Sujet de Thèse: Méthodes itératives à retard pour architecture massivement parallèles

Hanyu ZHANG

Laboratoire MICS, École Centrale Supélec, Université Paris Saclay

Soutenue le 29 septembre 2016
devant le jury composé de

M. Frédéric MAGOULÈS	Directeur de thèse
M. Christophe GUYEUX	Rapporteur
Mme. Corinne ANCOURT LE QUELLENEC	Rapporteur
M. Pascal LAURENT	Examinateur
M. Che-Lun HUNG	Examinateur

Summary

Background

Difficulties and Motivations

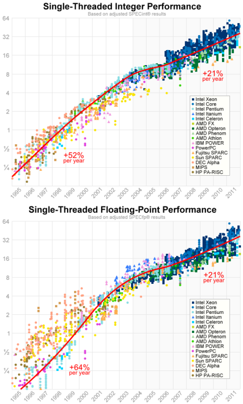


Figure: (up) CPU Integer performance evolution; (down) CPU Float-point performance evolution.

Source: http://preshing.com/20120208/a-look-back-at-single-threaded-cpu-performance/

Table: Large matrix size evolution.

1950	$n = 20$	(Wilkinson)
1965	$n = 200$	(Forsythe and Moler)
1980	$n = 2000$	(LINPACK)
1995	$n = 20000$	(LAPACK)
2010	$n = 1000000$	

Motivation

- Drop of improvements of CPU's performance
- Great involvement in scientific computing
- Explosive growth of typical problem's size
- $O(n^2) \sim O(n^3)$ algorithm complexity
- Fast development of machines of parallel architecture

Model problem

Configuration of numerical tests

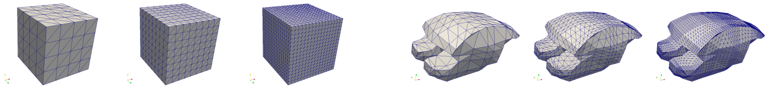


Figure: Test meshes: 3 levels of refinements for numerical tests. (Cube) 125 points, 729 points, 4913 points; (Car) 294 points, 1773 points, 12051 points

Model problem

Heat distribution:

$$\frac{\partial u}{\partial t} - \alpha \nabla^2 u = f$$

with α the thermal diffusivity and f some heat source. Discretize it on time with an implicit Euler scheme as:

$$\frac{u_{n+1} - u_n}{\Delta t} - \alpha \nabla^2 u_{n+1} = f_{n+1}$$

with n the time point. The discretisation in space is done with P_1 finite elements.

Decomposition

First step to parallelisation

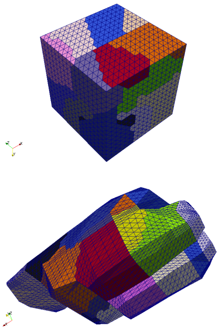


Figure: (up) Domain decomposition (16 blocks) of a cube; (down) Domain decomposition (16 blocks) of a car compartment

Parallelisation is a "divide and conquer" strategy, so the first step is to divide the problem. Usually there is two possibilities,

- Split the matrix directly as we need, Load balance may be considered during the decomposition.
- The underlying domain is already decomposed, then the matrix is splitted according to the underlying domain decomposition.

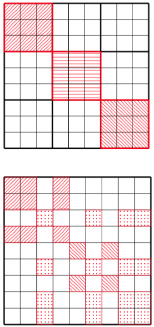


Figure: (up) Illustration of a matrix splitted into 9 common blocks; (down) Illustration of a matrix splitted into 9 blocks according to underlying domain decomposition

Parallelisation

Synchronous and asynchronous communication

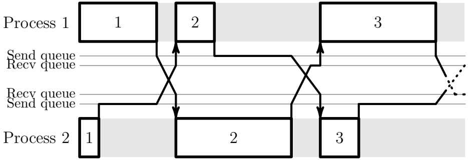


Figure: Synchronous communication

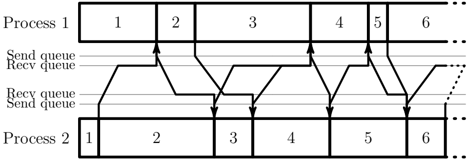


Figure: Asynchronous communication

Synchronous communicaton:

- No need of an extra convergence analysis;
- Simple convergence detection;
- A lot of idle time, communication penalty, not scalable;

Asynchronous communicaton:

- No idle time, no communication penalty, scalable;
- Need a new convergence analysis, usually difficult;
- Complex convergence detection;
- Usually need more iterations to converge, but compensated by assiduity;

Norms

Vector norms and induced matrix norms

Vector norms:

- Weighted maximum norm

$$||x||_\infty^\omega = \max_{1 \leq i \leq n} \frac{||x_i||}{\omega_i}$$

- Block maximum norm

$$||x||_\infty = \max_{1 \leq i \leq p} ||x_i||_i$$

Induced matrix norm:

$$||A|| = \sup_x \frac{||Ax||}{||x||}$$

$$\|x\|_1 = \sum_{i=1}^{m} |x_i|,$$

$$\|x\|_2 = \left(\sum_{i=1}^{m} |x_i|^2 \right)^{1/2} = \sqrt{x^* x},$$

$$\|x\|_\infty = \max_{1 \leq i \leq m} |x_i|,$$

$$\|x\|_p = \left(\sum_{i=1}^{m} |x_i|^p \right)^{1/p} \quad (1 \leq p < \infty).$$

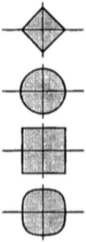


Figure: Geometrical representations of four common norms in two dimensions

Theorem (Norm – Spectral radius inequality)

For any induced matrix norm $|| \cdot ||$, and any $n \times n$ matrix A, we have

$$\lim_{k \to \infty} ||A||^k = \rho(A) \leq ||A||$$

Main references

- Cauchy, A., 1847. Méthode générale pour la résolution des systemes d'équations simultanées. Comp. Rend. Sci. Paris 25, 536–538.
- Akaike, H., 1959. On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method. Ann Inst Stat Math 11, 1–16. doi:10.1007/BF01831719
- Chazan, D., Miranker, W., 1969. Chaotic relaxation. Linear Algebra and its Applications 2, 199–222. doi:10.1016/0024-3795(69)90028-7
- Barzilai, J., Borwein, J.M., 1988. Two-Point Step Size Gradient Methods. IMA J Numer Anal 8, 141–148. doi:10.1093/imanum/8.1.141
- Raydan, M., 1993. On the Barzilai and Borwein choice of steplength for the gradient method. IMA J Numer Anal 13, 321–326. doi:10.1093/imanum/13.3.321
- Raydan, M., others, 1991. Convergence properties of the Barzilai and Borwein gradient method. PhD thesis, Rice University, Houston, Texas.
- Dai, Y.-H., Liao, L.-Z., 20003. R-linear convergence of the Barzilai and Borwein gradient method. IMA J Numer Anal 22, 1-10. doi:10.1093/imanum/22.1.1

Piecewise Constant Gradient Method

Algorithm of piecewise constant gradient method

Main ideas

- Inspired by the classical constant gradient method;
- step-length γ_i determined locally by each piece i;
- more flexible.

Synchronisation needed,

$$x_{,t+1} = \sum_{i=1}^{p} U_i x_{i,t+1} = \sum_{i=1}^{p} U_i (x_{i,t} - \gamma_i g_{i,t})$$

$$= (I - \sum_{i=1}^{p} \gamma_i U_i U_i^T A) x_{,t} + \sum_{i=1}^{p} \gamma_i U_i U_i^T b$$

Algorithm 1: Piecewise constant gradient algorithm

Data: A, b, $x_{,0} \leftarrow x_{init}$
Result: Approximate solution of $Ax = b$
Set $A_{ij} \leftarrow U_i^T A U_j$; $b_i \leftarrow U_i^T b$; $x_{i,0} \leftarrow U_i^T x_{,0}$; γ_i for each processor i;
$t = 0$;
while t $<$ MaxIter do
  $g_{i,t} \leftarrow \sum_{j=1}^{p} A_{ij} x_{j,t} - b_i$;
  $x_{i,t+1} \leftarrow x_{i,t} - \gamma_{\mathbf{i}} g_{i,t}$;
  Gather $x_{i,t+1}$ into $x_{,t+1}$;
  if Global convergence detected then
    return;
  end
  $t \leftarrow t + 1$;
end

Synchronous convergence

Sufficient and necessary theorem

Iterative mapping

$$x_{,t+1} = (I - \sum_{i=1}^{p} \gamma_i U_i U_i^T A) x_{,t} + \sum_{i=1}^{p} \gamma_i U_i U_i^T b$$

The second term is irrelevant to the convergence, as

$$e_{,t+1} = (I - \sum_{i=1}^{p} \gamma_i U_i U_i^T A) e_{,t}$$

Theorem (Sufficient and necessary)

The synchronous piecewise constant gradient method converges if and only if

$$\rho(I - \sum_{i=1}^{p} \gamma_i U_i U_i^T A) < 1$$

Synchronous convergence

Contraction with respect to maximum norm

Theorem

Suppose the matrix A is diagonal dominant, which means for every i, there holds

$$a_{ii} > \sum_{j \neq i} |a_{ij}|, \quad \forall i \in [[1, n]]$$

and in addition the step length γ_i for each piece i satisfies

$$0 < \gamma_i < \min_{k \in S_i} \frac{2}{\sum_{j=1}^{n} |a_{kj}|}, \quad i \in [[1, p]]$$

then the synchronous piecewise constant gradient method converges.

Proof: Sufficient to consider the induced maximum norm,

$$||I - \sum_{i=1}^{p} \gamma_i U_i U_i^T A||_\infty < 1$$

Then we can conclude the proof by the norm-spectral radius inequality and the sufficient and necessary theorem. □

Asynchronous Framework

Description of general asynchronous iterative methods

Define a permuted Cartesian product with p given sets $\mathcal{X}_i \subset \mathbb{R}^{n_i}$, $i \in [[1,p]]$, n_i are their corresponding dimensions, as

$$\mathcal{X} = \{x = \sum_i^p U_i x_i \mid x_i \in \mathcal{X}_i,\ i \in [[1,p]]\}$$

Let there be a set of infinite times $T^{(i)} \subset T = \{0,1,2,\dots\}$ for each processor i, Then the local piece $x_{i,t}^{(i)}$ on processor i keeps the same when $t \notin T^{(i)}$ while it is updated by f_i which is a mapping from $\mathcal{X}$ to $\mathcal{X}_i$ when $t \in T^{(i)}$, i.e.

$$x_{i,t+1}^{(i)} = \begin{cases} f_i(x_{,t}^{(i)}) \quad \textit{with} \quad x_{,t}^{(i)} = \sum_{j=1}^p U_j x^{(j)}_{j,t-\tau_j^i(t)} & \forall t \in T^{(i)}, \\ x_{i,t}^{(i)} & \forall t \notin T^{(i)}, \end{cases}$$

with $\tau_j^i(t)$ a delay function which depends on three variables i,j,t, satisfying

$$\forall i,j \in [[1,p]],\ \forall t \in T,\ \exists k < t, \quad 0 \le \tau_j^i(t) \le \min(t, k + \tau_j^i(t-k) - 1) \ \textit{or const}.$$

Then a whole iterative mapping is defined with f_i and $\tau_j^i(t)$ as

$$f(\tilde{x}_{,t}) = (f_1(\tilde{x}_{,t}), f_2(\tilde{x}_{,t}), \cdots, f_p(\tilde{x}_{,t})),\ \tilde{x}_{,t} = \left(x^{(1)}_{1,t-\tau_1^1(t)}, x^{(2)}_{2,t-\tau_2^2(t)}, \cdots, x^{(p)}_{p,t-\tau_p^p(t)} \right)$$

Assumptions

Essential conditions for asynchronous convergence

Three essential assumptions

- **Embedded sets condition :**

  There is a sequence of nonempty set $\{\mathcal{X}(t)\}_{t\in T}$,

  $$\cdots \subset \mathcal{X}(t+1) \subset \mathcal{X}(t) \subset \cdots \subset \mathcal{X}(0) \subset \mathbb{R}^n$$

- **Synchronous convergence condition :**

  For any sequence $\{x_{,t}\}_{t\in T}$ generated synchronously by f, there holds

  $$f(x_{,t}) \in \mathcal{X}(t+1), \quad \forall t, \; x_{,t} \in \mathcal{X}(t)$$

  and every limit point of $\{x_{,t}\}_{t\in T}$ is a fixed point of synchronous f.

- **Box Condition :**

  For every t, there exists sets $\mathcal{X}_i(t) \subset \mathcal{X}_i$ such that

  $$\mathcal{X}(t) = \{x = \sum_i^p U_i x_i \mid x_i \in \mathcal{X}_i(t), \; i \in [[1, p]]\}$$

Asynchronous convergence framework

Two important theorems

Theorem (Bertsekas & Tsitsiklis 1989)

If the three assumptions are verified, then every limit point of the sequence $\{\tilde{x}_{,t} = \sum_{i=1}^{p} U_i x_{i,t}^{(i)}\}_{t \in T}$ generated by an asynchronous iterative algorithm is a fixed point of f for any initial estimate vector $x_{,0} = \left(x_{1,0}, \dots, x_{p,0}\right) \in \mathcal{X}(0)$.

Theorem (ex. Bahi 2007)

Suppose that the iterative mapping $f : \mathbb{R}^n \to \mathbb{R}^n$ is a contraction with respect to the block maximum norm, i.e.,

$$||f(x) - f(y)||_\infty \leq \alpha ||x - y||_\infty, \ 0 \leq \alpha < 1, \ \forall x, y \in \mathcal{X}$$

then the asynchronous iterative algorithm converges to the unique fixed point admitted by the contraction.

Proof : key point is to consider the following sets,

$$\mathcal{X}(t) = \{x \in \mathbb{R}^n \ : \ ||x - x^*||_\infty \leq \alpha^t ||x_{,0} - x^*||_\infty, \ 0 \leq \alpha < 1\}, \ t \in T$$
$$\mathcal{X}_i(t) = \{x_i \in \mathcal{X}_i \ : \ ||x_i - U_i^T x^*||_i \leq \alpha^t ||x_{,0} - x^*||_\infty\}, \ t \in T, \ \forall i \in [[1, p]]$$

Then conclude the proof by the theorem [Bertsekas & Tsitsiklis 1989]. □

Asynchronous convergence

With respect to the maximum norm

Theorem

Given a diagonal dominant matrix A, then the asynchronous piecewise constant gradient method converges with p step-length γ_i satisfying

$$\gamma_i < \min_{k \in S_i} \frac{2}{\sum_{j=1}^{n} |a_{kj}|}, \quad \forall i \in [[1, p]]$$

Proof : The iterative mapping defined by the synchronous piecewise constant gradient is a contraction with respect to the maximum norm,

$$||f(x) - f(y)||_\infty = ||(I - \sum_{i=1}^{p} \gamma_i U_i U_i^T A)(x - y)||_\infty \leq ||I - \sum_{i=1}^{p} \gamma_i U_i U_i^T A||_\infty ||x - y||_\infty$$

with

$$||I - \sum_{i=1}^{p} \gamma_i U_i U_i^T A||_\infty < 1$$

therefore, by the theorem [ex. Bahi 2007], the asynchronous counterpart of the algorithm converges too. □

Asynchronous convergence

With respect to the block maximum norm based on the Euclidean 2-norm

Theorem

Given a block diagonal dominant and positive definite matrix A, i.e.,

$$\sum_{j\neq i} ||A_{ii}^{-1}||_2 ||A_{ij}||_2 < 1, \quad \forall i \in [[1, p]],$$

then the asynchronous piecewise constant gradient method converges with each step length γ_i satisfying

$$0 < \gamma_i < \frac{2}{\lambda_{i,\max} + \sum_{j\neq i} ||A_{ij}||_2}$$

especially we can take

$$\gamma_i = \frac{2}{\lambda_{i,\max} + \lambda_{i,\min}}$$

where $\lambda_{i,\min}$ and $\lambda_{i,\max}$ are the minimum and the maximum eigenvalue of the block matrix A_{ii} respectively.

Proof : Consider the induced norm of every the band matrix $U_i^T(I - \gamma_i A)$,

$$||U_i^T(I - \gamma_i A)||_2 \leq ||I_{ii} - \gamma_i A_{ii}||_2 + \sum_{j\neq i} \gamma_i ||A_{ij}||_2, \quad \forall i \in [[1, p]]$$

Asynchronous convergence

With respect to the block maximum norm based on the Euclidean 2-norm

Proof continued: In order to obtain a contraction, we need to ensure the right part is strictly less than unit. There is only two possiblities,

$$|1-\gamma_i\lambda_{i,\max}| > |1-\gamma_i\lambda_{i,\min}|$$

which gives

$$\frac{2}{\lambda_{i,\max}+\lambda_{i,\min}} < \gamma_i < \frac{2}{\lambda_{i,\max}+\sum_{j\neq i}||A_{ij}||_2} \text{ and } (7)$$

or

$$|1-\gamma_i\lambda_{i,\max}| \leq |1-\gamma_i\lambda_{i,\min}|$$

which leads to

$$0 < \gamma_i \leq \frac{2}{\lambda_{i,\max}+\lambda_{i,\min}} \text{ and } (7)$$

Notice that

$$||A_{ii}^{-1}||_2 = \frac{1}{\lambda_{i,\min}}$$

Then we concludes the proof by combining all the arguments above. □

Numerical test results

Scalability

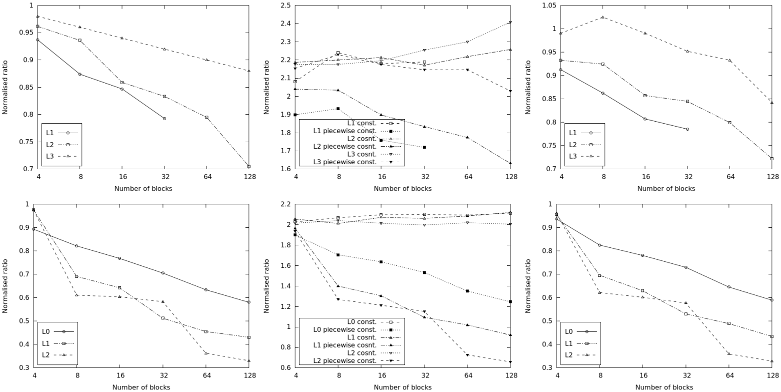


Figure: (left): ratio of synchronous piecewise constant gradient method to the synchronous constant gradient method; (center): ratio of asynchronous methods to the synchronous constant gradient method; (right): ratio of asynchronous piecewise constant gradient method to the asynchronous constant gradient method; (up): cube domain; (down) car compartment domain.

Comparison of results in both synchronous and asynchronous cases

A typical example

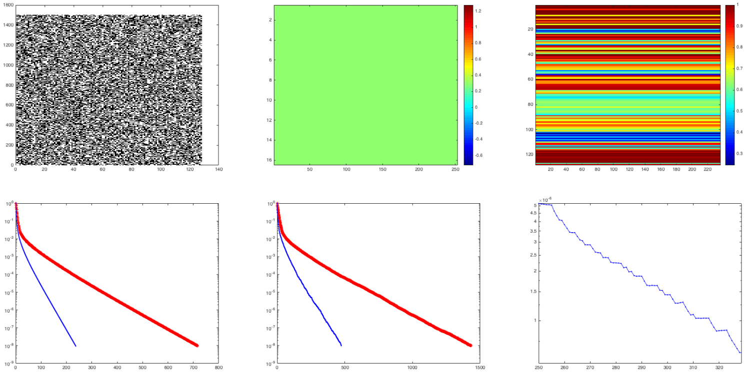


Figure: (up left): communication pattern; (up center): step-length for constant gradient method. (up right): step-length for piecewise constant gradient method; (down left): synchronous residuals, red for constant gradient method and blue for piecewise gradient descent method; (down center): asynchronous residuals; (down right): a zoom of the asynchronous piecewise gradient descent method.

Piecewise gradient descent method

Algorithm of the piecewise gradient descent method

Main ideas

- Inspired by the steepest descent method.
- A local step-length $\gamma_{i,t}$ is calculated on piece i own as

$$\gamma_{i,t} = \frac{g_{i,t}^T g_{i,t}}{g_{i,t}^T A_{ii} g_{i,t}}$$

- less computational work required

Conceptually we still calculate the step length for each piece as the inverse of the Rayleigh quotient with respect to the matrix A along piece of the negative gradient direction at some iteration point.

Algorithm 2: Piecewise gradient descent algorithm

Data: A, b, $x_{,0} \leftarrow x_{init}$
Result: Approximate solution of $Ax = b$
Set $A_{ij} \leftarrow U_i^T A U_j$; $b_i \leftarrow U_i^T b$; $x_{,0}^{(i)} \leftarrow x_{,0}$; c_i for each processor i;
$t = 0$;
while t $<$ MaxIter do
  $g_{i,t} \leftarrow \sum_{j=1}^{p} A_{ij} x_{j,t}^{(i)} - b_i$;
  $\gamma_{i,t} \leftarrow g_{i,t}^T g_{i,t} / g_{i,t}^T A_{ii} g_{i,t}$;
  $x_{i,t+1}^{(i)} \leftarrow x_{i,t}^{(i)} - c_i \gamma_{i,t} g_{i,t}$;
  Gather $x_{j,t+1}^{(j)}$ from other processors into $x_{,t+1}^{(i)}$;
  if Global convergence detected then
    return;
  end
  $t \leftarrow t + 1$;
end

Geometrical insights

An geometrical interpretation of the synchronous piecewise gradient descent method

A two dimensional illustration of the idea:

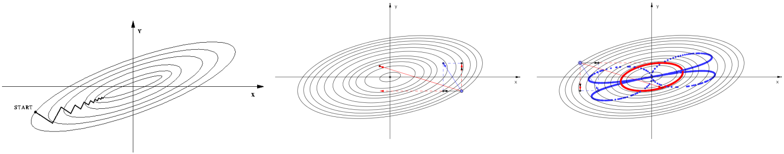


Figure: (left) An illustration of the steepest descent method; (center) Comparison of the steepest descent method and piecewise gradient descent method for one particular iteration; (right) Traces of one iteration effect of the piecewise gradient descent method (red) and the steepest descent method (blue)

However in dimensions higher than 2, a simple recombination of descent vector not necessarily reduces the objective function. Hence we need to introduce some scale coefficients to make the recombined descent vector not too large. The most simple idea was to introduce a uniform scale coefficient for all pieces, which leads us to the following theorem.

Synchronous convergence

Theorem and Proof

Theorem

Given a symmetric positive definite (SPD) matrix A and a coefficient $c_i = 2/p$ for every $i \in [[1, p]]$, then the piecewise gradient descent method converges at least as linearly (measured with respect to the A-norm) as

$$||x_{,t} - x^*||_A \leq \alpha^t ||x_{,0} - x^*||_A, \quad \textit{with} \quad \alpha^2 = \frac{\kappa^2 - (2 - \frac{2}{p})}{\kappa^2 + (2 - \frac{2}{p})}$$

where $\kappa = ||A|| ||A^{-1}||$ is the condition number of matrix A.

Proof: With the convention of notation of error, $e_{,t} = x_{,t} - x^*$, we have

$$e_{,t+1} = e_{,t} - \frac{2}{p} \sum_{i=1}^{p} \gamma_{i,t} U_i g_{i,t}$$

Consider the square of A-norm $||e_{,t}||_A^2$.

Synchronous convergence

Proof continued

Proof continued : We get $||e_{,t+1}||_A^2 = \alpha^2 ||e_{,t}||_A^2$ with

$$\alpha^2 = 1 - \frac{4}{p^2} \sum_{j=i+1}^{p} \sum_{i=1}^{p} (\gamma_{i,t} g_{i,t}^T g_{i,t} + \gamma_{j,t} g_{j,t}^T g_{j,t} - 2\gamma_{i,t}\gamma_{j,t} g_{i,t}^T A_{ij} g_{j,t}) / ||e_{,t}||_A^2$$

Denote $U_i g_{i,t} = \sum_{k=1}^{n} \beta_{ik} v_k, \ \forall i \in [[1,p]]$ where v_i are eigenvectors of A. Then we get

$$\gamma_{i,t} g_{i,t}^T g_{i,t} = \frac{(\sum_{k=1}^{n} \beta_{ik}^2)^2}{\sum_{k=1}^{n} \lambda_k \beta_{ik}^2}$$

$$\gamma_{i,t}\gamma_{j,t} g_{i,t}^T A_{ij} g_{j,t} = \frac{(\sum_{k=1}^{n} \beta_{ik}^2)(\sum_{k=1}^{n} \beta_{jk}^2)(\sum_{k=1}^{n} \lambda_k \beta_{ik} \beta_{jk})}{(\sum_{k=1}^{n} \lambda_k \beta_{ik}^2)(\sum_{k=1}^{n} \lambda_k \beta_{jk}^2)}$$

and again by $e_{,t} = A^{-1} g_{,t}$, Insert them into α^2, we have

$$\alpha^2 = 1 - \frac{2}{p^2} \left(\sum_{j=1}^{p} \sum_{i=1}^{p} \frac{\sum_{k=1}^{n} \lambda_k (\beta_{ik} (\sum_{k=1}^{n} \beta_{jk}^2) - \beta_{jk} (\sum_{k=1}^{n} \beta_{ik}^2))^2}{(\sum_{k=1}^{n} \lambda_k \beta_{ik}^2)(\sum_{k=1}^{n} \lambda_k \beta_{jk}^2)} \right) / \sum_{k=1}^{n} \frac{(\sum_{i=1}^{p} \beta_{ik})^2}{\lambda_k}$$

Synchronous convergence

Proof continued

Proof continued : Taking $\lambda_k = \lambda_{\max}$ and $\lambda_k = \lambda_{\min}$ for every $k \in [[1, n]]$ properly,

$$\alpha^2 \leq 1 - \frac{2}{p^2\kappa^2}\left(\sum_{j=1}^{p}\sum_{i=1}^{p}\frac{\sum_{k=1}^{n}(\beta_{ik}(\sum_{k=1}^{n}\beta_{jk}^2) - \beta_{jk}(\sum_{k=1}^{n}\beta_{ik}^2))^2}{(\sum_{k=1}^{n}\beta_{ik}^2)(\sum_{k=1}^{n}\beta_{jk}^2)}\right)/\sum_{k=1}^{n}\left(\sum_{i=1}^{p}\beta_{ik}\right)^2$$

Notice that $g_{i,t}$ are perpendicular among them, which means

$$\sum_{k}^{n}\beta_{ik}\beta_{jk} = 0, \quad \forall i \neq j$$

Expand it,

$$\alpha^2 \leq 1 - \frac{4(p-1)}{p^2\kappa^2}\left(\sum_{i=1}^{p}\sum_{k=1}^{n}\beta_{ik}^2\right)/\left(\sum_{k=1}^{n}\sum_{i=1}^{p}\beta_{ik}^2\right) = 1 - \frac{4(p-1)}{p^2\kappa^2} \leq \frac{\kappa^2 - (2-\frac{2}{p})}{\kappa^2 + (2-\frac{2}{p})}$$

which concludes the proof. □

But the A-norm doesn't satisfies the Box condition. So how do we pass to the asynchronous convergence ?

Geometrical illustration

An geometrical interpretation of asynchronous convergence in two dimensions

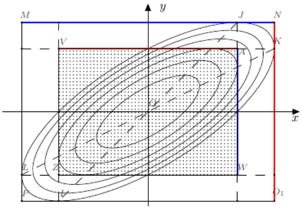


Figure: Two-dimensional Illustration

The red right boundary of outer rectangle is mapped into the up red line of the inner rectangle. The blues boundries have a similar relation.

- Suppose $x^T Ax = C$ with the SPD matrix A as,

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} a & b \\ b & c \end{pmatrix}$$

- Tangent points:

$$(-\frac{b}{a}\sqrt{\frac{aC}{\det A}}, \sqrt{\frac{aC}{\det A}}), \quad (\frac{b}{a}\sqrt{\frac{aC}{\det A}}, -\sqrt{\frac{aC}{\det A}})$$

$$(-\sqrt{\frac{cC}{\det A}}, \frac{b}{c}\sqrt{\frac{cC}{\det A}}), \quad (\sqrt{\frac{cC}{\det A}}, -\frac{b}{c}\sqrt{\frac{cC}{\det A}})$$

- Vector ω:

$$w = \begin{pmatrix} \sqrt{\frac{a_{22}}{\det A}} & \sqrt{\frac{a_{11}}{\det A}} \end{pmatrix}^T$$

- Convergence condition:

$$\det A = ac - b^2 > 0$$

Geometrical insights

General components gradient descent method

Theorem (Multiple dimensional components asynchronous case)

Given a symmetric positive definite (SPD) matrix A, if the matrix additionally satisfies the following condition,

$$\sum_{j \neq i} |a_{ij}| \sqrt{M_{jj}} < a_{ii} \sqrt{M_{ii}}, \quad \forall i \in [[1, n]]$$

where M_{st} is the (s, t)-th minor of the matrix A, then the asynchronous component gradient descent method with scale coefficients $c_i = 1, \forall i \in [[1, n]]$ converges from any start point.

Inspired by the geometrical analysis we give the following definition,

Weighted diagonal dominance

A matrix A is called weighted diagonal dominant with respect to a positive vector ω if it holds the following condition,

$$a_{ii} > \sum_{j \neq i} |a_{ij}| \frac{\omega_j}{\omega_i} \quad \forall i \in [[1, n]]$$

Asynchronous convergence

With respect to the weighted maximum norm

Theorem

Given a positive definite matrix A which is also weighted diagonal dominant with respect to a positive vector ω, with coefficients c_i satisfying

$$c_i < \min_{k \in S_i} \frac{2\omega_k \lambda_{i,\min}}{\sum_{j=1}^{n} |a_{kj}|\omega_j}, \quad \forall i \in [[1,p]]$$

then the asynchronous piecewise gradient descent method converges.

proof : Consider the weighted maximum norm $||\cdot||_\infty^\omega$,

$$||I - \sum_{i=1}^{p} c_i \gamma_{i,t} U_i U_i^T A||_\infty^\omega < 1 \iff \max_{k \in S_i, i \in [[1,p]]} \left(|1 - c_i \gamma_{i,t} a_{kk}| + c_i \gamma_{i,t} \sum_{j \neq k} |a_{kj}| \frac{\omega_j}{\omega_k} < 1 \right)$$

which gives the condition of weighted diagonal dominance and the following condition,

$$c_i \gamma_{i,t} < \min_{k \in S_i} \frac{2\omega_k}{\sum_{j=1}^{n} |a_{kj}|\omega_j}, \quad i \in [[1,p]]$$

With $\lambda_{i,\min} \leq \dfrac{1}{\gamma_{i,t}} \leq \lambda_{i,\max}$, we can completes the proof. □

Asynchronous convergence

With respect to the block maximum norm based on the Euclidean 2-norm

Theorem

Given a block diagonal dominant and positive definite matrix A, then the asynchronous piecewise gradient descent method converges with each step length γ_i satifying

$$0 < c_i < \frac{2\lambda_{i,\min}}{\lambda_{i,\max} + \sum_{j \neq i} ||A_{ij}||_2}$$

especially we can take

$$c_i = \frac{2\lambda_{i,\min}}{\lambda_{i,\max} + \lambda_{i,\min}}$$

where $\lambda_{i,\min}$ and $\lambda_{i,\max}$ are the minimum and the maximum eigenvalue of the block matrix A_{ii} respectively.

Proof : We should ensure that

$$c_i \gamma_{i,t} < \frac{2}{\lambda_{i,\max} + \sum_{j \neq i} ||A_{ij}||_2}$$

Again by the range of $\gamma_{i,t}$, $\lambda_{i,\min} \leq \dfrac{1}{\gamma_{i,t}} \leq \lambda_{i,\max}$, we prove the theorem. □

Numerical test results

Scalability

Table: Evolution of #iteration to #blocks for car mesh

#pieces	16	32	64	128
#iter. sync. SD	67	67	67	67
#iter. sync. p. SD	45	36	35	34
#iter. async. SD	131	138	141	132
#iter. async. p. SD	93	81	65	69

Table: Evolution of #iteration to #blocks for cube mesh

#pieces	16	32	64	128
#iter. sync. SD	39	39	39	39
#iter. sync. p. SD	42	42	40	38
#iter. async. SD	88	91	87	85
#iter. async. p.SD	75	83	82	70

Comparison of results in both synchronous and asynchronous cases

A typical example

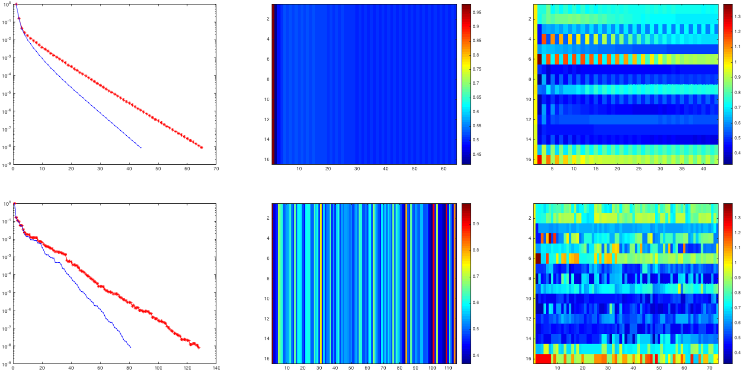


Figure: (up) Synchronous comparison of the steepest descent method (red) and the piecewise gradient descent method (blue); (up center): Step-length evolution for the whole matrix; (up right): Local step-length's evolutions for every pieces; (down) similar to the figure above but in asynchronous case.

Piecewise constant gradient method with retard(s)

Algorithm of piecewise constant gradient method with retard(s)

Main ideas

- Inspired by the Barzilai-Borwein (BB) method.
- Step lengths are still calculated locally as inverse of Rayleigh quotient.
- A uniform or multiple retards scheme for pieces.

The BB method converges much faster than the Steepest descent method. There are two advantages of parallelisation with retards, economise time by doing calculation during sending receiving informations, an "inheritance" of fast convergence is also observed.

Algorithm 3: Piecewise gradient descent algorithm with a uniform retard or multiple retards

Data: A, b, r, $x_{,0} \leftarrow x_{init}$
Result: Approximate solution of $Ax = b$
Set $A_i \leftarrow U_i^T A$; $b_i \leftarrow U_i^T b$; $x_{i,0} \leftarrow U_i^T x_{,0}$; c_i for each processor i;
$t = 0$;
while t $<$ MaxIter do
  $g_{i,t} \leftarrow A_i x_{,t} - b_i$;
  $x_{i,t+1} \leftarrow x_{i,t} - c_i \gamma_{i,t-r} g_{i,t}$ (or $\gamma_{i,t-r_i}$);
  $\gamma_{i,t} \leftarrow g_{i,t}^T g_{i,t} / g_{i,t}^T A_{ii} g_{i,t}$;
  Gather $x_{i,t+1}$ into $x_{,t+1}$;
  if Global convergence detected then
    return;
  end
  $t \leftarrow t + 1$;
end

Synchronous and asynchronous convergence analysis

Theorem

Given a positive definite matrix A which is also weighted diagonal dominant with respect to a positive vector ω, with coefficients c_i satisfying

$$c_i < \min_{k \in S_i} \frac{2\omega_k \lambda_{i,\min}}{\sum_{j=1}^{n} |a_{kj}|\omega_j}, \quad \forall i \in [[1, p]]$$

then the asynchronous piecewise gradient descent method with retards converges.

Theorem

Given a block diagonal dominant and positive definite matrix A, If each step length γ_i satisfies

$$0 < c_i < \frac{2\lambda_{i,\min}}{\lambda_{i,\max} + \sum_{j \neq i} ||A_{ij}||_2}$$

especially we can take

$$c_i = \frac{2\lambda_{i,\min}}{\lambda_{i,\max} + \lambda_{i,\min}}$$

where $\lambda_{i,\min}$ and $\lambda_{i,\max}$ are the block matrix A_{ii}'s the minimum and the maximum eigenvalue, then the asynchronous piecewise gradient descent method with retards converges

Numerical test results

Scalability

Table: Evolution of #iteration to #blocks for cube mesh

#blocks	16	32	64	128
#iter. sync. unif.	22	22	22	20
#iter. sync. p. unif.	35	36	36	34
#iter. async. unif.	81	93	83	86
#iter. async. p. unif.	70	76	71	69
#iter. async. multi.	88	91	94	84
#iter. async. p. multi.	87	85	87	81

Table: Evolution of #iteration to #blocks for car mesh

#blocks	16	32	64	128
#iter. sync. unif.	33	33	33	33
#iter. sync. p. unif.	36	32	32	32
#iter. async. unif.	102	120	103	120
#iter. async. p. unif.	66	69	67	61
#iter. async. multi.	101	134	142	120
#iter. async. p. multi.	91	80	77	71

Comparison of results in both synchronous and asynchronous cases

A typical example

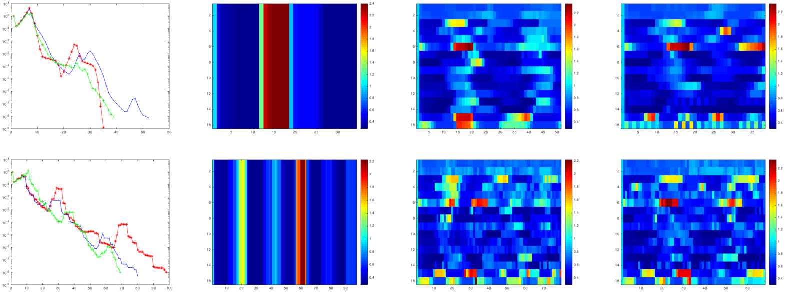


Figure: (1): Residual comparison among steepest descent method with retard (red), asynchronous piecewise gradient descent method with retard (blue) and asynchronous Piecewise Gradient Descent Method with multi-retards (green) synchronous (up) or asynchronous (down); (2) Uniform step-length with uniform retard; (c) Local step-length with uniform retard; (d) Local step-length with multiple retards

Shanks transformation

ϵ-algorithm for accelerating the convergence of a scalar sequence

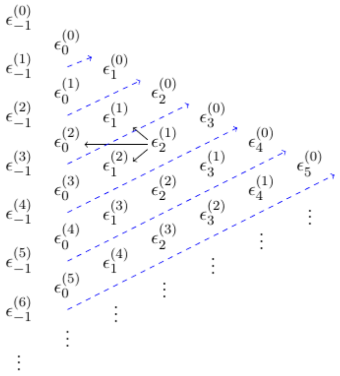


Figure: Illustration of ϵ-algorithm

- ϵ-algorithm**??**:

$$\epsilon_{-1}^{(t)} = 0,\ \epsilon_0^{(t)} = x_t,\ t = 0, 1, 2, \cdots$$

$$\epsilon_{k+1}^{(t)} = \epsilon_{k-1}^{(t+1)} + \left(\epsilon_k^{(t+1)} - \epsilon_k^{(t)}\right)^{-1}, k, t = 0, 1, \cdots$$

- Shanks transformation:

$$s_{t,k} = \frac{\begin{vmatrix} x_t & x_{t+1} & \cdots & x_{t+k} \\ \Delta x_t & \Delta x_{t+1} & \cdots & \Delta x_{t+k} \\ \vdots & \vdots & & \vdots \\ \Delta x_{t+k-1} & \Delta x_{t+k} & \cdots & \Delta x_{t+2k-1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ \Delta x_t & \Delta x_{t+1} & \cdots & \Delta x_{t+k} \\ \vdots & \vdots & & \vdots \\ \Delta x_{t+k-1} & \Delta x_{t+k} & \cdots & \Delta x_{t+2k-1} \end{vmatrix}} = \epsilon_{2k}^{(t)}$$

with $\Delta x_j = x_{j+1} - x_j$

Topological ϵ-algorithm (TEA)

A generalisation of ϵ-algorithm to vector sequences

Define the inverse of an ordered pair (a, b) of vectors such that $a^T b \neq 0$ to be the ordered pair (b^{-1}, a^{-1}), where

$$b^{-1} = \frac{1}{b^T a} a, \quad a^{-1} = \frac{1}{b^T a} b$$

For even subscripts $2k$, the inverse of $\Delta\epsilon_{2k}^{(n)} = \epsilon_{2k}^{(n+1)} - \epsilon_{2k}^{(n)}$ is with respect to a fixed vector y, which is arbitrary except for the restriction that all the required inverse should exist for each k and n. For odd subscripts, the inversion is respect to the y^{-1}, which leads to the topological epsilon algorithm (TEA)

$$\begin{aligned}
&\epsilon_{-1}^{(t)} = 0, \ \epsilon_0^{(t)} = x_t, \ t = 0, 1, 2, \cdots \\
&\epsilon_{2k+1}^{(t)} = \epsilon_{2k-1}^{(t+1)} + y / \left(y^T \Delta\epsilon_{2k}^{(t)} \right), k, t = 0, 1, 2, \cdots \\
&\epsilon_{2k+2}^{(t)} = \epsilon_{2k}^{(t+1)} + \Delta\epsilon_{2k}^{(t)} / \left((\Delta\epsilon_{2k+1}^{(t)})^T \epsilon_{2k}^{(t)} \right), k, t = 0, 1, 2, \cdots
\end{aligned}$$

Brezinski demonstrated that

$$\epsilon_{2k}^{(t)} = s_{t,k}$$

Vector sequences can be accelerated

Acceleration theorem

Consider a sequence of vector $\{x_t\}_{t\in T}$ satisfying

$$x_t \sim x^* + \sum_{i=1}^{\infty} v_i \lambda_i^t \quad (as \;\; t \to \infty)$$

where $\lambda_i \neq 1$, $\lambda_i \neq \lambda_j, i \neq j$, and $|\lambda_1| \geq |\lambda_2| \geq \cdots$.

Theorem

Assume that there exists a vector u that

$$u^T v_i \neq 0, \quad 1 \leq i \leq k,$$

and v_i are linearly independent, λ_i satisfy

$$|\lambda_1| \geq \cdots \geq |\lambda_k| \geq |\lambda_{k+1}| \geq |\lambda_{k+2}| \geq \cdots$$

Then for all sufficiently large n, $s_{t,k}$ as given by topological Shanks transformation exists, furthermore, we have

$$s_{t,k} - x^* = \Lambda(n)\lambda_{k+1}^t\,(1 + o(1)) \;\; as \; t \to \infty$$

where $\Lambda(t)$ are all bounded for sufficiently large t.

Consistent iterations

Conditions for acceleration

Let us be restricted to the following chaotic relaxation,

$$x_{i,t+1}^{(i)} = \begin{cases} B_i x_{,t}^{(i)} + C_i d \quad with \quad x_{,t}^{(i)} = \sum_{j=1}^{p} U_j x_{j,t-\tau_j^i(t)}^{(j)} & \forall t \in T^{(i)}, \\ x_{i,t}^{(i)} & \forall t \notin T^{(i)}, \end{cases}$$

Definition (Consistent iteration)

Consistent iteration with respect to $Ax = d$ if its solution x^* is a fixed point of the mapping above in synchronous case, i.e. $x^* = Bx^* + Cd$

- *Jacobi method :* $B = I - D^{-1}A$
- *Gaussi-Seidel method :* $B = -(D+L)^{-1}U$
- *Piecewise constant gradient method :* $B = I - \sum_{i=1}^{p} \gamma_i U_i U_i^T A$

Problem: However they usually converge s-l-o-w-l-y ... But once it is convergent,

$$\frac{||s_{t,k} - x^*||}{||x_{t+k+1} - x^*||} = O\left(\left(\frac{\lambda_{k+1}}{\lambda_1}\right)^t\right)$$

Periodic relaxation with retard(s)

Methods between Jacobi and Gauss-Seidel model

Periodic update scheme:

$$T^{(i)} = \{t \in \mathbb{N} : i \equiv (t \bmod n)\}$$

Retard(s):

- *Jacobi :* $\tau_j^i(t) = i$
- *Gauss-Seidel :* $\tau_j^i(t) = 1$

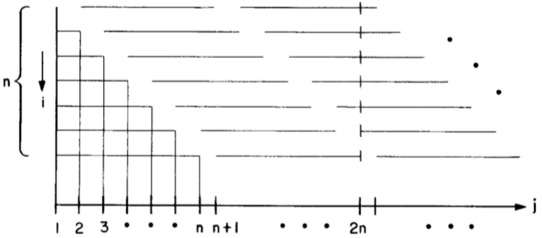


Figure: An illustration figure of periodic relaxation from [Chazan & Miranker 1969]

Retard effects

Larger retards

An artificial configuration:

- communication time : computation time $= 3 : 1$
- (up) retard $= 2$; (down) retard $= 3$

Idle time ratio:

- (up) 0.625 (down) 0.375

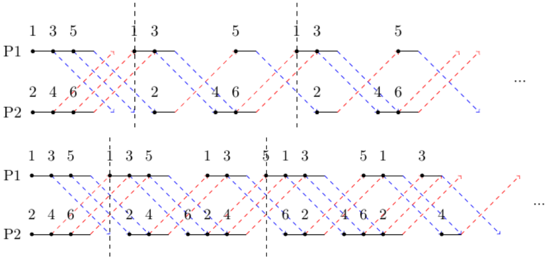


Figure: Retard affects idle time ratio

Relaxation method with uniform retard

Periodic scheme

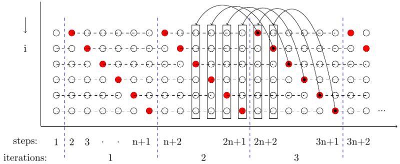


Figure: Every component is updated with a uniform retard

By the regular pattern of updating we add an additional equation

$$x_i^{qn+1+i} = x_i^{qn+2+i} = \cdots = x_i^{qn+n+i}$$

$$\begin{aligned}
x_1^{(q+1)n+1+1} &= \sum_{\gamma=1}^{n-r+1} b_{1,\gamma} x_\gamma^{qn+1+\gamma} + \sum_{\gamma=n-r+2}^{n} b_{1,\gamma} x_\gamma^{qn+\gamma} \\
x_2^{(q+1)n+1+2} &= \sum_{\gamma=1}^{n-r+2} b_{2,\gamma} x_\gamma^{qn+1+\gamma} + \sum_{\gamma=n-r+3}^{n} b_{2,\gamma} x_\gamma^{qn+\gamma} \\
&\vdots \\
x_r^{(q+1)n+1+r} &= \sum_{\gamma=1}^{n} b_{r,\gamma} x_\gamma^{qn+1+\gamma} \\
x_{r+1}^{(q+1)n+1+(r+1)} &= \sum_{\gamma=1}^{1} b_{r+1,\gamma} x_\gamma^{(q+1)n+1+\gamma} + \sum_{\gamma=2}^{n} b_{r+1,\gamma} x_\gamma^{qn+1+\gamma} \\
x_{r+2}^{(q+1)n+1+(r+2)} &= \sum_{\gamma=1}^{2} b_{r+2,\gamma} x_\gamma^{(q+1)n+1+\gamma} + \sum_{\gamma=3}^{n} b_{r+2,\gamma} x_\gamma^{qn+1+\gamma} \\
&\vdots \\
x_n^{(q+1)n+1+n} &= \sum_{\gamma=1}^{n-r} b_{n,\gamma} x_\gamma^{(q+1)n+1+\gamma} + \sum_{\gamma=n-r+1}^{n} b_{n,\gamma} x_\gamma^{qn+1+\gamma}
\end{aligned}$$

Assemblage of most recently updated components

Now let z^q be the n-dimensional vector whose components are

$$z_i^q = x_i^{(q-1)n+1+i}, i = 1, \cdots, n$$

Multiplying the ith equation above by a_{ii}, $i = 1, \cdots, n$, we obtain a recurrence relation for z^q:

$$Dz^{q+2} = B_0 z^{q+2} + B_1 z^{q+1} + B_2 z^q$$

Where B_0, B_1, B_2 are $n \times n$ matrices, satisfying $B_0 + B_1 + B_2 = D - A$ and whose elements $(B_p)_{i,j}$ are:

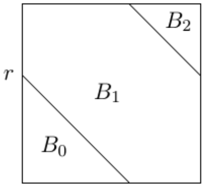


Figure: Illustration of matrix splitting

$$[B_0]_{ij} = \begin{cases} -a_{ij}, & i \geq r+1, \ j \leq i-r \\ 0, & otherwise \end{cases}$$

$$[B_2]_{ij} = \begin{cases} -a_{ij}, & i \leq r-1, \ j \geq n-r+1+i \\ 0, & otherwise \end{cases}$$

$$[B_1]_{ij} = \begin{cases} 0, & i \geq r+1, \ j \leq i-r \\ 0, & i \leq r-1, \ j \geq n-r+1+i \\ 0, & i = j \\ -a_{ij}, & otherwise \end{cases}$$

A specific acceleration theorem

for sparse matrix

Theorem

$$\textit{Note}, \quad m = \min\{r, n-r+1\} \quad \textit{and} \quad [D_1]_{ij} = \begin{cases} [B_1]_{ij}, & |i-j| < m \\ 0, & \textit{otherwise} \end{cases}$$

Suppose $B_2 = 0$, $D + D_1$ and A are symmetric positive definite (SPD). Then the sequence of assembled vector z^q generated by periodic scheme can be accelerated by TEA algorithm.

Proof :

- Since $B_2 = 0$, then the iteration scheme reduces to

$$Dz^{q+1} = B_0 z^{q+1} + B_1 z^q$$

- Let λ, z be the largest absolute value of eigenvalue of $(D - B_0)^{-1} B_1$ and the corresponding eigenvector, then

$$|\lambda| = \frac{|z^T D_1 z + z^T (B_1 - D_1) z|}{|z^T D z - z^T (B_1 - D_1) z|}$$

- The SPD property of $D + D_1$ and A leads us to $|\lambda| < 1$. $\square$

A general convergence theorem

Theorem

If the consistent iteration mapping is a contraction with respect to the weighted maximum norm diagonal with a positive vector ω, i.e.,

$$||B||_{\infty}^{\omega} < 1$$

then the iterates z^q generated periodically with fixed retard(s) converges and can be accelerated by TEA algorithm. Particularly, this happens if the spectral radius of $|B|$ is less than unit, i.e.,

$$\rho(|B|) < 1$$

Proof idea :

- There is a vector Z^q composed by a set $\{z^{q_i}\}_{i \in I}$ and a corresponding matrix $\mathcal{B}$ such that the periodic updates with fixed retard defines a linear transformation, i.e.,

$$Z^{q+1} = \mathcal{B} Z^q$$

- The linear transformation is a contraction by the main theorem of chaotic relaxation.

General acceleration for small uniform retard

An example proof of acceleration for uniform retard less than dimension

Proof : Note the three terms recurrence

$$(D - B_0)z^{q+2} = B_1 z^{q+1} + B_2 z^q$$

can actually be written as two terms of recurrence in higher dimensions as follows,

$$\begin{pmatrix} z^{q+2} \\ z^{q+1} \end{pmatrix} = \begin{pmatrix} (D - B_0)^{-1} B_1 & (D - B_0)^{-1} B_2 \\ I & 0 \end{pmatrix} \begin{pmatrix} z^{q+1} \\ z^q \end{pmatrix}$$

Now take

$$Z^{q+1} = \begin{pmatrix} z^{q+2} \\ z^{q+1} \end{pmatrix}$$

we have

$$Z^q = Z^* + \sum_{i=0}^{2n} \alpha_i V_i \lambda_i^q$$

where $Z^* = (x^*, x^*)$ is the limit, then choose a proper vector y. □

General acceleration for large uniform retard

Another example proof of acceleration for uniform retard larger than dimension

Proof : For $r \geq n$, suppose that

$$r = in + \tau, \quad 1 \leq \tau \leq n$$

Then we have another recurrence relation as,

$$Dz^{(q+i+1)} = B_0 z^{(q+1)} + B_1 z^{(q)} + B_2 z^{(q-1)}$$

Similarly, it can be written as two terms of recurrence in higher dimensions,

$$\begin{pmatrix} z^{q+i+1} \\ z^{q+i} \\ \vdots \\ z^{q+1} \\ z^{q} \end{pmatrix} = \begin{pmatrix} 0 & \dots & 0 & D^{-1}B_0 & D^{-1}B_1 & D^{-1}B_2 \\ I & 0 & \dots & 0 & 0 & 0 \\ 0 & I & 0 & \dots & 0 & 0 \\ 0 & 0 & I & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \dots & 0 & I & 0 \end{pmatrix} \begin{pmatrix} z^{q+i} \\ \vdots \\ z^{q+1} \\ z^{q} \\ z^{q-1} \end{pmatrix}$$

The rest is similar. $\square$

General acceleration for multiple retards scheme

An illustration

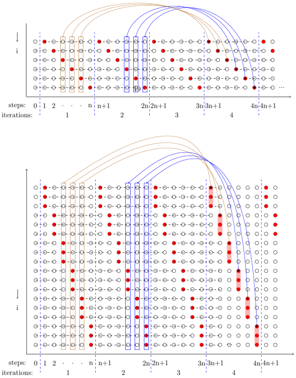


Figure: Multi retard

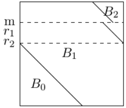


Figure 1: $m < r_1 < r_2$

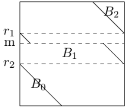


Figure 2: $r_1 < m < r_2$

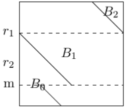


Figure 3: $r_1 < r_2 < m$

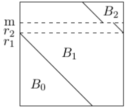


Figure 4: $m < r_2 < r_1$

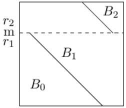


Figure 5: $r_2 < m < r_1$

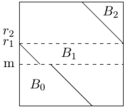


Figure 6: $r_2 < r_1 < m$

Fixed order of update and fixed pattern of retards, it is a linear transform. Again try to find

$$Z^{q+1} = \mathcal{B} Z^{q}$$

Numerical test resultss

12 times of TEA acceleration

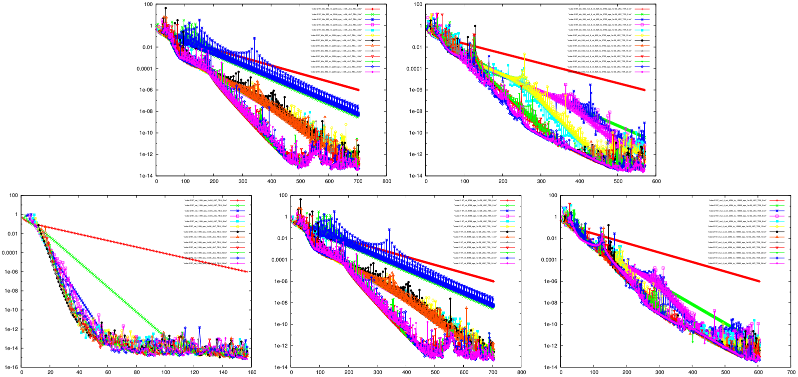


Figure: (up left): block uniform large retard; (up right): block multiple large retards; (down left): small uniform retard; (down center): large uniform retard; (down right): large multiple retards

Conclusions

My contribution mainly consists of

- Proposition of a possible parallelisation scheme for constant gradient method, steepest descent method and gradient method with retards;
- Convergence established in both synchronous and asynchronous cases under certain conditions;
- Demonstration by numerical tests of a superiority to their underlying methods with a direct parallelisation scheme and a better scalability;
- Proposition of a general methodology not only can be used to accelerate relaxation methods, but also supposed to be used on parallel, though it is by nature synchronous, by some auto adaptive scheme or by proper choice of retards, it is able to simulate the effect of a asynchronous method to some extent;
- Demonstration of the effectiveness of the methodology by numerical tests .

Perspectives

- More numerical tests of larger size;
- Apply the idea to other gradient like methods;
- A closer look into the conditions;
- Inheritance investigation;
- Convergence rate estimation;
- Stability of the parallel algorithms;
- Parallelisation of Krylov methods.

Stabilisation method

Stability parameter τ

- Galerkin/least-squares (GLS) method

$$a(v^h, u^h) + (\mathcal{L}v^h, \tau\mathcal{L}u^h)_{\tilde{\Omega}} = (v^h, f) + (\mathcal{L}v^h, \tau f)_{\tilde{\Omega}}$$

- Galerkin-gradient/least-squares GGLS method

$$a(v^h, u^h) + (\nabla\mathcal{L}v^h, \tau^G\nabla\mathcal{L}u^h)_{\tilde{\Omega}} = (v^h, f) + (\nabla\mathcal{L}v^h, \tau^G\nabla f)_{\tilde{\Omega}}$$

with $\mathcal{L} = -\Delta - k^2$

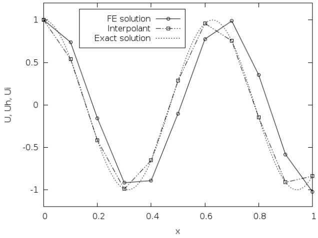


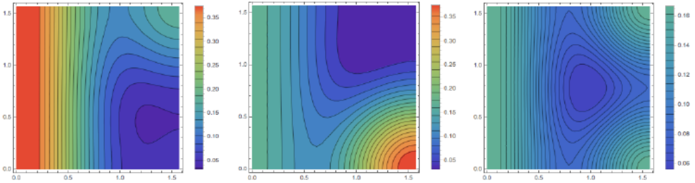
Figure: $k = 10$ and $h = 0.1$ with dirichlet condition on $x = 0$

Distribution of errors of numerical wave number

According to propagation direction and element's shape

Theorem

The difference between the exact wavenumber k and the numerical wavenumber k^h depends on the wave's propagation direction. In fact the coefficient α_3 belongs to the range $[-\frac{\beta_{\max}^2}{24}, -\frac{1}{24(\frac{1}{\beta_1^2}+\frac{1}{\beta_2^2}+\frac{1}{\beta_3^2})}]$ with $\beta_{\max} = \max\{\beta_1, \beta_2, \beta_3\}$. Moreover α_3 hits the upper bound while $\tan^2(\phi) = \frac{\beta_1^2}{\beta_2^2}$ and $\tan^2(\theta) = \frac{\beta_3^2(\beta_1^2+\beta_2^2)}{\beta_1^2\beta_2^2}$ and α_3 takes the lower bound while the wave's propagation is aligned with the longest edge.



(a) $h_x : h_y : h_z = 1:2:3$ (b) $h_x : h_y : h_z = 3:1:2$ (c) $h_x : h_y : h_z = 1:1:1$

Figure: error percent

Stability parameter

Dispersion analysis

Dispersion analysis gives,

$$\tau k^2 = 1 - \frac{\gamma_0 + \gamma_x f_x + \gamma_y f_y + \gamma_z f_z + \gamma_{xy} f_x f_y + \gamma_{yz} f_y f_z + \gamma_{xz} f_x f_z + \gamma_{xyz} f_x f_y f_z}{k^2(2+f_x)(2+f_y)(2+f_z)}$$

with the coefficient γ's are defined as,

$$\begin{aligned}
\gamma_0 &= 24(h_x^{-2} + h_y^{-2} + h_z^{-2}) \quad \gamma_x = 12(h_y^{-2} + h_z^{-2} - 2h_x^{-2}) \\
\gamma_y &= 12(h_x^{-2} + h_z^{-2} - 2h_y^{-2}) \quad \gamma_z = 12(h_x^{-2} + h_y^{-2} - 2h_z^{-2}) \\
\gamma_{yz} &= 6(h_x^{-2} - 2h_y^{-2} - 2h_z^{-2}) \quad \gamma_{xz} = 6(h_y^{-2} - 2h_x^{-2} - 2h_z^{-2}) \\
\gamma_{xy} &= 6(h_z^{-2} - 2h_x^{-2} - 2h_y^{-2}) \quad \gamma_{xyz} = -6(h_x^{-2} + h_y^{-2} + h_z^{-2})
\end{aligned}$$

$$\begin{aligned}
f_x &= \cos(k_1 h_x),\ k_1 = k\cos(\phi)\sin(\theta) \\
f_y &= \cos(k_2 h_y),\ k_2 = k\sin(\phi)\sin(\theta) \\
f_z &= \cos(k_3 h_z),\ k_3 = k\cos(\theta)
\end{aligned}$$

with θ the polar angle and ϕ the azimuthal angle of the wave's propagation direction.

Stabilised Finite Element method for the Helmholtz Equation

In the cases that the mesh's elements are cubes.

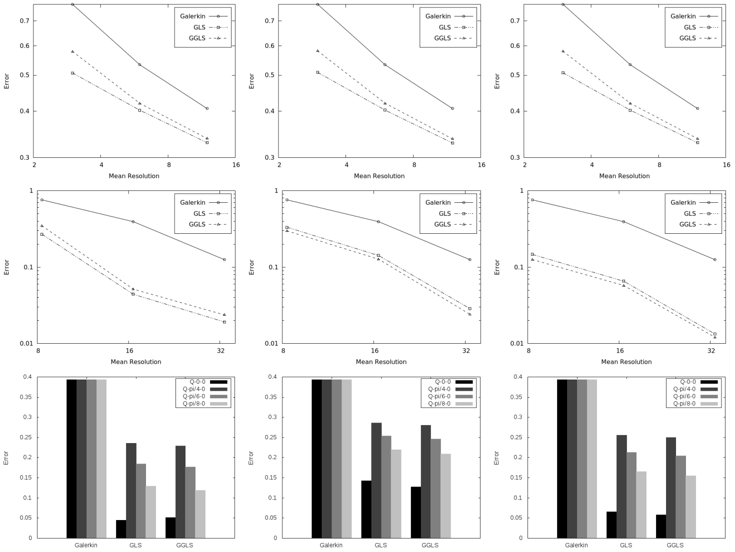
GLS stability parameter

$$\tau_{GLS} k^2 = 1 - \frac{18}{k^2 h^2} \frac{4 - f_x f_y - f_x f_z - f_y f_z - f_x f_y f_z}{(f_x + 2)(f_y + 2)(f_z + 2)}$$

GGLS stability parameter

$$\tau_{GGLS} k^4 = \frac{h^2 k^2}{18} \frac{(f_x + 2)(f_y + 2)(f_z + 2)}{(4 - f_x f_y - f_x f_z - f_y f_z - f_x f_y f_z)} - 1$$

Numerical test results



Merci beaucoup!
Question?