# Augmented Subspace Method for Eigenvalue and Nonlinear Problems

谢和虎 (Hehu Xie)

Email: hhxie@lsec.cc.ac.cn

Homepage: lsec.cc.ac.cn/~hhxie

中国科学院数学与系统科学研究院

**暑期课程: 2022 年 6 月 26 日**

# Contents

# Contents

# Contents

# Contents

# Contents

# Outline

# Section 1: Multigrid method

- Solve the linear system: $A_h u_h = f_h$.

## Multigrid method for linear problem

- Current approximation: $u_h^{(k)}$
- Presmoothing: $u_h^{(k+1/3)} = S_h u_h^{(k)}$
- Compute the residual: $r_h = f_h - A_h u_h^{(k+1/3)}$
- Solve the correction equation on the coarse space: $A_H e_H = I_h^H r_h$.
- Do the correction: $u_h^{(k+2/3)} = u_h^{(k+1/3)} + I_H^h e_H$.
- Do the postsmoothing: $u_h^{(k+1)} = S_h u_h^{(k+2/3)}$

- The linearity: $A_h(u_h + \delta_h) = A_h u_h + A_h \delta_h = A_h u_h + A_h I_H^h e_H$.
- Nonlinear problem?

# Multigrid method for nonlinear problems

- Outer iteration (nonlinear iteration) + inner iteration (multigrid method);
- Full Approximation Scheme (FAS);
- Brandt (1977), Hackbusch (1979).

**Brandt (1977): Full Approximation Scheme (FAS)**

- Restrict the fine-grid approximation and its residual: $r_H = I_h^H(f_h - A_h(v_h))$ and $v_H = I_h^H v_h$.
- Solve the coarse-grid problem: $A_H(u_H) = A_H(v_H) + r_H$.
- Compute the coarse-grid approximation to the error: $e_H = u_H - v_H$.
- Interpolate the error approximation to the fine grid and correct the current fine-grid approximation: $v_h \leftarrow v_h + I_H^h e_H$.

# Outline

## Finite element method for Laplace problem

- $$\left\{\begin{array}{rcll} -\Delta u &=& f, & \text{in } \Omega, \\ u &=& 0, & \text{on } \partial\Omega. \end{array}\right.$$

- Variation form
$$(\nabla u, \nabla v) \quad = \quad (f, v), \quad \forall v \in H_0^1(\Omega),$$

- Construct a triangulation $\mathcal{T}_h$ of $\Omega$ and build the linear finite element space
$$V_h = \left\{v_h \in H_0^1(\Omega) : v_h|_K \in \mathcal{P}_1, \quad \forall K \in \mathcal{T}_h\right\}.$$

- Discrete equation
$$(\nabla u_h, \nabla v_h) \quad = \quad (f, v_h), \quad \forall v_h \in V_h.$$

# Error estimate for finite element method

## Error estimates

$$|u-u_h|_1 = \inf_{v_h\in V_h} |u-v_h|_1.$$

When $\Omega$ is convex, $u\in H^2(\Omega)$, $|u-u_h|_1 \leq Ch\|u\|_2$.

## Aubin [1967], Nitsche [1968], Oganesyan-Rukhovets [1969]

Choose $\phi\in L^2(\Omega)$ such that $\|u-u_h\|_0 = (u-u_h,\phi)$,

$$\begin{aligned}\|u-u_h\|_0 &= (u-u_h,\phi) = (\nabla(u-u_h),\nabla\psi)\\ &= (\nabla(u-u_h),\nabla(\psi-v_h)),\quad \forall v_h\in V_h,\end{aligned}$$

where $\psi$ satisfies $(\nabla v,\nabla\psi) = (v,\phi),\ \forall v\in H_0^1(\Omega)$.
Then we have

$$\begin{aligned}\|u-u_h\|_0 &\leq C|u-u_h|_1 \inf_{v_h\in V_h}|\psi-v_h|_1 \leq Ch|u-u_h|_1\\ &\leq Ch^2\|u\|_2.\end{aligned}$$

# Aubin-Nitsche技巧的一个应用

## 校正方法[Q. Lin: RAIRO Numerical Analysis, 16(1) (1982), 39-47]

求解非对称问题

$$(\nabla u, \nabla v) + (\mathbf{b} \cdot \nabla u, v) + (\phi u, v) = (f, v), \quad \forall v \in H_0^1(\Omega).$$

校正方法(线性问题)

- 求解有限元方程: 求$u_h \in V_h$满足
$$(\nabla u_h, \nabla v_h) + (\mathbf{b} \cdot \nabla u_h, v_h) + (\phi u_h, v_h) = (f, v_h), \quad \forall v_h \in V_h.$$
- 定义如下的Laplace 方程: 求$u \in H_0^1(\Omega)$满足
$$(\nabla u^0, \nabla v) = (f, v_h) - (\mathbf{b} \cdot \nabla u_h, v) - (\phi u_h, v), \quad \forall v \in H_0^1(\Omega).$$
- 误差估计: $\|u^0 - u\|_1 \leq C\|u - u_h\|_0 \leq Ch\|u - u_h\|_1$.

- 林群, 应用数学学报(Acta Math. Sinca), 22(2) (1979), 219-230.
- 林群、谢干权, 科学通报, 26(8) (1981), 449-452.

# Two-grid method

## Two-grid method: Lin(1979), Lin-Xie(1981), Xu-Zhou(2001), $\cdots$

1. Solve the eigenvalue problem on an coarse mesh $\mathcal{T}_H$:
   Find $(\lambda_H, u_H) \in \mathcal{R} \times V_H$ such that $\|u_H\|_0 = 1$ and
   $$(\nabla u_H, \nabla v_H) = \lambda_H (u_H, v_H), \quad \forall v_H \in V_H$$
2. Construct a finer finite element space $V_h$ and solve the following symmetric positive definite boundary value problem: Find $u_h \in V_h$ such that
   $$(\nabla u_h, \nabla v_h) \;=\; \lambda_H (u_H, v_h), \quad \forall v_h \in V_h.$$
3. Compute the Rayleigh quotient:
   $$\lambda_h = \frac{(\nabla u_h, \nabla u_h)}{(u_h, u_h)}.$$

Then we have a better eigenpair approximation $(\lambda_h, u_h)$.

# Error estimate of two-grid method

## Error estimates

- $(\bar{\lambda}_h, \bar{u}_h)$ denote the exact finite element solution.
- The original eigenpair approximation $(\lambda_H, u_H)$ ($\Omega$ convex):

$$\|\bar{u}_h - u_H\|_1 \lesssim H, \quad \|\bar{u}_h - u_H\|_0 \lesssim H^2, \quad |\bar{\lambda}_h - \lambda_H| \lesssim H^2.$$

- $V_h$: linear finite element space on a finer mesh $\mathcal{T}_h$ (produced from $\mathcal{T}_H$ with regular refinement)
- The corrected eigenpair approximation $(\lambda_h, u_h)$:

$$\|\bar{u}_h - u_h\|_1 \lesssim H^2 + h, \quad |\bar{\lambda}_h - \lambda_h| \lesssim H^4 + h^2.$$

- The good choice: $h = H^2$.
- But there is no $\|\bar{u}_h - u_h\|_0 \ll \|\bar{u}_h - u_h\|_1$ (No Aubin-Nitsche technique).
- We can not do correction again.
- It is not an eigensolver such that $\|\bar{u}_h - u_h\|_a \to 0$.

# A new version of the duality argument

## A new case for duality argument [Lin-Xie (2010)]

- The finite element function $\widetilde{u}_h$ has no Aubin-Nitsche estimate.
- We construct a low dimensional space $V_H$ ($V_H \subset V_h$) and build the finite element space $V_{H,h} = V_H + \mathrm{span}\{\widetilde{u}_h\}$. Then solve the following small equation: Find $u_h \in V_{H,h}$ such that
$$(\nabla u_h, \nabla v_{H,h}) = (f, v_{H,h}), \quad \forall v_{H,h} \in V_{H,h}.$$
- Accuracy
$$|u - u_h|_1 = \inf_{v_{H,h} \in V_{H,h}} |u - v_{H,h}|_1 \leq |u - \widetilde{u}_h|_1.$$
- Duality argument
$$\begin{aligned} \|u - u_h\|_0 &= (u - u_h, \phi) = (\nabla(u - u_h), \nabla\psi) \\ &= (\nabla(u - u_h), \nabla(\psi - v_H)). \end{aligned}$$
Then we have
$$\|u - u_h\|_0 \leq |u - u_h|_1 \inf_{v_H \in V_{H,h}} |\psi - v_H|_1 \leq CH\|u - u_h\|_1.$$

# A new version of the duality argument

## Linear vs nonlinear

- The function in ths space $V_{H,h} = V_H + \mathrm{span}\{\widetilde{u}_h\}$ can be denoted by $u_h = u_H + \alpha\widetilde{u}_h$, where $u_H \in V_H$ and $\alpha \in \mathbb{R}$.
- If $F$ is linear, $F(u_H + \alpha\widetilde{u}_h) = F(u_H) + \alpha F(\widetilde{u}_h)$. The normal multigrid method for linear problem with the choice of $\alpha = 1$ and recursive iteration. Then we do not need to define the space $V_{H,h}$.
- When $F$ is nonlinear, $F(u_H + \alpha\widetilde{u}_h) \neq F(u_H) + \alpha F(\widetilde{u}_h)$. We should do the nonlinear solving process in the space $V_{H,h}$.
- Since $\mathrm{dim}V_{H,h} = \mathrm{dim}V_H + 1$, the algebraic scale is very small.
- Different from the normal way: Outer iteration (nonlinear iteration) + Inner iteration (multigrid method).

# Aubin-Nitsche新形式的应用

## 非对称问题

$$(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) + (\phi u_h, v_h) = (f, v_h), \quad \forall v_h \in V_h.$$

## 一个校正方法[Lin-Xie (2010)]

假设我们有一个逼近$\widetilde{u}_h \in V_h$

- 求解如下方程: 求$\widehat{u}_h \in V_{H,h}$使得

$$(\nabla \widehat{u}_h, \nabla v_{H,h}) + (\mathbf{b}\cdot\nabla \widehat{u}_h, v_{H,h}) + (\phi \widehat{u}_h, v_{H,h}) = (f, v_{H,h}), \ \forall v_{H,h} \in V_{H,h}.$$

  误差估计

$$\|\widehat{u}_h - u_h\|_1 \le C\|\widetilde{u}_h - u_h\|_1, \quad \|\widehat{u}_h - u_h\|_0 \le CH\|\widehat{u}_h - u_h\|_1.$$

- 求解如下的Laplace方程

$$(\nabla \bar{u}_h, \nabla v_h) = (f, v_h) - (\mathbf{b}\cdot\nabla \widehat{u}_h, v_h) - (\phi \widehat{u}_h, v_h), \quad \forall v_h \in V_h.$$

  误差估计

$$\|\bar{u}_h - u_h\|_1 \le C\|\widehat{u}_h - u_h\|_0 \le CH\|\widehat{u}_h - u_h\|_1 \le CH\|\widetilde{u}_h - u_h\|_1.$$

# Outline

# Section 3: Augmented subspace method for semilinear problem

## Semilinear elliptic equation

Find $u$ such that

$$\left\{\begin{array}{rcll} -\nabla\cdot(\mathcal{A}\nabla u)+f(x,u) &=& g, & \text{in } \Omega, \\ u &=& 0, & \text{on } \partial\Omega, \end{array}\right.$$

- where $\mathcal{A}=(a_{i,j})_{d\times d}$ is a positive symmetric definite matrix with $a_{i,j}\in W^{1,\infty}\ \ (i,j=1,2,\cdots,d)$,
- $f(x,u)$ is a nonlinear function corresponding to the second variable.

## Variational form

- The weak form can be described as: Find $u\in V$ such that

$$a(u,v)+(f(x,u),v)=(g,v),\quad \forall v\in V,$$

where $V:=H_0^1(\Omega)$ and $a(u,v)=(\mathcal{A}\nabla u,\nabla v)$.

# Semilinear elliptic problem

## Some settings

- Obviously, $a(u,v)$ is bounded and coercive on $V$, i.e.,

$$a(u,v) \leq C_a \|u\|_{1,\Omega}\|v\|_{1,\Omega} \quad \text{and} \quad c_a\|u\|_{1,\Omega}^2 \leq a(u,u), \quad \forall u, v \in V.$$

- Then we use the norm $\|w\|_a := \sqrt{a(w,w)}$ for any $w \in V$ to replace the standard norm $\|\cdot\|_1$.
- Poincaré inequality

$$\|w\|_0 \leq C_p \|w\|_a, \qquad \forall w \in V,$$

where the constant $C_p$ depends on $\Omega$.

## Assumption for nonlinear function

The nonlinear function $f(x,\cdot)$ satisfies the following assumptions

$$(f(x,w) - f(x,v), w - v) \geq 0, \qquad \forall w,\ v \in V := H_0^1(\Omega),$$

$$(f(x,w) - f(x,v), \phi) \leq C_f \|w - v\|_0 \|\phi\|_1, \quad \forall w,\ v,\ \phi \in V.$$

# Finite element method

- We generate a regular triangulation $\mathcal{T}_h$ and build the simplest linear finite element space $V_h$ on $\mathcal{T}_h$.
- The standard finite element scheme is: Find $\bar{u}_h \in V_h$ such that

$$a(\bar{u}_h, v_h) + (f(x, \bar{u}_h), v_h) = (g, v_h), \quad \forall v_h \in V_h.$$

- Denote a linearized operator $L : H_0^1(\Omega) \to H^{-1}(\Omega)$ by:

$$(Lw, v) = (\mathcal{A}\nabla w, \nabla v), \quad \forall w, v \in V.$$

- In order to measure the error for the finite element approximations, we denote

$$\delta_h(u) = \inf_{v_h \in V_h} \|u - v_h\|_a.$$

- For the global prior error estimates, we introduce $\eta_a(V_h)$ as follows:

$$\eta_a(V_h) = \sup_{f \in L^2(\Omega), \|f\|_0 = 1} \inf_{v_h \in V_h} \|L^{-1}f - v_h\|_a.$$

**Lemma (Error estimate: Xie-Xie-Xu)**

*When Assumption A is satisfied, the semilinear equation is uniquely solvable and the following estimates hold*

$$\begin{aligned}\|u-\bar{u}_h\|_a &\leq (1+C_f\eta_a(V_h))\delta_h(u),\\ \|u-\bar{u}_h\|_0 &\leq (1+C_fC_p)\eta_a(V_h)\|u-\bar{u}_h\|_a.\end{aligned}$$

# Aubin-Nitsche技巧的应用

## 两网格方法: $f(\cdot,\cdot)$是非线性项

1. 首先在粗网格 $\mathcal{T}_H$ 上求解如下的半线性问题: 求 $u_H \in V_H$ 使其满足
$$a(u_H, v_H) + \big(f(x, u_H), v_H\big) = (g, v_H), \quad \forall v_H \in V_H.$$
2. 对粗网格 $\mathcal{T}_H$ 进行加密以得到更细的网格 $\mathcal{T}_h$, 并在此细网格上求解如下的线性问题: 求 $u_h \in V_h$ 满足
$$a(u_h, v_h) = -\big(f(x, u_H), v_h\big) + (g, v_h), \quad \forall v_h \in V_h.$$

## 两网格方法的特点

- 两网个方法的收敛速度 ($\bar{u}_h$表示精确有限元解):
$$|\bar{u}_h - u_h|_1 \leq (1 + C_f C_p)(1 + C_f H) C_f H \inf_{v_H \in V_H} |u - v_H|_1.$$
- 但$u_h$不满足$\|u - u_h\|_0 \leq CH|u - u_h|_1$, 只能做一次校正.
- 若要求代数误差$|\bar{u}_h - u_h|_1$不超过离散误差, 需要根据细网格尺寸 $h$ 来确定粗网格尺寸 $H$. 反过来, 当 $H$ 选定的时候, $h$ 也就被限制了.

# Augmented subspace method

Assume we have obtained an approximate solution $u_h^{(\ell)} \in V_h$.

1. Define the following auxiliary boundary value problem: Find $\widehat{u}_h^{(\ell+1)} \in V_{h_k}$ such that
$$a(\widehat{u}_h^{(\ell+1)}, v_h) = -(f(x, u_h^{(\ell)}), v_h) + (g, v_h), \quad \forall v_h \in V_h.$$
Solve this linear boundary value problem with multigrid method
$$\|\widetilde{u}_h^{(\ell+1)} - \widehat{u}_h^{(\ell+1)}\|_a \leq \theta \|u_h^{(\ell)} - \widehat{u}_h^{(\ell+1)}\|_a, \quad \theta < 1.$$
2. Define a finite element space $V_{H,h} := V_H + \mathrm{span}\{\widetilde{u}_h^{(\ell+1)}\}$ and solve the following semilinear equation: Find $u_h^{(\ell+1)} \in V_{H,h}$ such that
$$a(u_h^{(\ell+1)}, v_{H,h}) + \big(f(x, u_h^{(\ell+1)}), v_{H,h}\big) = (g, v_{H,h}), \ \forall v_{H,h} \in V_{H,h}.$$

In order to simplify the notation and summarize above two steps, we define
$$u_h^{(\ell+1)} = \texttt{SemilinearAug}(V_H, u_h^{(\ell)}, V_h).$$

# Error estimate

## Theorem (Error estimate)

假设存在常数 $C_1$ 使得给定解 $u_h^{(\ell)}$ 有如下误差估计

$$\|\bar{u}_h - u_h^{(\ell)}\|_0 \leq C_1\eta_a(V_H)\|\bar{u}_h - u_h^{(\ell)}\|_a.$$

则执行单步校正算法之后, 所得近似解 $u_h^{(\ell+1)}$ 有如下误差估计

$$\begin{aligned}
\|\bar{u}_h - u_h^{(\ell+1)}\|_a &\leq \gamma\|\bar{u}_h - u_h^{(\ell)}\|_a, \\
\|\bar{u}_h - u_h^{(\ell+1)}\|_0 &\leq (1 + C_pC_f)\eta_a(V_H)\|\bar{u}_h - u_h^{(\ell+1)}\|_a,
\end{aligned}$$

其中 $\gamma := \big(\theta + (1+\theta)C_1C_f\eta_a(V_H)\big)\big(1 + C_f\eta_a(V_H)\big)$.

- Enhanced error estimates: $\theta = 0$ (精确求解线性方程), $\gamma := C_1C_f\big(1 + C_f\eta_a(V_H)\big)\eta_a^2(V_H)$.

# Multigrid method for semilinear problem

- We first generate a coarse mesh $\mathcal{T}_H$ with the mesh size $H$ and define the linear finite element space $V_H$ on $\mathcal{T}_H$.
- The mesh $\mathcal{T}_{h_1}$ is produced from $\mathcal{T}_H$ by regular refinements. The sequence of meshes $\mathcal{T}_{h_k}$ $(k = 1, \cdots, n)$ are produced by regular refinements such that

$$h_k = \frac{1}{\beta} h_{k-1}, \qquad k = 2, \cdots, n,$$

where $\beta > 1$ (always equals $2$).
- Based on this sequence of meshes, we construct the corresponding nested linear finite element spaces

$$V_H \subseteq V_{h_1} \subset V_{h_2} \subset \cdots \subset V_{h_n}.$$

- The sequence of finite element spaces satisfied the following property

$$\eta_a(V_{h_k}) \approx \frac{1}{\beta} \eta_a(V_{h_{k-1}}), \qquad \delta_{h_k}(u) \approx \frac{1}{\beta} \delta_{h_{k-1}}(u), \quad k = 2, \cdots, n.$$

# 多重网格算法

## Algorithm (完全多重网格法)

1. 在 $V_{h_1}$ 中求解如下半线性问题: 求 $u_{h_1}\in V_{h_1}$, 使得

$$a(u_{h_1},v_{h_1})+(f(x,u_{h_1}),v_{h_1})=(g,v_{h_1}),\quad \forall v_{h_1}\in V_{h_1}.$$

2. 对 $k=2,\cdots,n$, 进行如下迭代:
   1. 令 $u_{h_k}^{(0)}=u_{h_{k-1}}$.
   2. 对 $\ell=0,\cdots,p-1$, 进行如下迭代

   $$u_{h_k}^{(\ell+1)}=\texttt{SemilinearAug}(V_H,u_{h_k}^{(\ell)},V_{h_k}).$$

   3. 定义 $u_{h_k}=u_{h_k}^{(p)}$.

最后得到一个在最细层空间内的近似解 $u_{h_n}\in V_{h_n}$.

# 完全多重网格算法的误差估计

**Theorem (误差估计)**

在粗网格尺寸 $H$ 足够小使得条件 $\gamma^p\beta < 1$ 成立时, 执行由完全多重网格算法得到的近似解 $u_{h_n}$ 有如下误差估计:

$$\begin{aligned}
\|\bar{u}_{h_n} - u_{h_n}\|_a &\leq \frac{2\gamma^p\beta}{1-\gamma^p\beta}\delta_{h_n}(u), \\
\|\bar{u}_{h_n} - u_{h_n}\|_0 &\leq (1+C_pC_f)\eta_a(V_H)\|\bar{u}_{h_n} - u_{h_n}\|_a,
\end{aligned}$$

其中 $\gamma = \big(\theta + (1+\theta)(1+C_pC_f)C_f\eta_a(V_H)\big)\big(1+C_f\eta_a(V_H)\big)$.

- 从上面的误差估计可以知道增大 $p$ (增加单步校正的次数)来提高逼近解的代数精度 $\|\bar{u}_{h_n} - u_{h_n}\|_a$
- 即这里的算法可以使得代数误差达到机器精度.
- 这与多重网格算法求解线性边值问题的性质一样.

# 完全多重网格算法的误差估计

## Corollary (代数误差+离散误差)

由多重网格算法得到的最后近似解 $u_{h_n}$, 有如下估计

$$\|u-u_{h_n}\|_a \leq \left(1+C_f\eta_a(V_{h_n})+\frac{2\gamma^p\beta}{1-\gamma^p\beta}\right)\delta_{h_n}(u),$$

$$\|u-u_{h_n}\|_0 \leq \left((1+C_fC_p)\eta_a(V_{h_n})+(1+C_pC_f)\frac{2\gamma^p\beta}{1-\gamma^p\beta}\eta_a(V_H)\right)\times \delta_{h_n}(u).$$

# Estimate of computational work

- First, we define the dimension of each level finite element space as $N_k := \text{dim}V_{h_k}$. Then
$$N_k \approx \Big(\frac{1}{\beta}\Big)^{d(n-k)} N_n, \quad k = 1, 2, \cdots, n.$$
- We need to solve a semilinear elliptic problem by some type of nonlinear iteration in the low dimensional subspace $V_{H,h_{k+1}}$.
- In each nonlinear iteration step, it is required to assemble the matrix on the finite element space $V_{H,h_k}$ $(k = 2, \cdots, n)$ which needs the computational work $\mathcal{O}(N_k)$.
- Fortunately, the matrix assembling can be carried out by the parallel way easily in the finite element space since it has no data transfer.

# Estimate of the computational work

## Computational work

- We use $\theta$ computing-nodes, nonlinear iteration times: $\varpi$.
- Semilinear elliptic solving in the coarse spaces $V_{H,h_k}$ $(k=2,\cdots,n)$ and $V_{h_1}$ need work $\mathcal{O}(M_H)$ and $\mathcal{O}(M_{h_1})$.
- Linear boundary value problem solving work $\mathcal{O}(N_k)$ $(k=2,\cdots,n)$.
- The computational work in each computing node has the following estimate

$$\text{Total work} = \mathcal{O}\left(\left(1+\frac{\varpi}{\theta}\right)N_n + M_H \log N_n + M_{h_1}\right).$$

# An efficient implementation method

**多项式形式的非线性问题**

求 $u$ 满足如下的方程

$$\left\{\begin{array}{rcll} -\Delta u + wu + \zeta u^3 &=& g, & \text{in } \Omega, \\ u &=& 0, & \text{on } \partial\Omega, \end{array}\right. \tag{3.1}$$

其中 $w \geq 0$ 和 $\zeta > 0$.

- 在单步校正算法的第二步中需要求解如下的半线性方程, 即为求解如下的非线性方程: 求 $u_h^{(\ell+1)} \in V_{H,h}$ 使得

$$(\nabla u_h^{(\ell+1)}, \nabla v_{H,h}) + (wu_h^{(\ell+1)}, v_{H,h}) + (\zeta(u_h^{(\ell+1)})^3, v_{H,h}) = (g, v_{H,h}), \quad \forall v_{H,h} \in V_{H,h}.$$

- 利用非线性迭代算法(如不动点迭代), 在每次非线性迭代的时候需要求解线性边值问题.

# An efficient implementation method

- Working space: $V_{H,h} = V_H + \mathrm{span}\{\widetilde{u}_h\}$,
  $V_H = \mathrm{span}\{\phi_{1,H}, \cdots, \phi_{N_H,H}\}$.
- The function in $V_{H,h}$ is denoted by $u_H + \alpha\widetilde{u}_h = \sum_{k=1}^{N_H} u_k\phi_{k,H} + \alpha\widetilde{u}_h$.
- The matrix version of the nonlinear eigenvalue problem:

$$\begin{pmatrix} A_H & b_{Hh} \\ b_{Hh}^T & \xi \end{pmatrix} \begin{pmatrix} \mathbf{u}_H \\ \alpha \end{pmatrix} = \begin{pmatrix} c_H \\ c_h \end{pmatrix},$$

  where $\mathbf{u}_H \in \mathbb{R}^{N_H}$ and $\alpha \in \mathbb{R}$.
- The matrix $A_H$ has the following expansion

$$\begin{aligned} (A_H)_{i,j} &= \int_\Omega \nabla\phi_{i,H}\nabla\phi_{j,H}d\Omega + \int_\Omega w\phi_{i,H}\phi_{j,H}d\Omega \\ &\quad + \int_\Omega \zeta(u_H + \alpha\widetilde{u}_h)^2\phi_{i,H}\phi_{j,H}d\Omega \\ &:= (A_{H,1})_{i,j} + (A_{H,2})_{i,j}, \end{aligned}$$

  where the linear part

$$(A_{H,1})_{i,j} = \int_\Omega \nabla\phi_{i,H}\nabla\phi_{j,H}d\Omega + \int_\Omega w\phi_{i,H}\phi_{j,H}d\Omega.$$

# An efficient implementation method

- $A_{H,2}$ has the following expansion

$$\begin{aligned}(A_{H,2})_{i,j} &= \int_\Omega \zeta(u_H+\alpha\widetilde{u}_h)^2\phi_{i,H}\phi_{j,H}d\Omega\\ &= \int_\Omega \zeta(u_H)^2\phi_{i,H}\phi_{j,H}d\Omega + 2\alpha\int_\Omega \zeta\widetilde{u}_h u_H\phi_{i,H}\phi_{j,H}d\Omega\\ &\quad +\alpha^2\int_\Omega \zeta(\widetilde{u}_h)^2\phi_{i,H}\phi_{j,H}d\Omega\\ &:= (A_{H,2,1})_{i,j}+2\alpha(A_{H,2,2})_{i,j}+\alpha^2(A_{H,2,3})_{i,j}.\end{aligned}$$

- It is obvious that the computational work for $A_{H,2,1}$ is $\mathcal{O}(N_H)$.
- Assembling $A_{H,2,3}$ need work $\mathcal{O}(N_h)$.
- But $A_{H,2,3}$ it will not change during the nonlinear iteration process.
- The problem is how to compute $A_{H,2,2}$.

# The application of tensor

- Assume $u_H = \sum_{k=1}^{N_H} u_k \phi_{k,H}$, where $N_H := \dim V_H$ and $\{\phi_{k,H}\}_{1\le k\le N_H}$ denotes the Lagrange basis functions for $V_H$.
- The matrix $A_{H,2,2}$ has the following expansion

$$(A_{H,2,2})_{i,j} = \sum_{k=1}^{N_H} u_k \int_\Omega \zeta \widetilde{u}_h \phi_{k,H}\phi_{i,H}\phi_{j,H} d\Omega.$$

- The expansion gives a hint to define a tensor $T_H$ as follows

$$(T_H)_{i,j,k} = \int_\Omega \zeta \widetilde{u}_h \phi_{k,H}\phi_{i,H}\phi_{j,H} d\Omega.$$

- Then the matrix $A_{H,2,2}$ has the following computational scheme

$$A_{H,2,2} = T_H \cdot \mathbf{u}_H,$$

where $\mathbf{u}_H = [u_1, \cdots, u_{N_H}]^T$.

# Implementation method with the tensor

- The number of nonzero elements for the sparse tensor $T_H$ is $\mathcal{O}(N_H)$.
- The computational work for the multiplication of the tensor $T_H$ and the vector $\mathbf{u}_H$ is $\mathcal{O}(N_H)$.
- The assembling for the Tensor $T_H$ needs computational work $\mathcal{O}(N_h)$.
- The tensor will not change during the nonlinear iteration.
- The computational work for each nonlinear iteration step is only $\mathcal{O}(M_H)$.
- Assume there needs $\varpi$ nonlinear iteration times.
- Then the computational work for the multigrid method is only $\mathcal{O}(N_h + \varpi M_H)$. (The asymptotic computational work is absolute optimal).

# Numerical results

在每次单步校正中进行1次的MG迭代，每次MG迭代的时候进行2次CG迭代的前后光滑

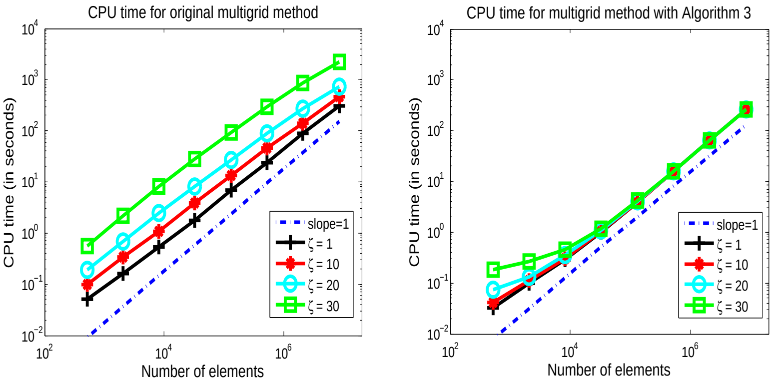


Figure: 左图为原始多重网格方法的CPU时间, 右图为高效多重网格方法的CPU时间.

# Outline

## Eigenvalue problems: 代数特征值、PDE、有限元方法、算子逼近

- Eigenvalue problem: Find $u \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}$ such that
$$Au = \lambda u,$$
where $A \in \mathbb{R}^{n\times n}$ is a positive symmetric definite matrix.
- Second order elliptic problem: Find $(\lambda, u)$ such that
$$\left\{\begin{array}{rcll} -\Delta u &=& \lambda u, & \text{in } \Omega, \\ u &=& 0, & \text{on } \partial\Omega. \end{array}\right.$$
- Eigenvalue dependent nonlinear eigenvalue problem: Find $(\lambda, u)$ such that
$$\left\{\begin{array}{rcll} \Delta^2 u + \lambda\Delta u &=& \lambda^2 u, & \text{in } \Omega, \\ u &=& 0, & \text{on } \partial\Omega. \end{array}\right.$$
- Eigenfunction dependent nonlinear eigenvalue problem: Find $(\lambda, u)$ such that
$$\left\{\begin{array}{rcll} -\Delta u + N(u) &=& \lambda u, & \text{in } \Omega, \\ u &=& 0, & \text{on } \partial\Omega. \end{array}\right.$$
- Large-scale eigenvalue problems by High Performance Computers.

# Numerical methods for algebraic eigenvalue problem

## Numerical methods for eigenvalue problem

- 幂法: $x^{(\ell+1)} = Ax^{(\ell)}$, 反幂法: $Ax^{(\ell+1)} = x^{(\ell)}$
- Lanczos (symmetric matrix) and Arnoldi (nonsymmetric matrix) methods: Krylov subspace: $\mathrm{span}\{x, Ax, \cdots, A^m x\}$
- Symmetric positive definite: LOBPCG, Optimization methods, ...
- Rayleigh商迭代: Jacobi-Davidson 迭代算法
- Integral type methods (复数运算, 求解几乎奇异线性方程)

## 特征值软件包

- 目前主流解法器: 稠密矩阵: LAPACK, ELPA 等;
  稀疏矩阵: ARPACK, SLEPc, Hypre 等

# Augmented subspace for eigenvalue problem

## Laplace eigenvalue problem

- Laplace 特征值问题:
$$\begin{cases} -\Delta u &= \lambda u \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega. \end{cases}$$
- Weak form: Find $(\lambda, u) \in \mathcal{R} \times V := H_0^1(\Omega)$ such that $\|u\|_a = 1$ and
$$a(u,v) = \lambda b(u,v) \quad \forall v \in V,$$
where
$$a(u,v) = \int_\Omega \nabla u \nabla v d\Omega, \quad b(u,v) = \int_\Omega uv d\Omega.$$
- 定义能量范数:$\|w\|_a = \sqrt{a(w,w)}$, $L^2$范数:$\|w\|_b = \sqrt{b(w,w)}$, $\forall w \in V$.
- 代数特征值问题: 求$(\bar{\lambda}_h, \bar{u}_h) \in \mathcal{R} \times V_h$使得$a(\bar{u}_h, \bar{u}_h) = 1$ 且
$$a(\bar{u}_h, v_h) = \bar{\lambda}_h b(\bar{u}_h, v_h), \qquad \forall v_h \in V_h.$$
- 代数形式: $A\bar{u}_h = \bar{\lambda}_h M \bar{u}_h$.

# 特征值问题的有限元空间逼近: $V_h \subset V$

## Error estimates: $\mathcal{K} = V_h$

Assume the eigenpair approximation $(\bar{\lambda}_h, \bar{u}_h)$ has the property that $\bar{\mu}_h = 1/\bar{\lambda}_h$ is closest to the exact $\mu = 1/\lambda$. The corresponding spectral projectors $E_h : V \mapsto \mathrm{span}\{\bar{u}_h\}$ is defined as follows

$$a(E_h w, \bar{u}_h) = a(w, \bar{u}_h), \quad \forall w \in V.$$

Then the following error estimates hold

$$\begin{aligned}
\|u - E_h u\|_a &\leq \sqrt{1 + \frac{\bar{\mu}_{1,h}}{\delta^2_{\lambda,h}} \eta_a(V_h)} \|(I - \mathcal{P}_{\mathcal{K}})u\|_a, \\
\|u - E_h u\|_b &\leq \left(1 + \frac{\bar{\mu}_{1,h}}{\delta_{\lambda,h}}\right) \eta_a(V_h) \|u - E_{i,h} u\|_a,
\end{aligned}$$

where $\delta_{\lambda,h}$ is defined as follows

$$\delta_{\lambda,h} := \min_{\lambda_{j,h} \neq \lambda_h} |\mu_{j,h} - \mu| = \min_{\lambda_{j,h} \neq \lambda_h} \left| \frac{1}{\lambda_{j,h}} - \frac{1}{\lambda} \right|.$$

# 有限元空间的逼近性

## 有限元空间的逼近性质

对于有限元空间$V_h$, 有如下的估计

$$\eta_a(V_h) = \sup_{f\in L^2(\Omega),\|f\|_b=1} \inf_{v_h\in V_h} \|Tf - v_h\|_a \leq Ch,$$

其中算子$T : L^2(\Omega) \to V$ 定义如下

$$a(Tf, v) = b(f, v), \quad \forall f \in L^2(\Omega) \quad \text{and} \quad \forall v \in V.$$

## 思考: 有限元空间与Krylov空间的比较

- 有限元最重要的特点是由于$\eta_a(V_h) \leq Ch$, 所以$L^2$- 范数误差估计比能量范数误差估计高一阶.
- Krylov空间的量$\eta_a(\mathcal{K})$不会变小.
- 根本原因: 有限元空间对一般函数都有逼近性, Krylov只有对特定的空间有逼近性.

## PDE特征值问题的扩展子空间算法: 初值为 $u_h^{(0)}$

1. 当$\ell > 0$, 定义如下的线性边值问题: 求 $\widehat{u}_h^{(\ell+1)} \in V_h$ 满足如下的方程
$$a(\widehat{u}_h^{(\ell+1)}, v_h) = \lambda_h^{(\ell)} b(u_h^{(\ell)}, v_h), \quad \forall v_h \in V_h.$$
2. 定义扩展子空间 $V_{H,h} = V_H + \mathrm{span}\{\widehat{u}_h^{(\ell+1)}\}$, 求解如下特征值问题: 求 $(\lambda_h^{(\ell+1)}, u_h^{(\ell+1)}) \in \mathbb{R} \times V_{H,h}$ 使得 $\|u_h^{(\ell+1)}\|_a = 1$ 且
$$a(u_h^{(\ell+1)}, v_{H,h}) = \lambda_h^{(\ell+1)} b(u_h^{(\ell+1)}, v_{H,h}), \qquad \forall v_{H,h} \in V_{H,h}.$$
求解上面的特征值问题, 选取离空间 $\mathrm{span}\{\widetilde{u}_h^{(\ell+1)}\}$ 最近的特征方向作为 $u_h^{(\ell+1)}$, 结合相应的特征值 $\lambda_h^{(\ell+1)}$ 作为输出.

- 推广到多个特征对的情况: $V_{H,h} = V_H + \mathrm{span}\left\{\widehat{u}_{1,h}^{(\ell+1)}, \cdots, \widehat{u}_{k,h}^{(\ell+1)}\right\}$.
- 求解多个特征对可以将它们分开并行计算, 一种新型的特征值求解的并行算法.

# 扩展子空间的算法和优点

## 误差估计

- 执行扩展子空间算法得到的特征对 $(\lambda_h^{(\ell+1)}, u_h^{(\ell+1)})$ 有如下的收敛性

$$
\begin{aligned}
\|\bar{u}_h - u_h^{(\ell+1)}\|_a &\leq CH^2\|\bar{u}_h - u_h^{(\ell)}\|_a,\\
\|\bar{u}_h - u_h^{(\ell+1)}\|_b &\leq CH\|\bar{u}_h - u_h^{(\ell+1)}\|_a.
\end{aligned}
$$

## 扩展子空间算法的性质

- 针对不同的特征对可以并行进行迭代.
- 避免了高维空间$V_h$中的正交化, 只需要在低维空间$V_H$上进行正交化.
- 适合并行化求解大规模特征值问题.
- 存储量(for $m$ eigenpairs): $mN_{h_n} + M_H$.
- 计算量主要在求解大规模的线性方程组: 采用多重网格求解线性方程时, 每个特征对的计算量为$\mathcal{O}(N_{h_n})$.

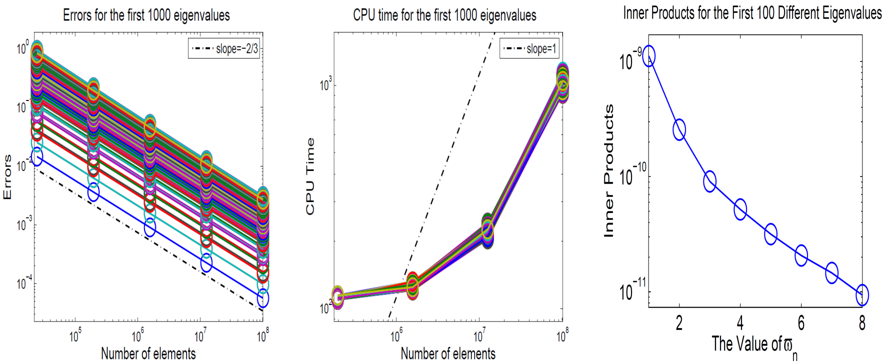


Figure: 左: 前1000个特征值的计算误差, 中: 前1000个特征对的计算时间, 右: 前100 个近似特征函数之间的内积: F. Xu, H. Xie and N. Zhang, A parallel augmented subspace method for eigenvalue problems, SISC, 42(5) (2020), A2655-A2677.

# Compared with LOBPCG from slepc

Table: The CPU time of multigrid and LOBPCG for the first $200$ eigenpairs, where the symbol "–" means the computer runs out of memory.

| Number of Dofs | 274625 | 2146689 | 16974593 |
|---|---|---|---|
| Time of LOBPCG | 367.76 | 5857.029 | – |
| Time of Multigrid | 130.73 | 237.00 | 1107.68 |
| Memory of LOBPCG | 1342 | 6273 | – |
| Memory of Multigrid | 353 | 1554 | 7251 |

Table: The CPU time of multigrid and LOBPCG for the first $1000$ eigenpairs, where the symbol "–" means the computer runs out of memory.

| Number of Dofs | 274625 | 2146689 | 16974593 |
|---|---|---|---|
| Time of LOBPCG | 2826.92 | – | – |
| Time of Multigrid | 132.50 | 242.77 | 1150.86 |
| Memory of LOBPCG | 3802 | – | – |
| Memory of Multigrid | 366 | 1627 | 7452 |

# 一般非线性问题

- 求解如下一般形式的非线性问题:

$$\left\{\begin{array}{rcll} -\nabla\cdot(\mathcal{A}\nabla u)+f(x,u) &=& g, & \text{在}\Omega\text{内}, \\ u &=& 0, & \text{在}\partial\Omega\text{上}, \end{array}\right.$$

- Newton迭代方法是基于对非线性问题的一阶多项式的近似:

$$-\nabla\cdot(\mathcal{A}\nabla u)+\frac{\partial f}{\partial u}(x,u_0)(u-u_0) \;=\; g-f(x,u_0), \quad \text{在}\Omega\text{内}.$$

- 用高阶多项式来构造相应的非线性迭代格式, 比如三阶多项式:

$$\left\{\begin{array}{rcll} -\nabla\cdot(\mathcal{A}\nabla u)+\frac{\partial f}{\partial u}(x,u_0)(u-u_0) &+& \frac{1}{2}\frac{\partial^2 f}{\partial u^2}(x,u_0)(u-u_0)^2 & \\ +\frac{1}{6}\frac{\partial^3 f}{\partial u^3}(x,u_0)(u-u_0)^3 &=& g-f(x,u_0) & \text{在}\Omega\text{内}, \\ u &=& 0, & \text{在}\partial\Omega\text{上}, \end{array}\right.$$

- 利用这里的多重网格算法求解上面的非线性问题只需要渐近最优的计算量, 即与一步Newton迭代格式的计算量相当.
- 由于使用的是高阶多项式的逼近, 这样的非线性迭代的收敛速度将会远快于Newton迭代从而使最终的计算量有大幅度的下降.

# Outline

## 总结

- 介绍了求解非线性问题的扩展子空间算法
- 对非线性问题包括(半线性椭圆问题、特征值问题、半线性特征值问题)可构造多重网格算法和代数多重网格算法
- 计算量$\mathcal{O}(N_h+\varpi M_H)$, ($\varpi$: 特征值迭代次数或非线性迭代次数)
- 适合于并行计算

## 展望

- 已完成对(多尺度)线性椭圆、重调和、特征值非线性、BEC非线性特征值问题, 不等式约束最优控制等问题、Kohn-Sham方程的多重网格算法的设计
- $V_H$与$V_h$不嵌套的多水平校正算法(2021)
- 用高次多项式去做非线性迭代(Newton迭代是一次多项式)

# 介绍材料和软件

## Information

- 介绍网页: www.xyzgate.com/article?id=5b93bca54f022e53886e4b3a
- Matlab程序包(介绍了算法实现的方法, 可以进行算法的测试): lsec.cc.ac.cn/~hhxie
- C语言软件包GitHub (适合并行Matrix-Free 和Vector-Free的): https://github.com/pase2017/pase
- 特征值并行求解软件包GCGE: https://github.com/Materials-Of-Numerical-Algebra/GCGE
- Ongoing: 分布式并行有限元软件包

# Thank You Very Much!