Deep Multigrid: learning restriction and prolongation matrices[1]

Alexandr Katrutsa
joint work with T. Daulbaev and I. Oseledets

Moscow Institute of Physics and Technology
Skolkovo Institute of Science and Technology

November 16, 2017

[1] `https://arxiv.org/abs/1711.03825`

General idea

Problem parametrization

- Parametrization fixes parameters set
- Parametrization controls the quality
- Parametrization gives differentiable steps

Loss function or its upper bound

- Computable in reasonable time
- Differentiable
- Stochastic gradient

Example: geometric multigrid method

- Parameters: restriction and prolongation operators
- Differentiable steps
- Loss function — ?

Problem statement

- Partial differential equation
  Domain is the segment $[0,1]$ and boundary conditions are $u(0) = 0,\ u(1) = 0$.
- Discretization: introducing n points mesh and finite differences approximation
- Linear system:
  $$Au = f$$
- Grid step: $h = \frac{1}{n+1}$

Two-grid idea

1. Perform s_1 steps of iterative process for $u^{(k)}$
2. Compute residual $r^{(k)} = Au^{(k)} - f$
3. Restrict $r^{(k)}$ on coarse grid: $r_c^{(k)} = Rr^{(k)}$
4. Project A on coarse grid: $A_c = RAP$
5. Solve system $A_c u_c^{(k)} = r_c^{(k)}$
6. Update $u^{(k)}$: $\hat{u}^{(k)} = u^{(k)} + Pu_c^{(k)}$
7. Perform s_2 steps of iterative process for $\hat{u}^{(k)}$, get $u^{(k+1)}$

Multigrid

Projection onto coarse grid can perform recursively in step 5

Two-grid as iterative process

Two-grid method is an iterative process

$$u^{(k+1)} = Cu^{(k)} + b$$

with the following iteration matrix

$$C = (M_2^{-1}K_2)^{s_2}(I + P(RAP)^{-1}RA)(M_1^{-1}K_1)^{s_1}$$

and

$$b = ((M_2^{-1}K_2)^{s_2}P(RAP)^{-1}R(s_1AM_1^{-1} - I) + s_2M_2^{-1})f.$$

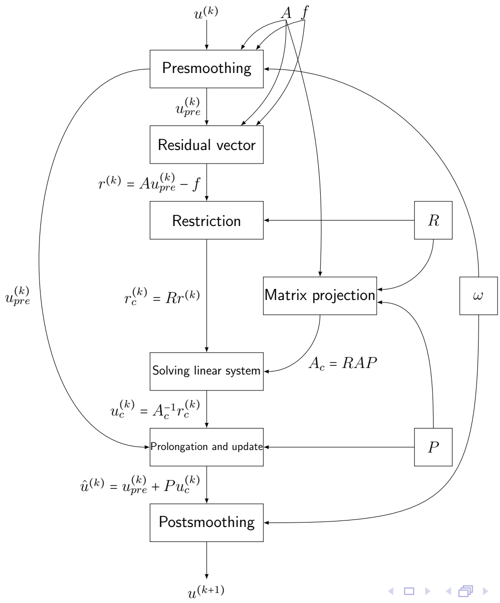
Pre- and postsmoothing — damped Jacobi method

- $M_1 = M_2 = \omega^{-1}D$
- $K_1 = K_2 = \omega^{-1}D - A$

Iterative process analysis

- The matrix C depends on R, P and ω
- Matrix-by-vector product Cx is one iteration of the two-grid method with $u^{(k)} \equiv x$

Neural Network reformulation



Parametrization

Matrices

- Restriction matrix $R \in \mathbb{R}^{m \times n}$, $m = \frac{n-1}{2}$
- Prolongation matrix $P \in \mathbb{R}^{n \times m}$, $m = \frac{n-1}{2}$

Constraints on matrices

- Non-symmetric, non-homogeneous — $3m$ numbers
- Symmetric, non-homogeneous — $2m$ numbers
- Non-symmetric, homogeneous — 3 numbers
- Symmetric, homogeneous — 2 numbers

Scalar

Damp factor $\omega \in \mathbb{R}_{++}$

Optimization problem

Loss function — spectral radius

$$\rho(C) = \max_{i=1,\dots,n} |\lambda_i(C)| \to \min_{R,P,\omega}$$

Hard to optimize! ☹

Gelfand formula

$$\rho(A) = \lim_{k\to\infty} \sqrt[k]{\|A^k\|}$$

Use approximation! ☺

Bounds

For any positive integer K:

$$\gamma^{(1+\ln K)/K} \|A^K\|_F^{1/K} \le \rho(A) \le \|A^K\|_F^{1/K}, \; \gamma \in (0,1)$$

Upper bound minimization

- Stochastic approximation from Hutchinson's estimator:

$$\|A^K\|_F^2 = \mathbb{E}_z\|A^K z\|_2^2,$$

where $z = [z_i]$, such that
  - $z_i \in \mathcal{N}(0, I)$
  - $z_i \in \mathcal{R}\ \left(\mathbb{P}(z_i = \pm 1) = \frac{1}{2}\right)$ — less variance

Optimization problem

$$F_K = \mathbb{E}_z\|C^K z\|_2^2 \to \min_{R,P,\omega}$$

Unbiased estimation

$$\hat{F}_K = \frac{1}{N}\sum_{i=1}^{N}\|C^K z^i\|_2^2$$

How to minimize?

- Stochastic gradient based method (SGD, AdaDelta, **Adam**, ...)
- Autodiff tool: **Autograd**, PyTorch, Theano, etc...
- Custom gradient implementations for some layers
- Baur-Strassen's theorem
- Initialization is crucial!

Initialization

- The problem is strongly non-convex
- Linear interpolation is good for Poisson equation

$$R_{\text{lin}} = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 & & & & \\ & & 1 & 2 & 1 & & \\ & & & & 1 & 2 & 1 \end{bmatrix} \quad P_{\text{lin}} = \frac{1}{2}\begin{bmatrix} 1 & & \\ 2 & & \\ 1 & 1 & \\ & 2 & \\ & 1 & 1 \\ & & 2 \\ & & 1 \end{bmatrix}$$

- But stuck in poor local minima in more complex cases
- How to deal with this issue?

Homotopy

- Homotopy with start matrix A_0 and target matrix A_1
  - Consider sequence of matrix

$$M_i = \alpha_i A_1 + (1 - \alpha_i) A_0,$$
$$\alpha_0 = 0,\ 0 < \alpha_1 < \alpha_2 < \ldots < \alpha_{k-1} < 1,\ \alpha_k = 1$$

  - Solution of the i-th problem is initialization for the $(i+1)$-th problem
  - Grid of α_i is adaptive with acceptance rate τ

Model 1D problems

- Poisson equation: $-\Delta u = f$

$$A = -\frac{1}{h^2}\begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix} \in \mathbb{R}^{n\times n}$$

- Helmholtz equation: $-\Delta u - k^2 u = f$
  - low frequency: $k \approx 10$
  - high frequency: $k \gtrsim 100$
  - piece-wise constant $k(x)$:

$$k(x) = \begin{cases} 1, & 0 \leq x < 0.5 \\ k_{\max}, & 0.5 \leq x \leq 1. \end{cases}$$

- Stationary singularly-perturbed diffusion-convection equation

Poisson equation

Spectral radii ρ for the compared methods

Grid size	Linear	AMG	DMG
7	0.061728	0.182358	0.015088
15	0.061728	0.193726	0.018481
31	0.061728	0.196578	0.027819
63	0.061728	0.197207	0.045068
127	0.061728	0.195878	0.045400

Helmholtz equation: low frequency

Spectral radii ρ for the compared methods

Grid size	k	Linear	AMG	DMG
7	5	0.226356	0.226214	0.012505
13	10	1.808608	0.255912	0.044337
17	15	0.826753	0.406821	0.062037
23	20	3.388036	0.418464	0.067183

Helmholtz equation: high frequency

Spectral radii ρ for the compared methods, grid size $n = 1115$

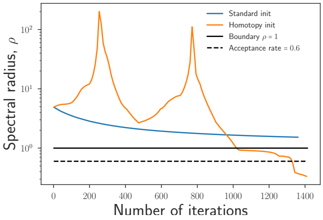
k	Linear	AMG	DMG
100	0.180680	0.198093	0.061088
300	13.389492	0.203956	0.053827
500	14.608550	0.218872	0.066820
700	99.555631	0.243871	0.060205
900	62.940589	0.377024	0.091268
1000	4789.842424	0.607620	0.116077

Helmholtz equation: non-constant $k(x)$

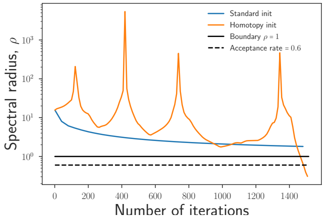
Spectral radii ρ for the compared methods

Grid size	$k_{\max}$	Linear	AMG	DMG
127	100	3.147622	0.330212	0.078162
255	100	1.642432	0.212405	0.047063
511	100	0.194238	0.200955	0.055769

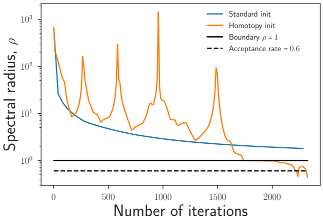
Homotopy performance — $3m$ numbers



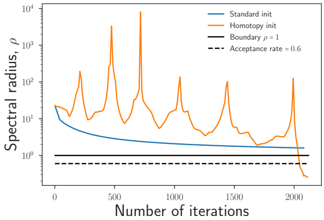
$k = 100, n = 113$



$k = 150, n = 169$

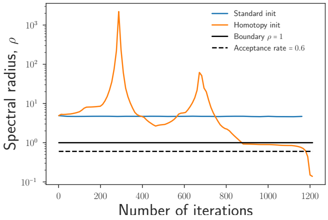


$k = 200, n = 223$

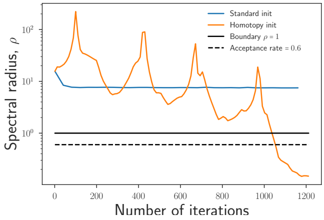


$k = 250, n = 279$

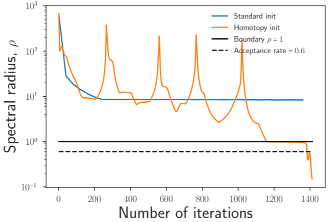
Homotopy performance — 2 numbers



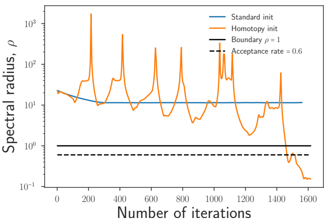
$$k = 100, n = 113$$



$$k = 150, n = 169$$

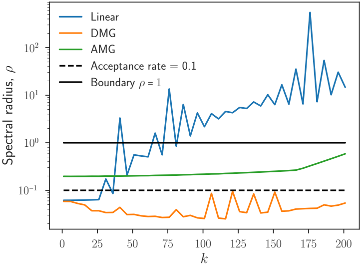


$$k = 200, n = 223$$

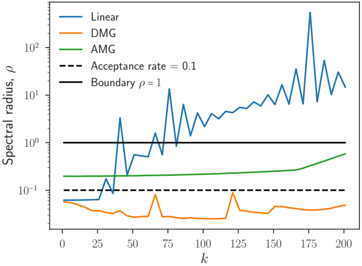


$$k = 250, n = 279$$

Moving frequency from low to high



$3m$ numbers



2 numbers

Stationary diffusion-convection equation

$$-\varepsilon \frac{d^2 u(x)}{dx^2} + \frac{du(x)}{dx} = f(x), \qquad u(0) = 0, \quad u(1) = 0$$

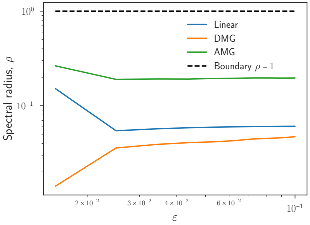
Non-symmetric matrix A:

$$A = -\frac{\varepsilon}{h^2} \begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix} + \frac{1}{h} \begin{bmatrix} -1 & 1 & & & \\ 0 & -1 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & -1 & 1 \\ & & & 0 & -1 \end{bmatrix} \in \mathbb{R}^{n \times n}$$

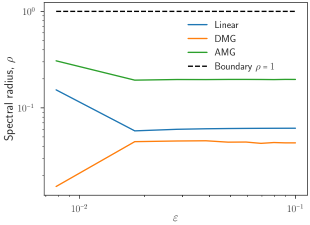
Boundary layer

Grid has to cover boundary layer → $h < \varepsilon$.

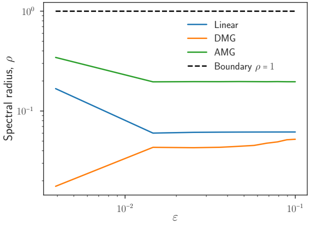
Methods comparison



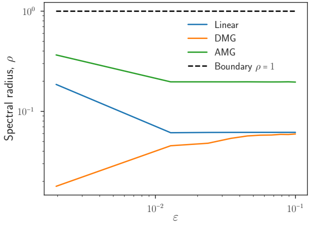
$n = 63$



$n = 127$



$n = 255$



$n = 511$

Summary

- General approach to find locally optimal parameters through NN reformulation
- Unbiased estimation of the loss function for iterative process
- Method to find locally optimal parameters for the two-grid method
- Homotopy initialization
- Robustness under different constraints on the operators

Future work

- Extend to 2D case — almost done
- Optimize sparse preconditioners
- Use GPU-based framework
- Extend approach to other problems