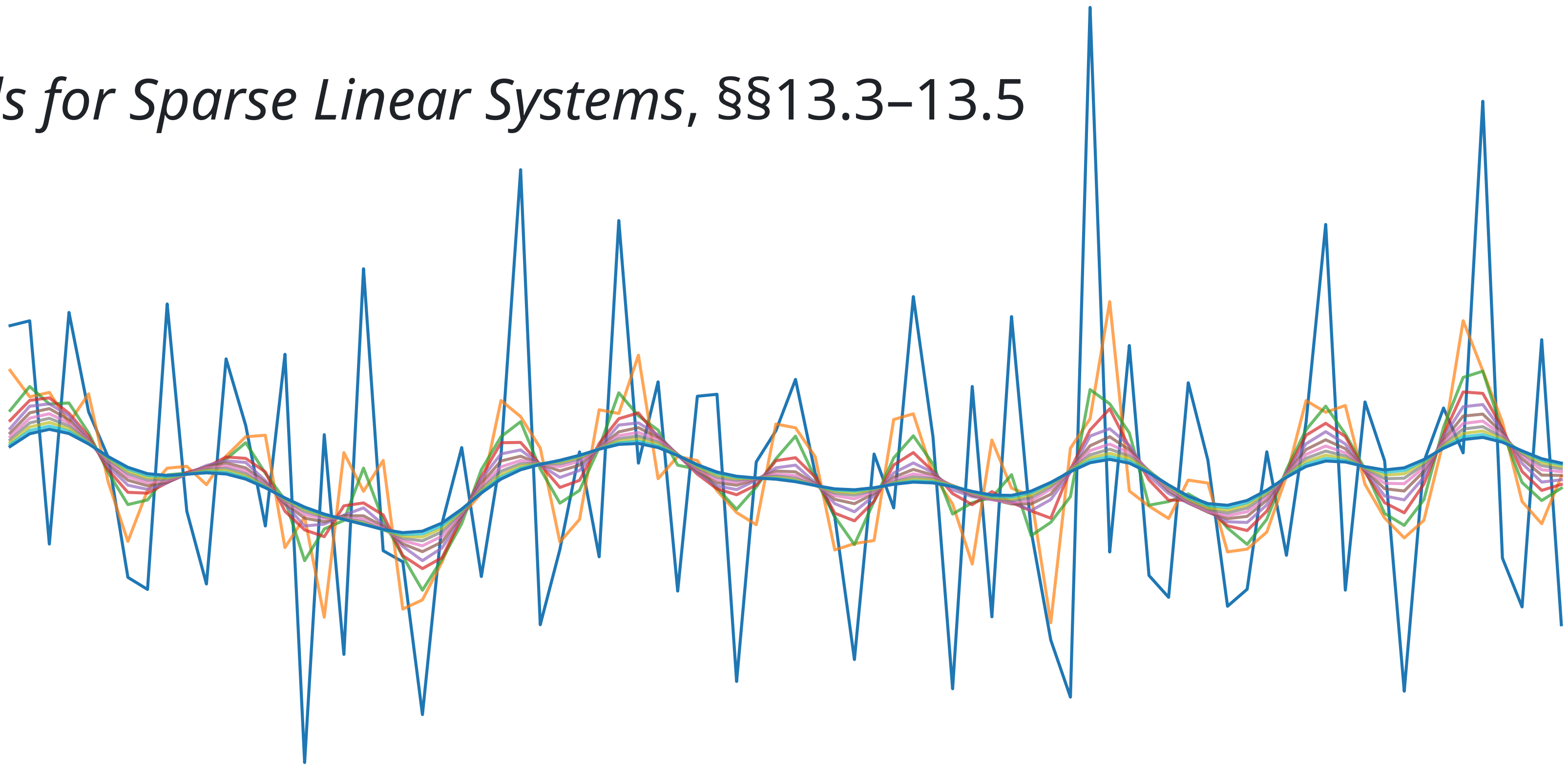
Multigrid Methods

Y. Saad, *Iterative Methods for Sparse Linear Systems*, §§13.3–13.5

Presenter: Jiaze Li



"**Talk is cheap. Show me the code.**" – Linus Torvalds

Code is available on Li-Jesse-Jiaze/multigrid-playbook.

Model Problem

1-D model problem

$$
\begin{aligned}
-u_{xx} &= f \\
u(0) = u(1) &= 0
\end{aligned}
$$

With finite differences

$$
\frac{-u_{i-1} + 2u_i - u_{i+1}}{h^2} = f_i \quad i = 1, \ldots, n \quad u_0 = u_{n+1} = 0
$$

As matrix form

$$
\frac{1}{h^2} \begin{bmatrix} 2 & -1 & \\ -1 & \ddots & \ddots \\ & \ddots & \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix}
$$

Use the Jacobi Method

Solve the problem

$$Ax = b$$

Iteratively

$$x^{(1)} = x^{(0)} + D^{-1} r^{(0)}$$

> Or consider it as a gradient descent with fixed step size 1 and preconditioner D^{-1}

Exact solution

$$x^*$$

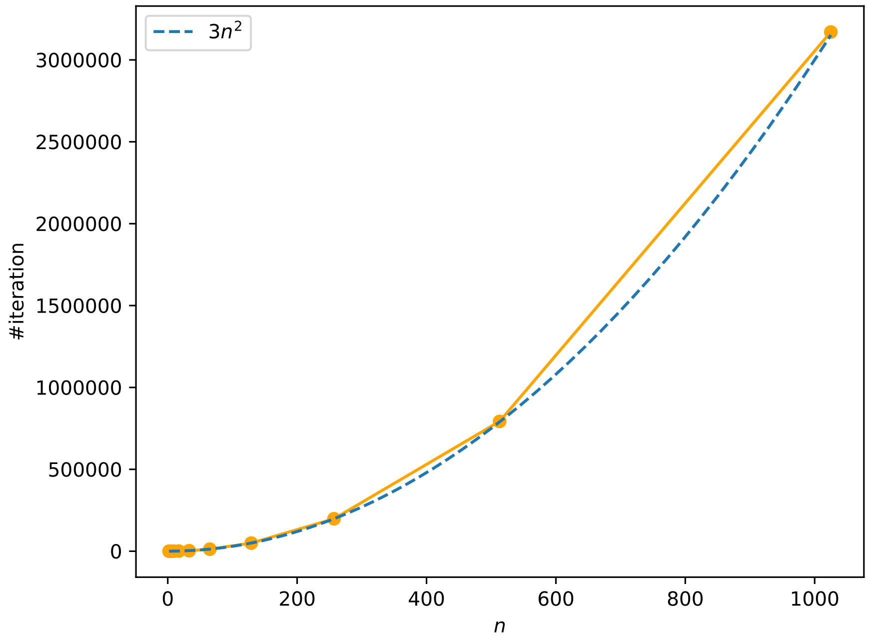
Error

$$e^{(0)} = x^* - x^{(0)}$$

Residual

$$r^{(0)} = b - Ax^{(0)} = Ae^{(0)}$$

How much computation do we need?



Burkardt, J., 2011. Jacobi Iterative Solution of Poisson's Equation in 1D [online]: Number of iterations required to get the tolerance of 1.0E-10 (RMS residual norm)

$$\text{Jacobi}: \mathcal{O}(N) \times \mathcal{O}(N^2) = \mathcal{O}(N^3)$$

$$\text{Multigrid}: \underbrace{\mathcal{O}(N)}_{\text{cost per iteration}} \times \underbrace{\mathcal{O}(1)}_{\text{number of iterations}} = \underbrace{\mathcal{O}(N)}_{\text{total complexity}}$$

Briggs et al. A multigrid tutorial P77

Look at the matrix

$$A = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}$$

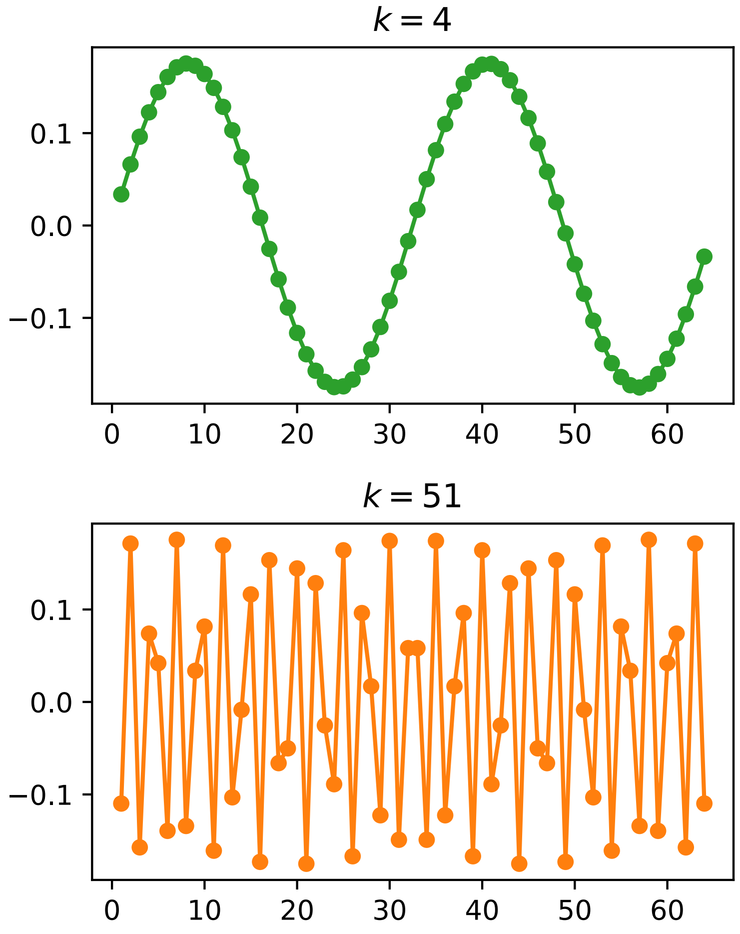
The eigenvalues

$$\lambda_k = 4\sin^2\left(\frac{k\pi}{2(n+1)}\right)$$

The eigenvectors (are Fourier modes)

$$v_k[j] = \sin\left(\frac{(j+1)*k\pi}{n+1}\right)$$

Eigenvectors for $n = 64$



Performs like low and high frequencies

What does Jacobi do to error?

The error propagation

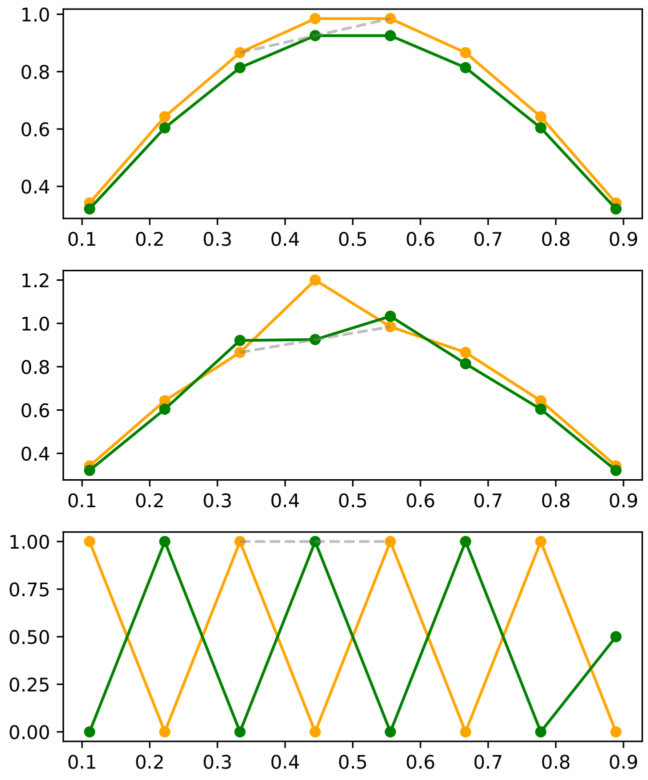
$$e \leftarrow Te \quad T = I - D^{-1}A$$

$$T = \begin{bmatrix} 0 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ & \ddots & \ddots & \ddots & \\ & & 1/2 & 0 & 1/2 \\ & & & 1/2 & 0 \end{bmatrix}$$

It is averaging (like a mean filter)

$$e_i^{\text{new}} \leftarrow \frac{1}{2}\left(e_{i-1}^{\text{old}} + e_{i+1}^{\text{old}}\right)$$

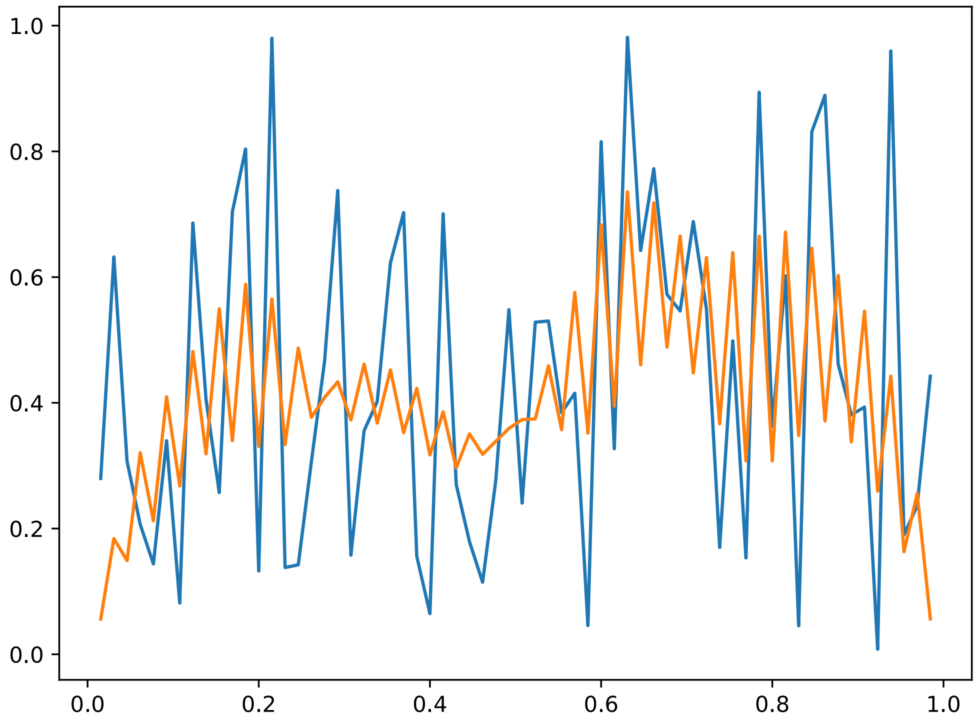
For different types of error



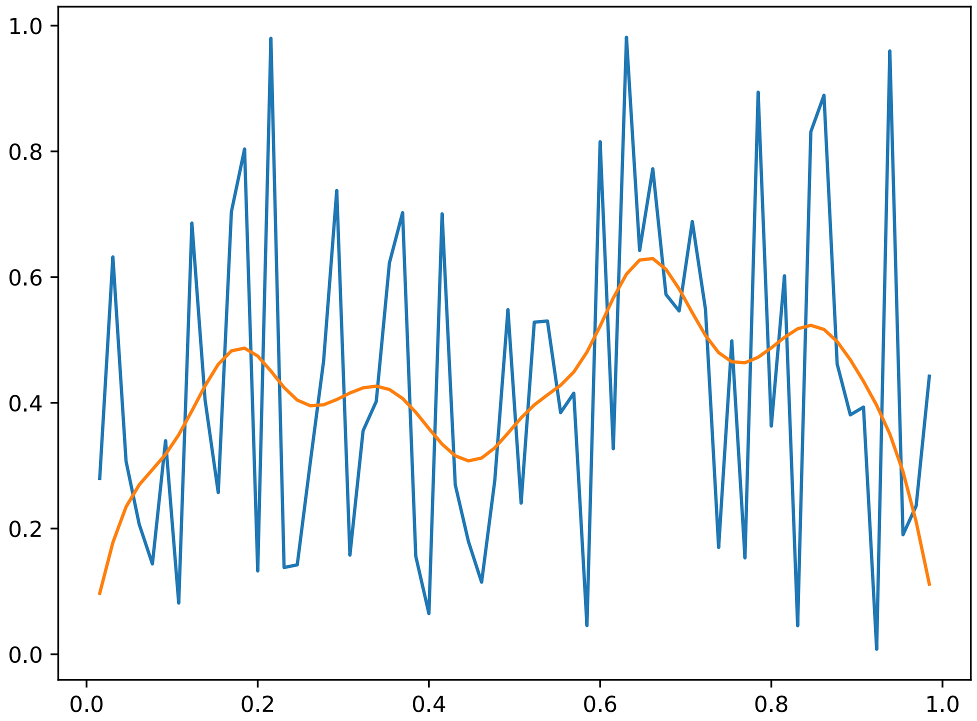
It *averages out* certain frequency quickly

From Jacobi to weighted-Jacobi

$$u \leftarrow u + D^{-1}r$$



$$u \leftarrow u + \omega D^{-1}r, \omega = 2/3$$



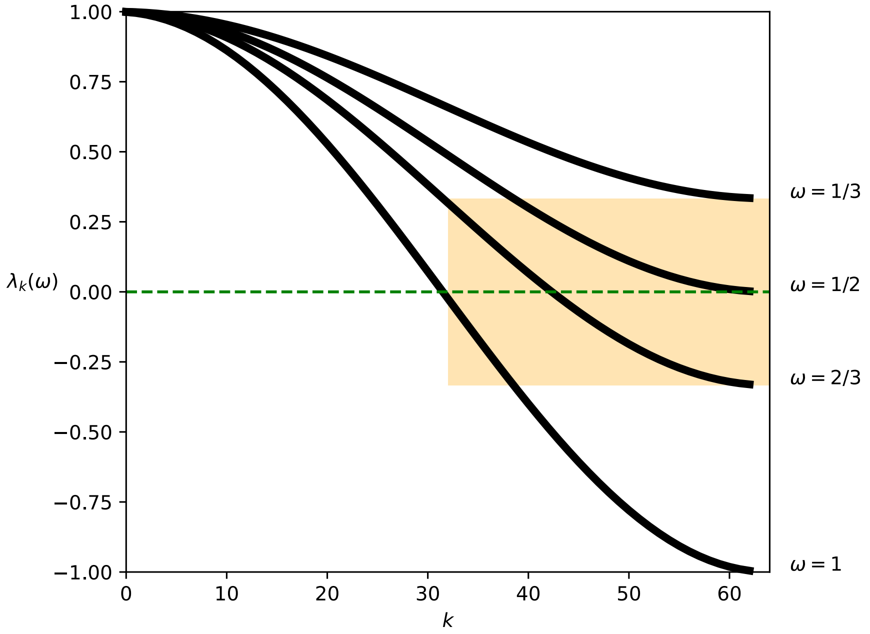
Weighted Jacobi is a better **smoother**.

Fourier Analysis

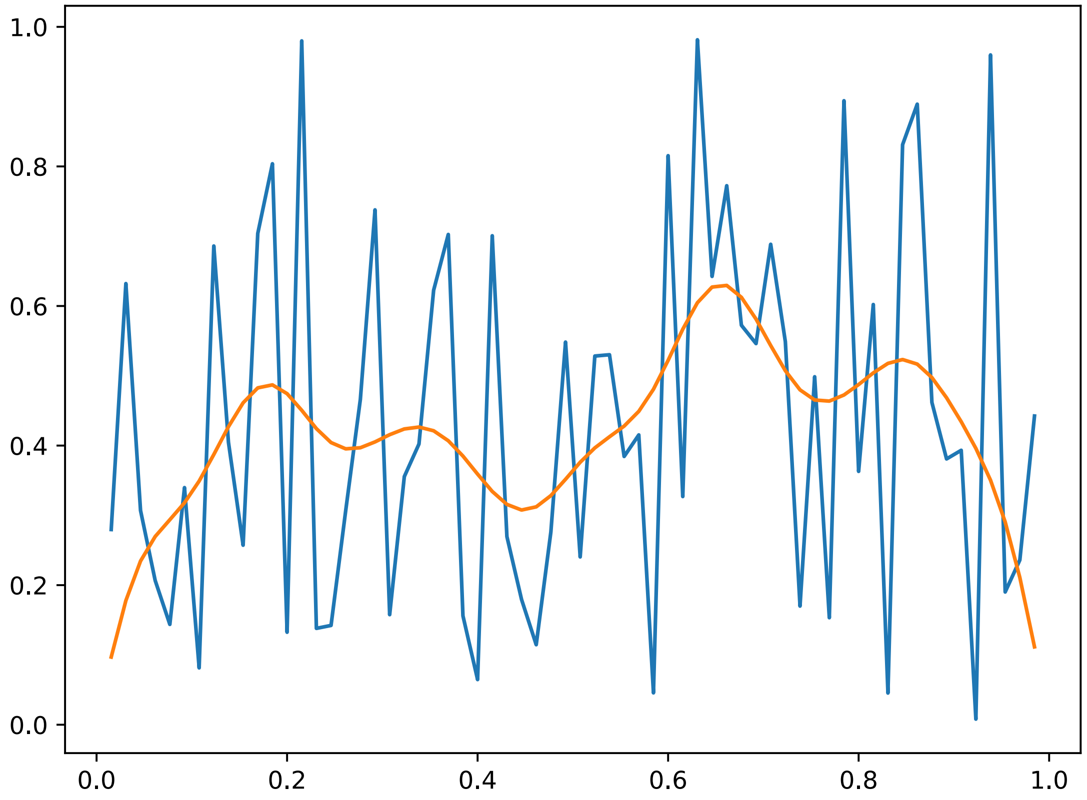
Using the eigenvectors of $T := I - \omega D^{-1} A$ as a basis for the error space

$$e^{(\sigma)} = T^\sigma e^{(0)} = T^\sigma \sum_{k=1}^{n} c_k v_k = \sum_{k=1}^{n} c_k T^\sigma v_k = \sum_{k=1}^{n} c_k \lambda_k^\sigma v_k$$

error on different direction v_k is reduced by different magnitude of λ_k

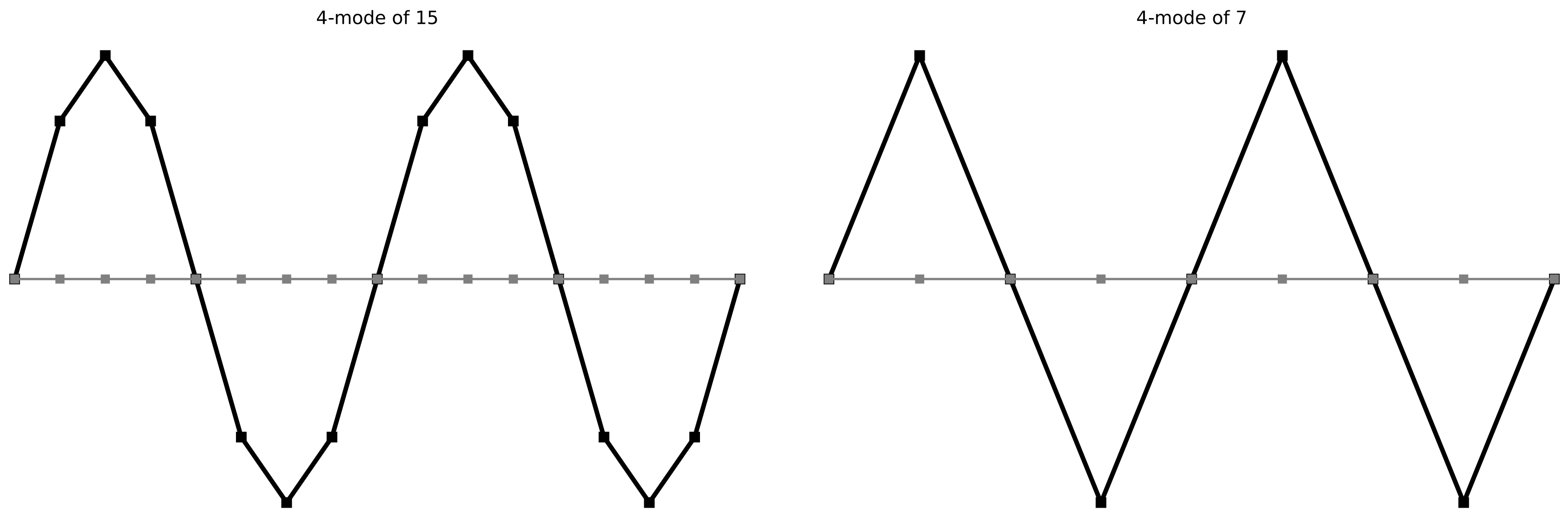


For $\omega = 2/3$, the high frequency part of the error is reduced by (at least) $1/3$



And then? How do we deal with the smooth error.

Sampling on a Coarse Grid



We lost a little bit of information. But

- Problem is smaller: Solve it directly $\Rightarrow$ two-grid method;
- Frequency is higher: Do Jacobi again $\Rightarrow$ multigrid method.

Solve it directly: like a projection method

Look for the "best" update:

$$x^{(1)} \leftarrow x^{(0)} + \Delta x$$

Over the coarse grid

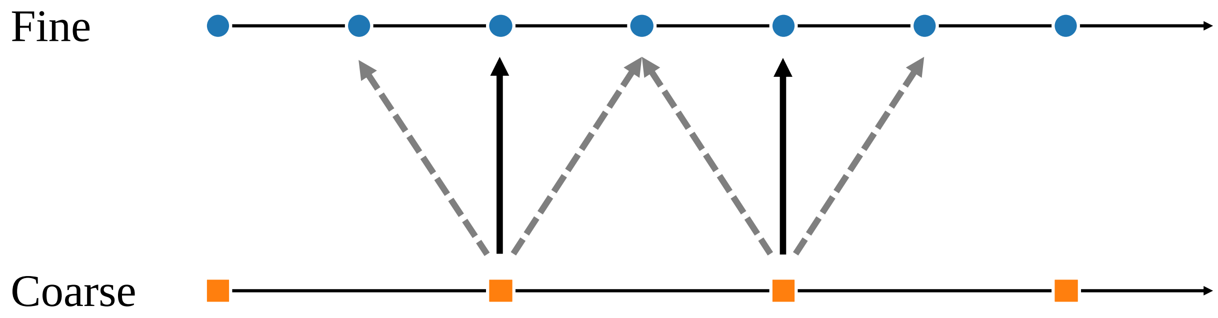
$$\min_{\Delta x \in \operatorname{span}\{V\}} \left\| x^* - x^{(1)} \right\|_A$$

So the update looks like

$$x^{(1)} = x^{(0)} + V \left(V^\top A V \right)^{-1} V^\top r^{(0)}$$

What should be the V here? Inter-grid operations

Prolongation: From Coarse to Fine



An operator

$$P : \Omega^{2h} \to \Omega^h$$

where Ω^{2h} is the coarse grid and Ω^h is the fine grid

Interpolation

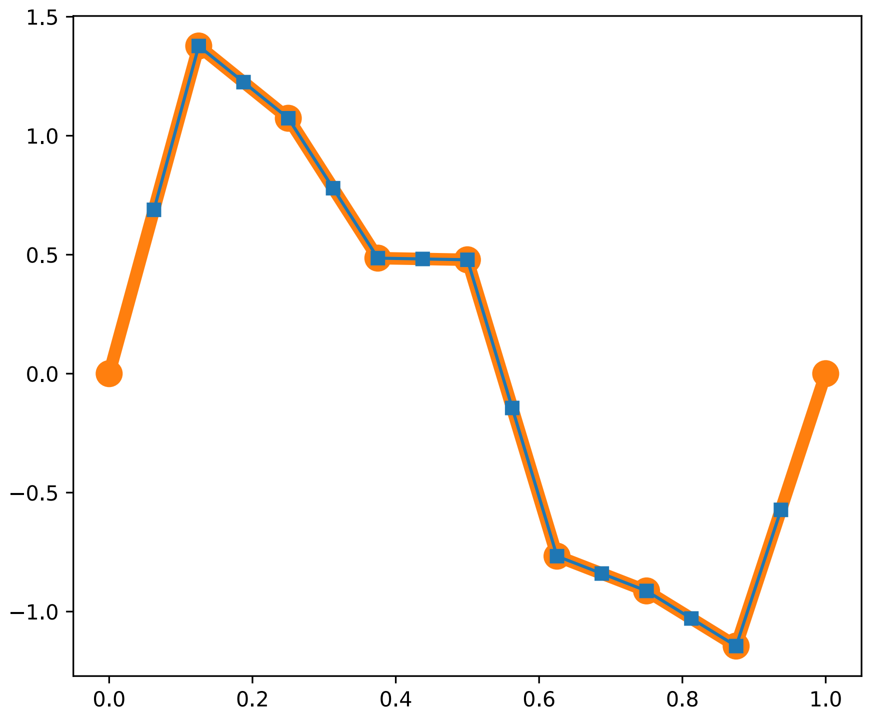
$$v_{2i}^h = v_i^{2h}$$

$$v_{2i+1}^h = \frac{1}{2}\left(v_i^{2h} + v_{i+1}^{2h}\right)$$

Or in matrix form

$$P = \frac{1}{2}\begin{bmatrix} 1 & & & \\ 2 & & & \\ 1 & 1 & & \\ & 2 & & \\ & & \vdots & \\ & & 1 & 1 \\ & & & 2 \\ & & & 1 \end{bmatrix}$$

Looks like



Notice: P is full rank

Two-grid Method

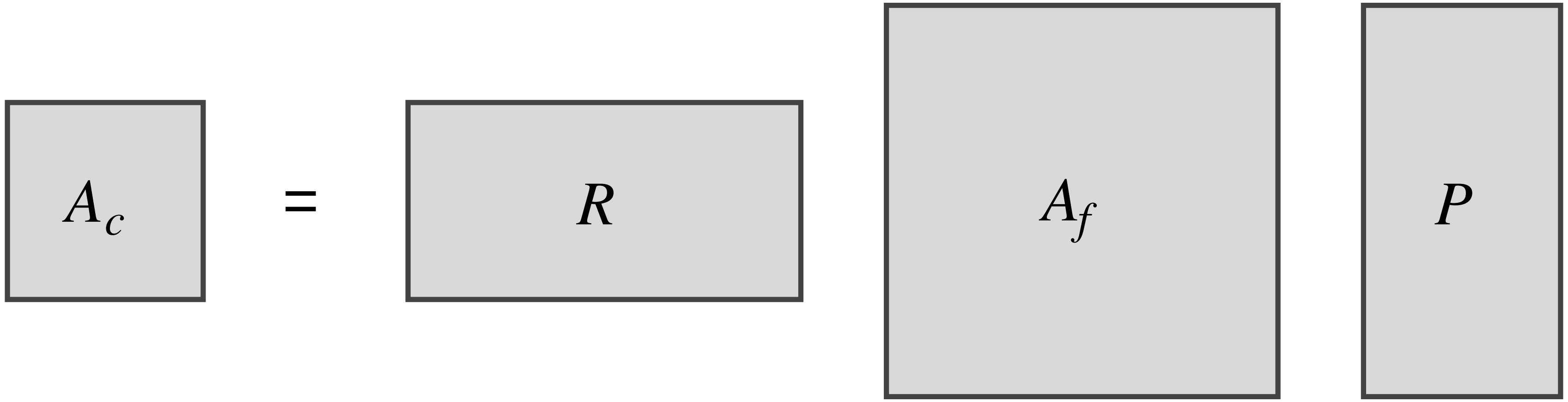
$$x^{(1)} = x^{(0)} + P\left(P^\top AP\right)^{-1} P^\top r^{(0)}$$

$$\begin{array}{lc}
\text{1. Given} & x \\
\text{2. Smooth a few times} & x \leftarrow x + \omega D^{-1}Ar \\
\text{3. Form residual} & r = b - Ax \\
\text{4. Restrict the residual} & P^\top r \\
\text{5. Solve the coarse problem} & P^\top AP\delta = P^\top r \\
\text{6. Interpolate the approx error} & P\delta \\
\text{7. Correct} & x \leftarrow x + P\delta
\end{array}$$

$R = P^\top$ is the **restriction** here

$A_c = P^\top AP$ is the **coarse level operator** here

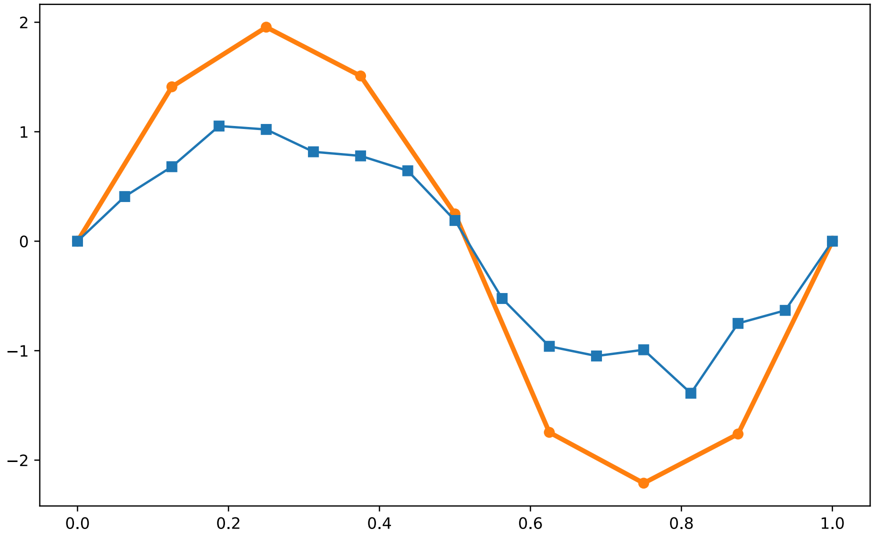
Let's look at the coarse level operator A_c



The restriction R don't have to be $P^\top$

It's called a balance pair if

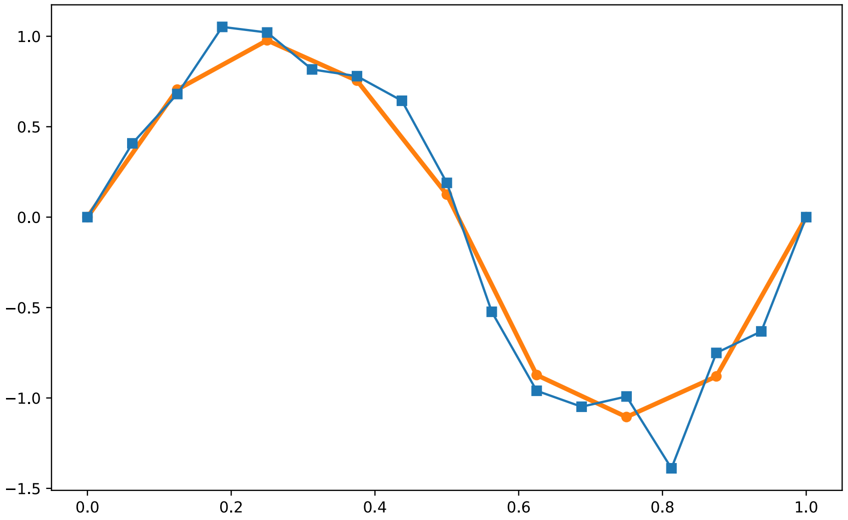
$$R = P^\top = \frac{1}{2}\begin{bmatrix} 1 & 2 & 1 & & & & & & \\ & & 1 & 2 & 1 & & & & \\ & & & & \cdots & \cdots & \cdots & & \\ & & & & & & 1 & 2 & 1 \end{bmatrix}$$



Other options for restriction

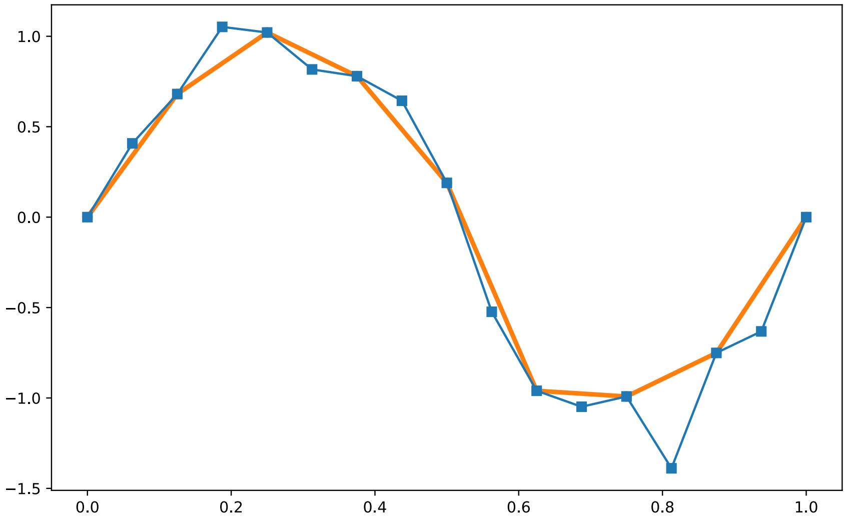
$\frac{1}{2}P^\top$:

$$\frac{1}{4}\begin{bmatrix} 1 & 2 & 1 & & & & & & \\ & & 1 & 2 & 1 & & & & \\ & & & & \cdots & \cdots & \cdots & & \\ & & & & & & 1 & 2 & 1 \end{bmatrix}$$



Injection:

$$\begin{bmatrix} 0 & 1 & 0 & & & & & & \\ & & 0 & 1 & 0 & & & & \\ & & & & \cdots & \cdots & \cdots & & \\ & & & & & & 0 & 1 & 0 \end{bmatrix}$$



each row of R sums to 1

Convergence

Multigrid

$$I - \mathcal{M}_{\mathrm{MG}} A = \left(I - M^\top A\right)^{\nu_{\mathrm{post}}} \left(I - P\left(P^\top A P\right)^{-1} P^\top A\right) \left(I - M A\right)^{\nu_{\mathrm{pre}}}$$

$$\boldsymbol{e} \leftarrow (I - \mathcal{M}_{\mathrm{MG}} A)\boldsymbol{e}$$

We seek for a sharp bound of this which is independent of n.

Smoothing:

$$S \quad \text{or} \quad \boldsymbol{u} \leftarrow S\boldsymbol{u} + (I - S)A^{-1}\boldsymbol{f}$$

Coarse grid correction (CGC)

$$T = I - P\left(P^\top A P\right)^{-1} P^\top A = I - \Pi$$

For 1-D Weighted-Jacobi Smoother

$\mathrm{range}(P)$ is exactly the low-frequency subspace $\mathcal{V}_L$

> See the proof in Practical fourier analysis for multigrid methods §3.1

So, only high-frequency errors left after CGC

$$||ST\boldsymbol{e}|| \leq \frac{1}{3}||\boldsymbol{e}|| \text{ for all } \boldsymbol{e}$$

Check it out: eigen_1d_2grid.ipynb

What kind of sufficient conditions are we looking for?

Relaxation should be effective on the range of coarse grid correction

$$\|ST\boldsymbol{e}\|_A^2 \leq (1-\delta)\|T\boldsymbol{e}\|_A^2 \text{ for all } \boldsymbol{e}$$

No side effects on the range of interpolation

$$\|S\boldsymbol{e}\|_A^2 \leq \|\boldsymbol{e}\|_A^2 \quad \text{for } \boldsymbol{e} \perp \text{range}(T)$$

Combine this two

$$\|Se\|_A^2 \leq \|STe\|_A^2 + \|S(I-T)e\|_A^2 \leq (1-\delta)\|Te\|_A^2 + \|(I-T)e\|_A^2 = \|e\|_A^2 - \delta\|Te\|_A^2$$

Theorem

If there exists $\delta > 0$ (independent of n) such that

$$\|Se\|_A^2 \leq \|e\|_A^2 - \delta\|Te\|_A^2 \quad \text{for all } e,$$

then

$$\|ST\|_A^2 \leq 1 - \delta$$

Proof:

Since $Te = \mathbf{0}$ gives $\|STe\|_A = 0$:

$$\|ST\|_A^2 = \sup_{e:Te\neq 0} \frac{\|STe\|_A^2}{\|e\|_A^2} = \sup_{e:Te\neq 0} \frac{\|STe\|_A^2}{\|Te\|_A^2 + \|(I-T)e\|_A^2}$$

Let $\hat{e}$ be the $\arg\sup$ then $(I - T)\hat{e} = 0$

Proof (Cont.):

$$\|ST\|_A^2 = \sup_{e:Te\neq 0} \frac{\|STe\|_A^2}{\|e\|_A^2} = \sup_{e:Te\neq 0} \frac{\|S(Te+(I-T)e)\|_A^2}{\|Te\|_A^2} = \sup_{e:Te\neq 0} \frac{\|Se\|_A^2}{\|Te\|_A^2}$$

According to the assumption

$$\|Se\|_A^2 \leq \|e\|_A^2 - \delta\|Te\|_A^2 \quad \text{for all } e,$$

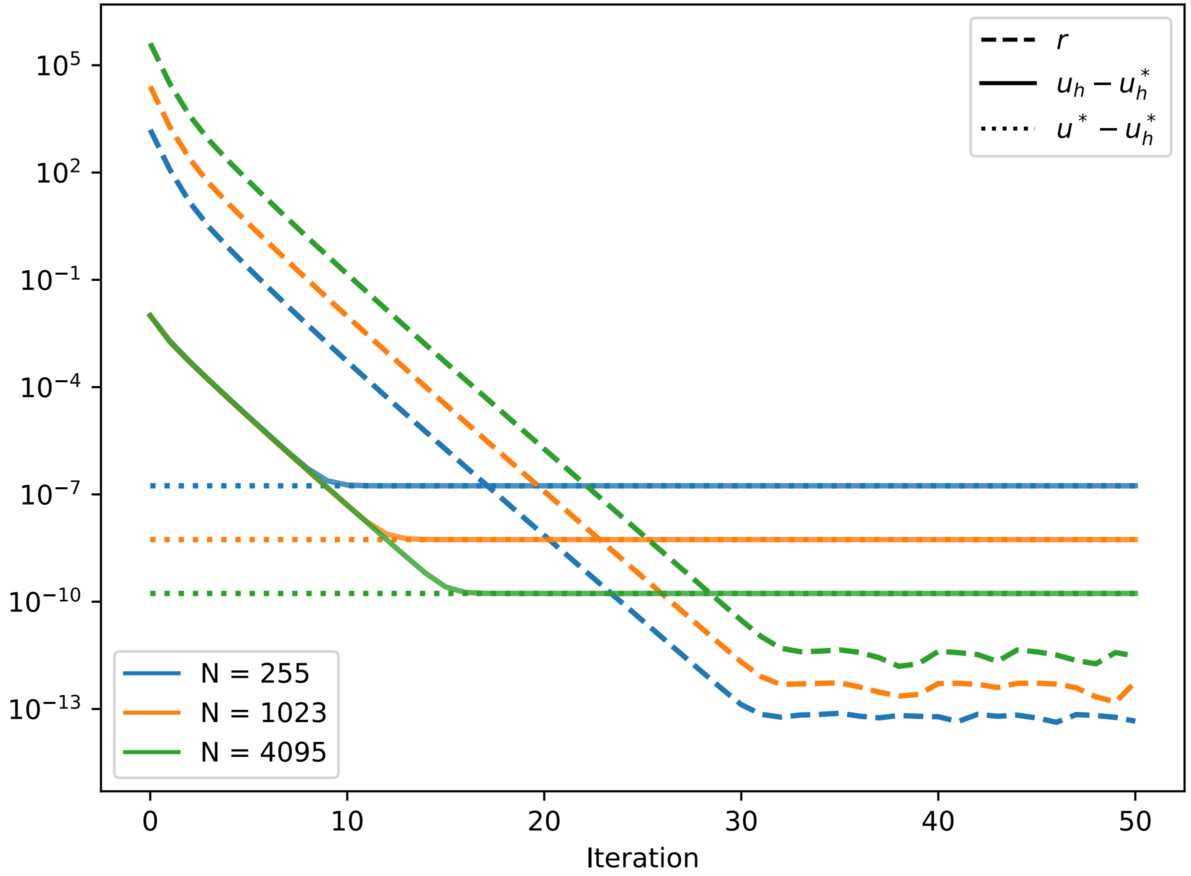
Finally, we get

$$\|ST\|_A^2 = \sup_{e:Te\neq 0} \frac{\|Se\|_A^2}{\|Te\|_A^2} \leq \sup_{e:Te\neq 0} \frac{\|e\|_A^2 - \delta\|Te\|_A^2}{\|Te\|_A^2} = 1 - \delta$$

How Accurate is Multigrid?

The exact solution to the PDE u^* and to the linear system u_L^*

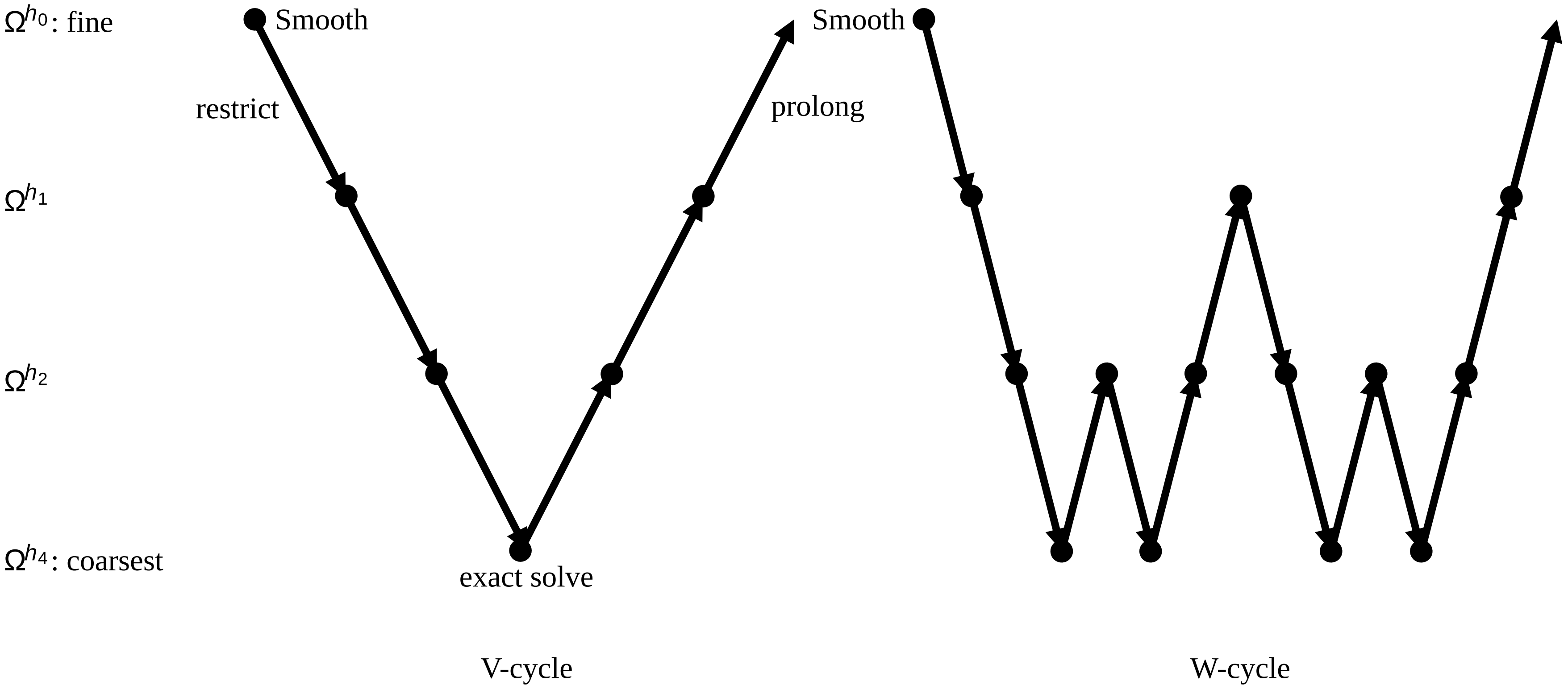
$$-u^{*\prime\prime} = f \quad Au_L^* = b$$



- The convergence rate is independent of N;
- The total error is limited by the discretization error;
- Need $\mathcal{O}(\log N)$ steps to get the limit.

$\text{5. Solve the coarse problem } P^\top A P\delta = P^\top r \quad ?$

Do two-grid recursively



Exact solve? A_{coarest} is small enough

Take-home Messages

- Basic idea

$$\text{Multigrid} \approx \text{Damp the high frequencies} + \text{Correct the low frequencies}$$

- Convergence rate: **Independent of** n
- Computational cost

$$\mathcal{O}\left(N^3\right) \longrightarrow \mathcal{O}(N)$$