Residual-based Gauss-Seidel method

Jakub Sliacan
supervised by Prof. Peter Oswald at JUB

Max-Planck Institut für Informatik

Problem

Notation

$$\overset{A}{\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix}} \overset{x}{\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}} = \overset{b}{\begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}}$$

In terms of rows

$$\overset{A}{\begin{pmatrix} \text{———} & a_1 & \text{———} \\ \text{———} & a_2 & \text{———} \\ & \vdots & \\ \text{———} & a_n & \text{———} \end{pmatrix}} \overset{x}{\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}} = \overset{b}{\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}}$$

What are we solving?

Given

$A \in \mathbb{R}^{n\times n}$ spd, $b \in \mathbb{R}^n$

What are we solving?

Given

$A \in \mathbb{R}^{n \times n}$ spd, $b \in \mathbb{R}^n$

Find

$x \in \mathbb{R}^n$ which solves $Ax = b$

What are we solving?

Given

$A \in \mathbb{R}^{n \times n}$ spd, $b \in \mathbb{R}^n$

Find

$x \in \mathbb{R}^n$ which solves $Ax = b$

- Why such restriction on A?

What are we solving?

Given

$A \in \mathbb{R}^{n\times n}$ spd, $b \in \mathbb{R}^n$

Find

$x \in \mathbb{R}^n$ which solves $Ax = b$

- Why such restriction on A?
  - spd matrices arise from applications
  - minimization problems
  - structural engineering, circuit simulations, compressed sensing, nuclear reactor diffusion, oil reservoir modelling [3]

What are we solving?

Given

$A \in \mathbb{R}^{n \times n}$ spd, $b \in \mathbb{R}^n$

Find

$x \in \mathbb{R}^n$ which solves $Ax = b$

- Why such restriction on A?
  - spd matrices arise from applications
  - minimization problems
  - structural engineering, circuit simulations, compressed sensing, nuclear reactor diffusion, oil reservoir modelling [3]

$\Longrightarrow$ tailored methods perform better

History of Iterative methods

$1840s$	Jacobi	Jacobi method
$1870s$	Seidel	Gauss-Seidel method
$1910s$	Richardson	Richardson's method
$1930s$	Temple	Method of steepest descend
$1940s$	Young & Frankel	Successive over-relaxation method (SOR)
$1950s$	Hestenes & Stiefel	Conjugate gradient method

Table: Approximate timeline: invention of major iterative methods

Jacobi (cyclic)

Update rule

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left[b_i - \sum_{j=1, j\neq i}^{n} a_{ij} x_j^{(k)} \right]$$

1 sweep through all equations = 1 step

Gauss-Seidel (cyclic)

Update rule

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left[b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right]$$

using most recent values of x saves memory

"Relaxed" Gauss-Seidel (cyclic)

Auxiliary $\tilde{x}^{(k+1)}$

$$a_{ii}\tilde{x}_i^{(k+1)} = \left[b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right]$$

Idea of relaxation applied

$$x_i^{(k+1)} = (1-\omega) x_i^{(k)} + \omega \tilde{x}_i^{(k+1)} \qquad = x_i^{(k)} + \omega(\tilde{x}_i^{(k+1)} - x_i^{(k)})$$

Update rule

$$x_i^{(k+1)} = x_i^{(k)} + \frac{\omega}{a_{ii}} \left[b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - a_{ii} x_i^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right]$$

GS-Southwell (non-cyclic)

Update rule

$$x_i^{(k+1)} = x_i^{(k)} + \frac{\omega}{a_{ii}} \left[b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - a_{ii} x_i^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right]$$

equation to update is NOT the next one, but is picked based on the size of residual

Update rule simplified:

$$a_{ii}x_i^{new} = a_{ii}x_i + \omega\Big(-\sum_{j<i} a_{ij}x_j - a_{ii}x_i - \sum_{j>i} a_{ij}x_j \Big)$$

$$x_i^{new} = x_i + \omega\Big(- a_i x + b_i \Big)$$

In the language of residuals:

$$r = b - Ax \quad \Longrightarrow \quad x_i^{new} = x_i + \frac{\omega}{a_{ii}} r_i$$

Choose equation to update

Update: $$x_{i^*}^{new} = x_{i^*} + \frac{\omega}{a_{i^*i^*}} r_{i^*}$$

Choose equation to update

Update: $x_{i^*}^{new} = x_{i^*} + \frac{\omega}{a_{i^* i^*}} r_{i^*}$

1. Classical GS

$$i^* + +$$

2. GS-Southwell:

$$|r_{i^*}| \geq \beta \cdot ||r||_\infty, \quad 0 < \beta \leq 1$$

Summary of GSS procedure

1. **(Compute the residual)**

$$r^{(k)} = b - Ax^{(k)}$$

2. **(Choose i^*)**

$$|r_{i^*}^{(k)}| \geq \beta \max_i \left\{ |r_i^{(k)}| \right\}$$

3. **(Update)**

$$x_{i^*}^{(k+1)} = x_{i^*}^{(k)} + \frac{\omega}{a_{i^*i^*}} r_{i^*}^{(k)}$$

Sketch of convergence proof

$$e^{(k)} = x - x^{(k)}, \quad a^*_{ii} = \max_i\{a_{ii}\}, \quad \tilde{r}_i = (0\ldots r_i \ldots 0)^T$$

$$e^{(k+1)} - e^{(k)} = -\frac{\omega}{a_{i^*i^*}}\tilde{r}^{(k)}_{i^*} \tag{1}$$

Sketch of convergence proof

$e^{(k)} = x - x^{(k)}, \quad a_{ii}^* = \max_i\{a_{ii}\}, \quad \tilde{r}_i = (0 \ldots r_i \ldots 0)^T$

$$e^{(k+1)} - e^{(k)} = -\frac{\omega}{a_{i^*i^*}} \tilde{r}_{i^*}^{(k)} \qquad (1)$$

$$||e^{(k+1)}||_A^2 = ||e^{(k)}||_A^2 - \frac{\omega(2-\omega)}{a_{i^*i^*}} \left(r_{i^*}^{(k)}\right)^2 \qquad (2)$$

Sketch of convergence proof

$e^{(k)} = x - x^{(k)}, \quad a_{ii}^* = \max_i\{a_{ii}\}, \quad \tilde{r}_i = (0 \dots r_i \dots 0)^T$

$$e^{(k+1)} - e^{(k)} = -\frac{\omega}{a_{i^*i^*}} \tilde{r}_{i^*}^{(k)} \tag{1}$$

$$||e^{(k+1)}||_A^2 = ||e^{(k)}||_A^2 - \frac{\omega(2-\omega)}{a_{i^*i^*}} \left(r_{i^*}^{(k)}\right)^2 \tag{2}$$

$$\left(e^{(k)}, e^{(k)}\right)_A^2 \leq n\beta^{-2} \left(r_{i^*}^{(k)}\right)^2 \left(e^{(k)}, e^{(k)}\right)_A \tag{3}$$

Sketch of convergence proof

$$e^{(k)} = x - x^{(k)}, \quad a^*_{ii} = \max_i\{a_{ii}\}, \quad \tilde{r}_i = (0 \dots r_i \dots 0)^T$$

$$e^{(k+1)} - e^{(k)} = -\frac{\omega}{a_{i^*i^*}} \tilde{r}^{(k)}_{i^*} \tag{1}$$

$$||e^{(k+1)}||^2_A = ||e^{(k)}||^2_A - \frac{\omega(2-\omega)}{a_{i^*i^*}} \left(r^{(k)}_{i^*}\right)^2 \tag{2}$$

$$\left(e^{(k)}, e^{(k)}\right)^2_A \leq n\beta^{-2} \left(r^{(k)}_{i^*}\right)^2 \left(e^{(k)}, e^{(k)}\right)_A \tag{3}$$

$$||e^{(k+1)}||^2_A \leq \left(1 - \frac{\omega(2-\omega)\beta^2}{n a^*_{ii}}\right)^k \cdot ||e^{(0)}||^2_A \tag{4}$$

Problem

1. Introduction

2. GS-Southwell(GSS)

3. Randomized Gauss-Seidel (RGS)

5. Testing

Outlook

Trade-off

classical GS	GS-Southwell
computationally cheap	faster convergence

Trade-off

classical GS	GS-Southwell
computationally cheap	faster convergence

1. combine the advantages of both methods

Trade-off

classical GS	GS-Southwell
computationally cheap	faster convergence

1. combine the advantages of both methods
2. greedy is only locally optimal
   (even if largest residual not chosen every time, we may perform well)

RGS algorithm

This version is unrelated to the GSS method. For now, focus on costs minimization.

1. **(Choose i^*)** $\forall i \in \{1, \ldots, n\}$ we have

$$\mathbb{P}[i^* = i] = p_i$$

2. **(Compute the residual)**

$$r_{i^*}^{(k)} = b_{i^*} - \left(Ax^{(k)}\right)_{i^*}$$

3. **(Update)**

$$x_{i^*}^{(k+1)} = x_{i^*}^{(k)} + \frac{\omega}{a_{i^*i^*}} r_{i^*}^{(k)}$$

RGS algorithm

This version is unrelated to the GSS method. For now, focus on costs minimization.

1. **(Choose i^*)** $\forall i \in \{1, \dots, n\}$ we have

$$\mathbb{P}[i^* = i] = p_i$$

2. **(Compute the residual)**

$$r_{i^*}^{(k)} = b_{i^*} - \left(Ax^{(k)}\right)_{i^*}$$

3. **(Update)**

$$x_{i^*}^{(k+1)} = x_{i^*}^{(k)} + \frac{\omega}{a_{i^*i^*}} r_{i^*}^{(k)}$$

- compute only necessary
- store only necessary

Performance

Need to know two things

1. converges?

Performance

Need to know two things

1. converges?
2. (if yes) how fast?

Performance

Need to know two things

1. converges?
2. (if yes) how fast?

One never knows for “sure”

Performance

Need to know two things

1. converges?
2. (if yes) how fast?

One never knows for “sure”

1. then best we can do is converge **almost** surely

Performance

Need to know two things

1. converges?
2. (if yes) how fast?

One never knows for "sure"

1. then best we can do is converge **almost** surely
2. we can also expect good error reduction at every step

Establishing convergence of RGS I

Theorem (1)

Assume that the next equation to update is chosen uniformly from the set of all n equations. Let $x^{(0)}$ be the initial guess. Then RGS method converges to the solution x with probability 1.

Lemma (2^{nd} Borel-Cantelli Lemma)

Let E_n be a sequence of independent events in a sample space Ω. Then

$$\sum_{n=1}^{\infty} \mathbb{P}(E_n) = \infty \quad \Longrightarrow \quad \mathbb{P}(\bigcap_{n=1}^{\infty} \bigcup_{m=n}^{\infty} E_m) = 1$$

In other words, if $\sum_{n=1}^{\infty} \mathbb{P}(E_n) = \infty$, then with probability 1 infinitely many of E_n happen.

Theorem (1)

Assume that the next equation to update is chosen uniformly from the set of all n equations. Let $x^{(0)}$ be the initial guess. Then RGS method converges to the solution x with probability 1.

Proof 1.

- Let E_k be event that at the k-th step the equation corresponding to the largest residual is chosen
- $\{E_k\}$ independent $\Longrightarrow \sum_{k=1}^{\infty} \mathbb{P}(E_k) = \sum_{k=1}^{\infty} 1/n = \infty$
- Lemma $\Longrightarrow$ with probability 1, infinitely many of E_k happen

$\square$

Theorem (2)

Let $x^{(0)}$ be the initial guess. And let $\mathbb{P}[i^* = i] = 1/n,\ \forall i$. Then the size of the relative error reduction in A-norm is

$$\mathbb{E}\left[||e^{(k+1)}||_A^2\right] \leq \left(1 - \frac{\omega(2-\omega)}{n\kappa(A)}\right) \cdot \mathbb{E}\left[||e^{(k)}||_A^2\right]$$

Theorem (3)

Let $x^{(0)}$ be the initial guess. And let $\mathbb{P}[i^* = i] = a_{ii}/tr(A)$. Then the size of the relative error reduction in A-norm is

$$\mathbb{E}\left[||e^{(k+1)}||_A^2\right] \leq \left(1 - \frac{\omega(2-\omega)\lambda_{min}}{tr(A)}\right) \cdot \mathbb{E}\left[||e^{(k)}||_A^2\right]$$

Proof.

$$||e^{(k+1)}||_A^2 = ||e^{(k)}||_A^2 - \frac{\omega(2-\omega)}{a_{ii}} \left(r_i^{(k)}\right)^2 \tag{5}$$

$$\mathbb{E}\left[||e^{(k+1)}||_A^2\right] = \mathbb{E}\left[||e^{(k)}||_A^2\right] - \mathbb{E}\left[\frac{\omega(2-\omega)}{a_{ii}} \left(r_i^{(k)}\right)^2\right] \tag{6}$$

$$\mathbb{E}\left[||e^{(k+1)}||_A^2\right] = \mathbb{E}\left[||e^{(k)}||_A^2\right] - \omega(2-\omega)\sum_{i=1}^{n}\left(\frac{\left(r_i^{(k)}\right)^2}{a_{ii}} \cdot \mathbb{P}[i]\right) \tag{7}$$

$\square$

Depending on the choice of the probability distribution, we transform equation 7 into Theorem 1 or 2.

Remark: Error reduction depends on $tr(A)$. Often

$$tr(A) \ll n\lambda_{max}$$

Problem

1. Introduction

2. GS-Southwell(GSS)

3. Randomized Gauss-Seidel (RGS)

5. Testing

Outlook

What can be tested?

1. How many indices to pick at random?
2. What are good starting vectors?
3. ...

Let k be the number of indices picked at random from the set $\{1, \ldots, n\}$. Then we can search this sample $\{i_1, \ldots, i_k\}$ to find the index corresponding to the largest residual (within the sample).

Remark

RGSS - Randomized Gauss-Seidel method with hint of Southwell.

Combination of RGS and GSS is dependent on k. In particular, RGS = RGSS(1) and GSS = RGSS(n).

Matrix A

Construct A as it was presented in [2] to demonstrate performance of GSS method.

$$A = toeplitz\left(\left[1\ c_0\left[\frac{1}{1}, \frac{0}{2}, \frac{-1}{3}, \frac{0}{4}, \frac{1}{5}, \frac{0}{6}, \ldots\right]\ \right]\right)$$

Size of k

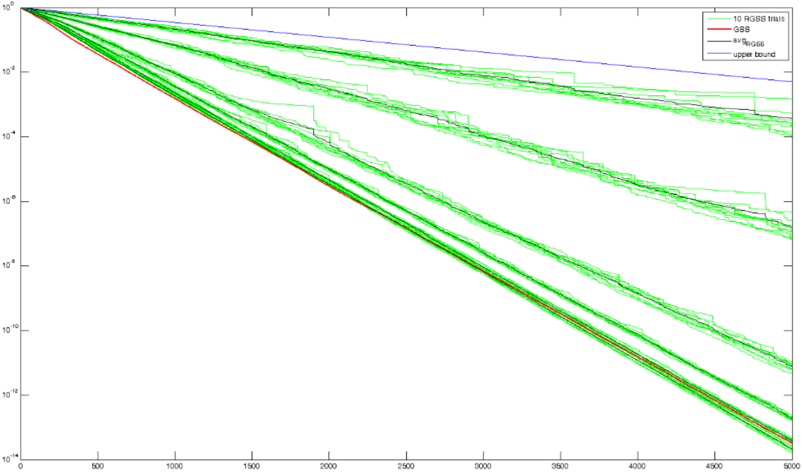

Figure: $n = 500$, matrix A, $k \in \{1, 2, 4, 6, 8, 10\}$

Different starting vectors

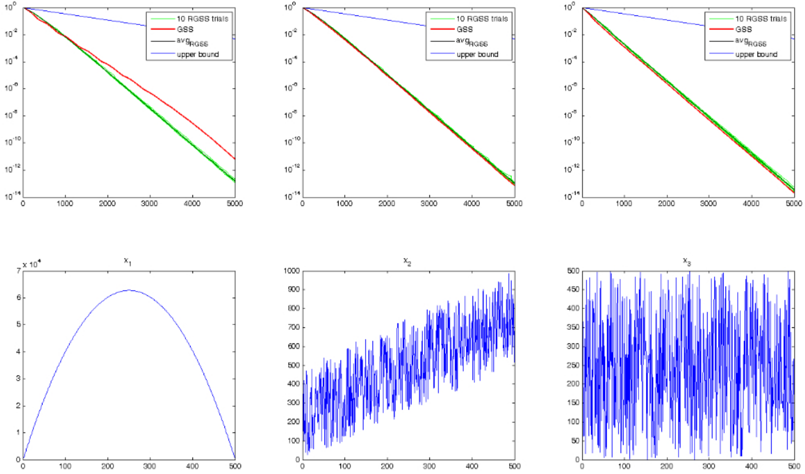

Figure: $n = 100$, $k = 8$, matrix A

Problem

1. Introduction

2. GS-Southwell(GSS)

3. Randomized Gauss-Seidel (RGS)

5. Testing

Outlook

What next?

- Various applications (computed tomography, signal processing, etc.) require solutions to an overdetermined but consistent system of equations $Ax = b$. Kaczmarz method of iterative projections have been found useful. Related to Gauss-Seidel.
- What is optimal size of k?
- What is optimal (or good) choice of the probability distribution for choice of i^*?
- What is RGSS's robustness to different types of spd matrices?

References I

- Jo A. M. Bollen, Numerical stability of descent methods for solving linear equations. *Numer. Math.*, 1984, 43:361-377
- M. Griebel, P. Oswald, Schwarz-Southwell Methods
- Y. Saad, H. A. van der Vorst, Iterative solution of linear systems in the 20th Century, *J. Comput. Applied. Math.*, 2000, 123:1-33
- T. Strohmer, R.Vershynin, A randomized Kaczmarz algorithm with exponential convergence, *J. Fourier Anal. Appl.*, 2009, $15 : 262 - 278$
- S. Frankel, Convergence rates of iterative treatments of partial differential equations, *MTAC*, 1950, 6575
- D.M. Young, Iterative methods for solving partial differential equations of elliptic type, Ph.D. Thesis, Harvard University, Cambridge, MA, USA, 1950
- L.F. Richardson, The approximate arithmetical solution by finite differences of physical problems involving differential equations with an application to the stresses to a masonry dam, *Philos. Trans. Roy. Soc. London Ser*, A, 1910, 210:307-357
- Y. Censor, G. T. Herman, M. Jiang, A note on the behavior of the randomized Kaczmarz algorithm of Strohmer and Vershynin, *Journal of Fourier Analysis and Applications*, 2009, 15:431436
- G.E. Forsythe, Solving linear algebraic equations can be interesting, *Bulletin of AMS*, 1953, 59:299
- D.W. Martin, G. J. Tee, Iterative Methods for Linear Equations with Symmetric Positive Definite Matrix, *The Computer Journal* (1961) 4(3): 242-254

References II

Y.C. Eldar, D. Needell, Acceleration of Randomized Kaczmarz Method via the Johnson-Lindenstrauss Lemma, http://arxiv.org/abs/1008.4397v2

A.S. Householder, Theory of Matrices in Numerical Analysis, Blaisdell Publishing Company, Johnson, CO, 1964, 102

thank you