde{\Pi}_1\right)^*\mathsf{K}_1^{m_2}\boldsymbol{v}_1\right)_{\mathsf{A}_1} \\
&= \left(\boldsymbol{u}_1, (\mathsf{K}_1^*)^{m_1}\left(\mathsf{I}_1 - \tilde{\Pi}_1\right)^*\mathsf{K}_1^{m_2}\boldsymbol{v}_1\right)_{\mathsf{A}_1}.
\end{aligned}$$

Observe that

$$\left(\mathsf{I}_1 - \tilde{\Pi}_1\right)^* = \left(\mathsf{I}_1 - \tilde{\Pi}_1\right).$$

## Proof (Cont.)

Indeed,

$$\left(\left(\mathsf{I}_1 - \tilde{\Pi}_1\right) \boldsymbol{u}_1, \boldsymbol{v}_1\right)_{\mathsf{A}_1} = (\boldsymbol{u}_1, \boldsymbol{v}_1)_{\mathsf{A}_1} - \left(\tilde{\Pi}_1 \boldsymbol{u}_1, \boldsymbol{v}_1\right)_{\mathsf{A}_1}$$

and

$$\begin{aligned}
\left(\tilde{\Pi}_1 \boldsymbol{u}_1, \boldsymbol{v}_1\right)_{\mathsf{A}_1} &= \left(\mathsf{A}_1 \mathsf{R}_0^\top \mathsf{A}_0^{-1} \mathsf{R}_0 \mathsf{A}_1 \boldsymbol{u}_1, \boldsymbol{v}_1\right)_1 \\
&= \left(\mathsf{R}_0^\top \mathsf{A}_0^{-1} \mathsf{R}_0 \mathsf{A}_1 \boldsymbol{u}_1, \mathsf{A}_1 \boldsymbol{v}_1\right)_1 \\
&= \left(\mathsf{A}_1 \boldsymbol{u}_1, \left(\mathsf{R}_0^\top \mathsf{A}_0^{-1} \mathsf{R}_0\right)^\top \mathsf{A}_1 \boldsymbol{v}_1\right)_1 \\
&= \left(\mathsf{A}_1 \boldsymbol{u}_1, \mathsf{R}_0^\top \mathsf{A}_0^{-1} \mathsf{R}_0 \mathsf{A}_1 \boldsymbol{v}_1\right)_1 \\
&= \left(\mathsf{A}_1 \boldsymbol{u}_1, \tilde{\Pi}_1 \boldsymbol{v}_1\right)_1 \\
&= \left(\boldsymbol{u}_1, \tilde{\Pi}_1 \boldsymbol{v}_1\right)_{\mathsf{A}_1}.
\end{aligned}$$

## Proof (Cont.)

Therefore

$$\tilde{\Pi}_1^* = \tilde{\Pi}_1,$$

and

$$\left(\mathsf{I}_1 - \tilde{\Pi}_1\right)^* = \left(\mathsf{I}_1 - \tilde{\Pi}_1\right).$$

The symmetry of $\mathsf{E}_1$ follows iff $m_1 = m_2$.

(PSD-ness): Let $\boldsymbol{u}_1 \in \mathbb{R}^{n_1}$ be arbitrary and suppose $m_1 = m_2 = m$. Since the Galerkin condition holds, from a corollary, we have

$$\left(\mathsf{I}_1 - \tilde{\Pi}_1\right)^2 = \left(\mathsf{I}_1 - \tilde{\Pi}_1\right).$$

## Proof (Cont.)

Now

$$
\begin{aligned}
(\mathsf{E}_1 \boldsymbol{u}_1, \boldsymbol{u}_1)_{\mathsf{A}_1} &= \left( (\mathsf{K}_1^*)^m \left( \mathsf{I}_1 - \tilde{\Pi}_1 \right) \mathsf{K}_1^m \boldsymbol{u}_1, \boldsymbol{u}_1 \right)_{\mathsf{A}_1} \\
&= \left( \left( \mathsf{I}_1 - \tilde{\Pi}_1 \right) \mathsf{K}_1^m \boldsymbol{u}_1, \mathsf{K}_1^m \boldsymbol{u}_1 \right)_{\mathsf{A}_1} \\
&= \left( \left( \mathsf{I}_1 - \tilde{\Pi}_1 \right)^2 \mathsf{K}_1^m \boldsymbol{u}_1, \mathsf{K}_1^m \boldsymbol{u}_1 \right)_{\mathsf{A}_1} \\
&= \left( \left( \mathsf{I}_1 - \tilde{\Pi}_1 \right) \mathsf{K}_1^m \boldsymbol{u}_1, \left( \mathsf{I}_1 - \tilde{\Pi}_1 \right)^* \mathsf{K}_1^m \boldsymbol{u}_1 \right)_{\mathsf{A}_1} \\
&= \left( \left( \mathsf{I}_1 - \tilde{\Pi}_1 \right) \mathsf{K}_1^m \boldsymbol{u}_1, \left( \mathsf{I}_1 - \tilde{\Pi}_1 \right) \mathsf{K}_1^m \boldsymbol{u}_1 \right)_{\mathsf{A}_1} \\
&= \left\| \left( \mathsf{I}_1 - \tilde{\Pi}_1 \right) \mathsf{K}_1^m \boldsymbol{u}_1 \right\|_{\mathsf{A}_1}^2 \geq 0.
\end{aligned}
$$

□