Multigrid Methods for Markov Chains

Hans De Sterck

Department of Applied Mathematics, University of Waterloo

NII Tokyo

collaborators

- Killian Miller

  Department of Applied Mathematics, University of Waterloo, Canada

- Steve McCormick, Tom Manteuffel, John Ruge

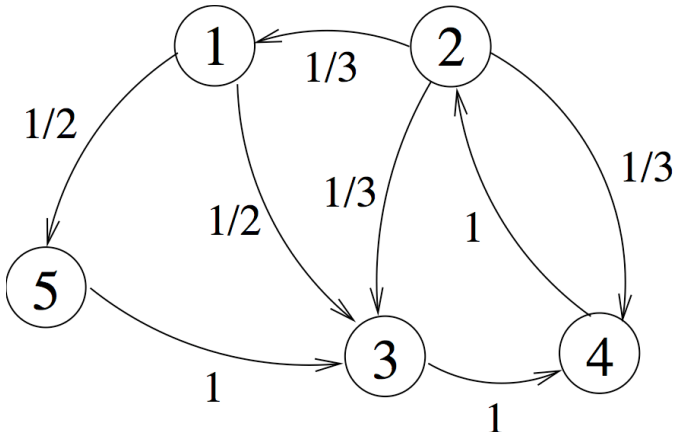
  Department of Applied Mathematics, University of Colorado at Boulder, USA

- Geoff Sanders

  Lawrence Livermore National Laboratory, USA

NII Tokyo
hdesterck@uwaterloo.ca

1. simple Markov chain example



$$B = \begin{bmatrix} 0 & 1/3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1/2 & 1/3 & 0 & 0 & 1 \\ 0 & 1/3 & 1 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- start in one state with probability 1: what is the stationary probability vector after ∞ number of steps?

$$\mathbf{x}_{i+1} = B\,\mathbf{x}_i$$

- stationary probability:

$$B\,\mathbf{x} = \mathbf{x} \qquad \|\mathbf{x}\|_1 = 1$$

$$\mathbf{x}^T = [2/19\ 6/19\ 4/19\ 6/19\ 1/19]$$

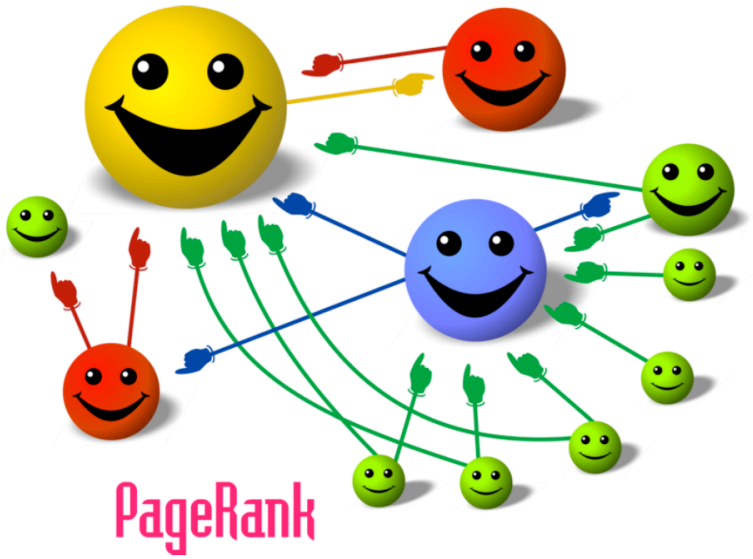
- this particular Markov chain is an example of a random walk on a graph

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

applications of Markov Chains

- information retrieval
- performance modelling of computer systems
- analysis of biological systems
- queueing theory
- Google's PageRank
- ...



University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

2. problem statement

$$B\,\mathbf{x} = \mathbf{x} \qquad \|\mathbf{x}\|_1 = 1 \qquad x_i \geq 0\,\forall i$$

- *B* is column-stochastic

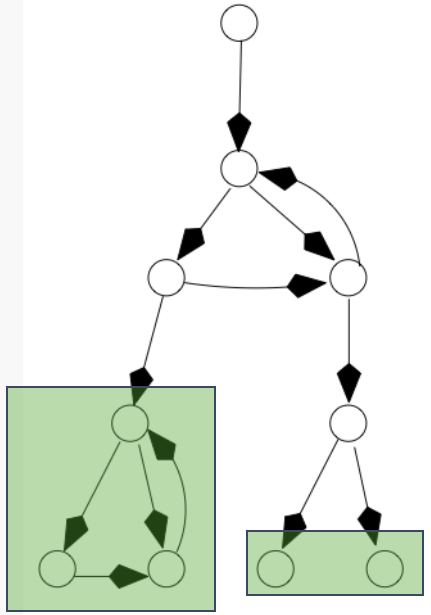
$$0 \leq b_{ij} \leq 1\ \forall i,\ j \qquad \mathbf{1}^T B = \mathbf{1}^T$$

- *B* is irreducible (every state can be reached from every other state in the directed graph)

$$\Rightarrow$$

$$\exists!\ \mathbf{x}: \qquad B\,\mathbf{x} = \mathbf{x} \qquad \|\mathbf{x}\|_1 = 1 \qquad x_i > 0\ \forall i$$

(no probability sinks!)



probability sinks
not irreducible

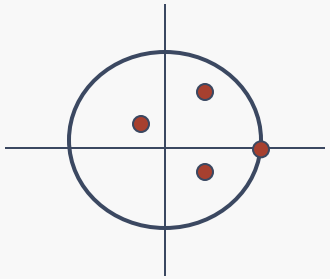
University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

3. power method

$$B\,\mathbf{x} = \mathbf{x} \quad \text{or} \quad (I - B)\,\mathbf{x} = 0 \quad \text{or} \quad A\,\mathbf{x} = 0$$

- largest eigenvalue of *B*: $\lambda_1 = 1$



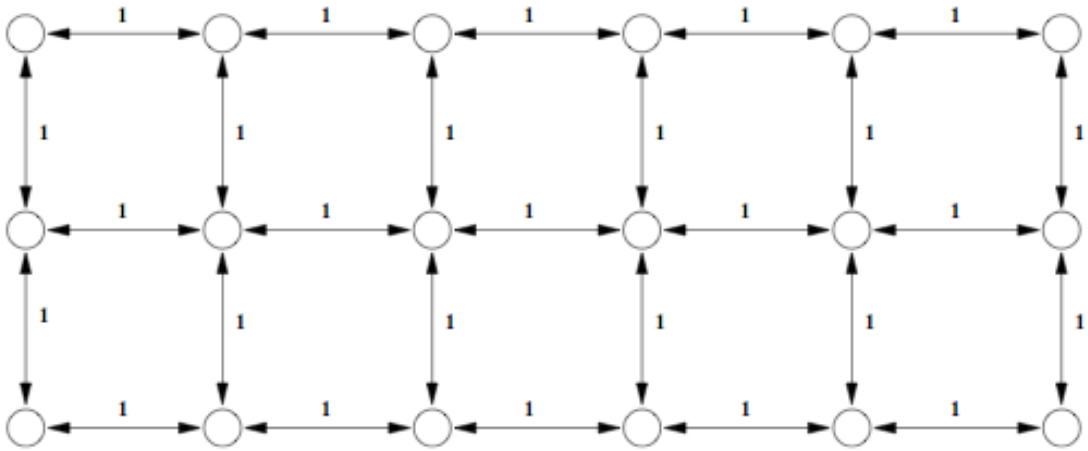
- power method: $\mathbf{x}_{i+1} = B\mathbf{x}_i$

  – convergence factor: $|\lambda_2|$
  – convergence is very slow when $|\lambda_2| \approx 1$
  (slowly mixing Markov chain) (JAC, GS also slow)

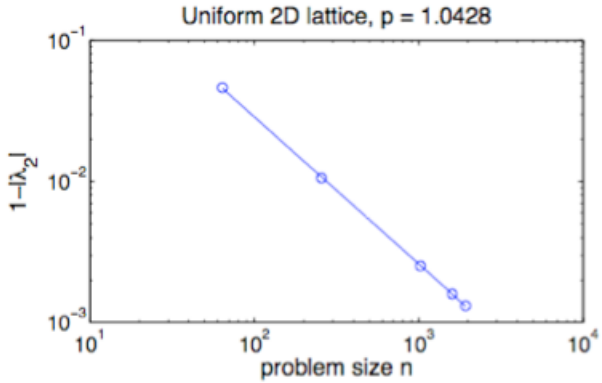
NII Tokyo
hdesterck@uwaterloo.ca

some example Markov chains

- uniform 2D lattice



symmetric,
real spectrum for B



University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

some example Markov chains

- tandem queuing network

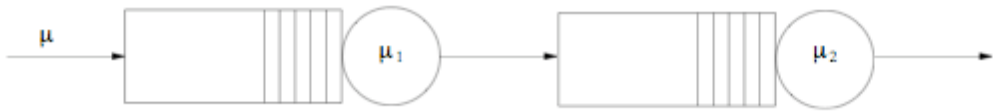


FIG. 5.6. *Tandem queueing network.*

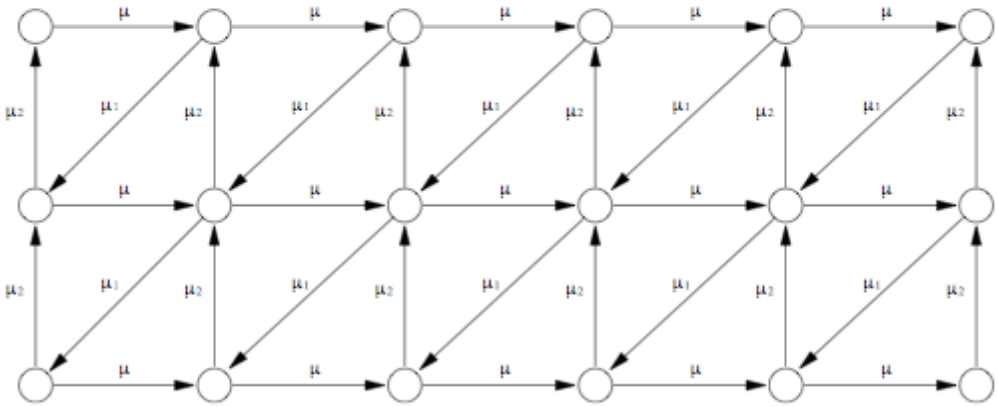
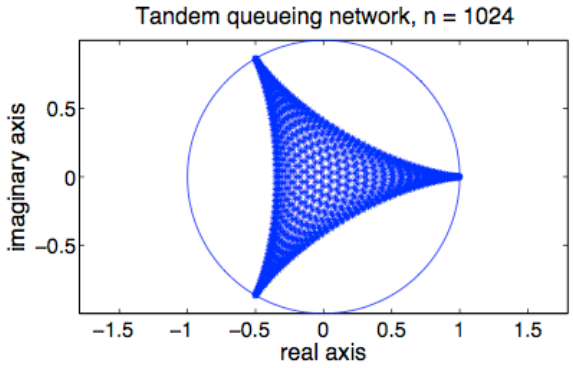
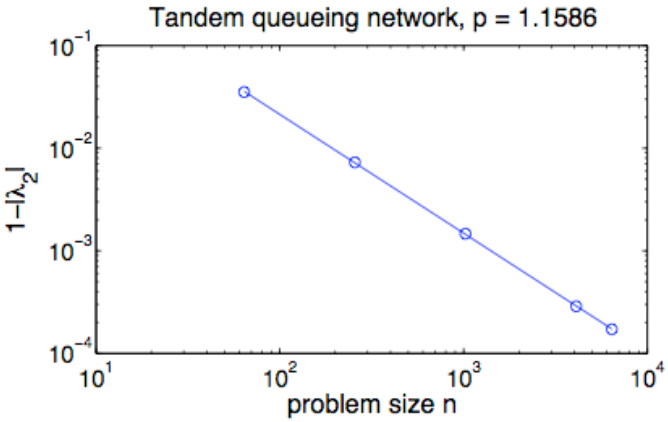


FIG. 5.7. *Graph for tandem queueing network.*



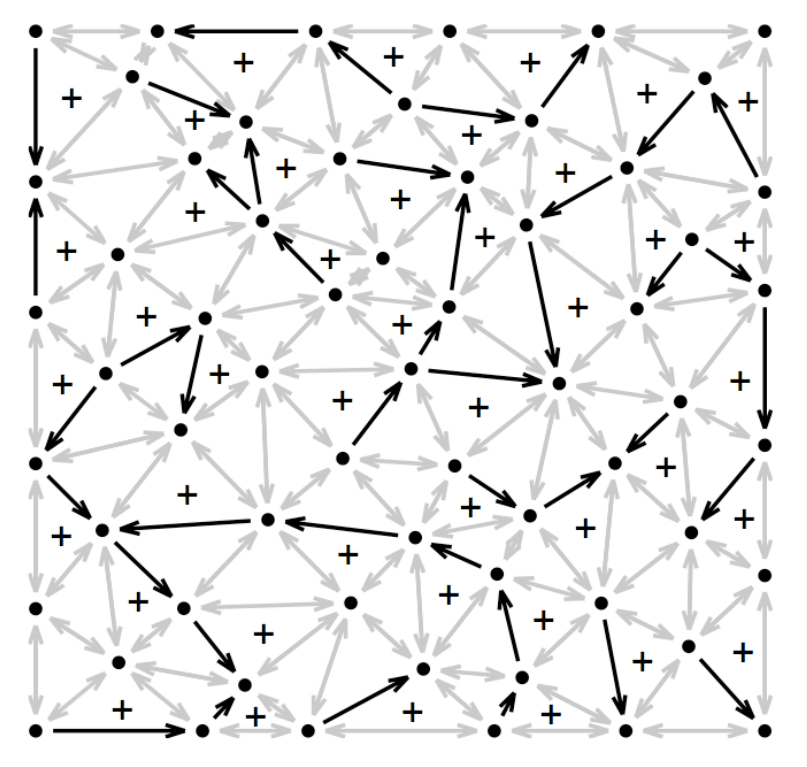


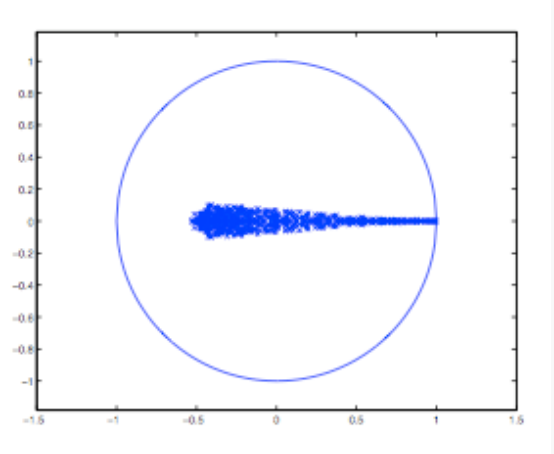
University of Waterloo

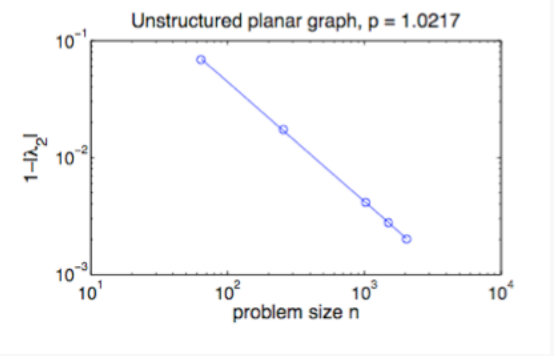
NII Tokyo
hdesterck@uwaterloo.ca

some example Markov chains

- directed, unstructured planar graph







University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

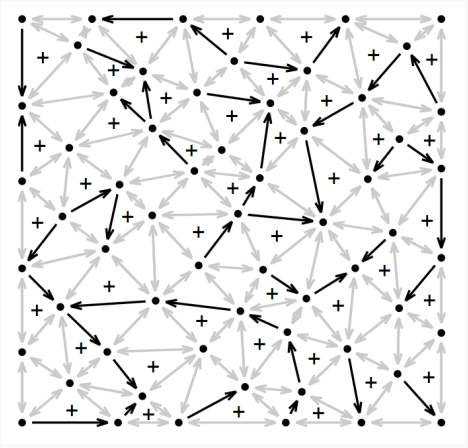
numerical results: one-level (power) iteration for random graph problem

- start from random intial guess $\mathbf{x}_0$
- let $A = D - (L + U)$
- iterate on $\mathbf{x}_{i+1} = (I + w\, D^{-1}\, A)\, \mathbf{x}_i$

  with $w = 0.7$

  until $\dfrac{\|A\,\mathbf{x}_i\|_1}{\|A\,\mathbf{x}_0\|_1} < 10^{-8}$

- W=O(n^2) method
  (A sparse, O(n) iterations)

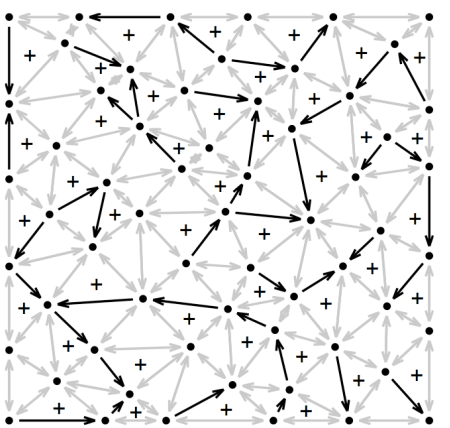


n	it
128	322
256	494
512	1010
1024	1768

NII Tokyo
hdesterck@uwaterloo.ca

when is power method slow?

- power method is slow on graphs with local links, $|\lambda_2| \approx 1$
- power method is fast on graphs with global links, short distances
- PageRank is fast mixing:
  you can just do power method
  (PageRank is made fast by artificially adding global links from all webpages to all webpages with probability 0.15)

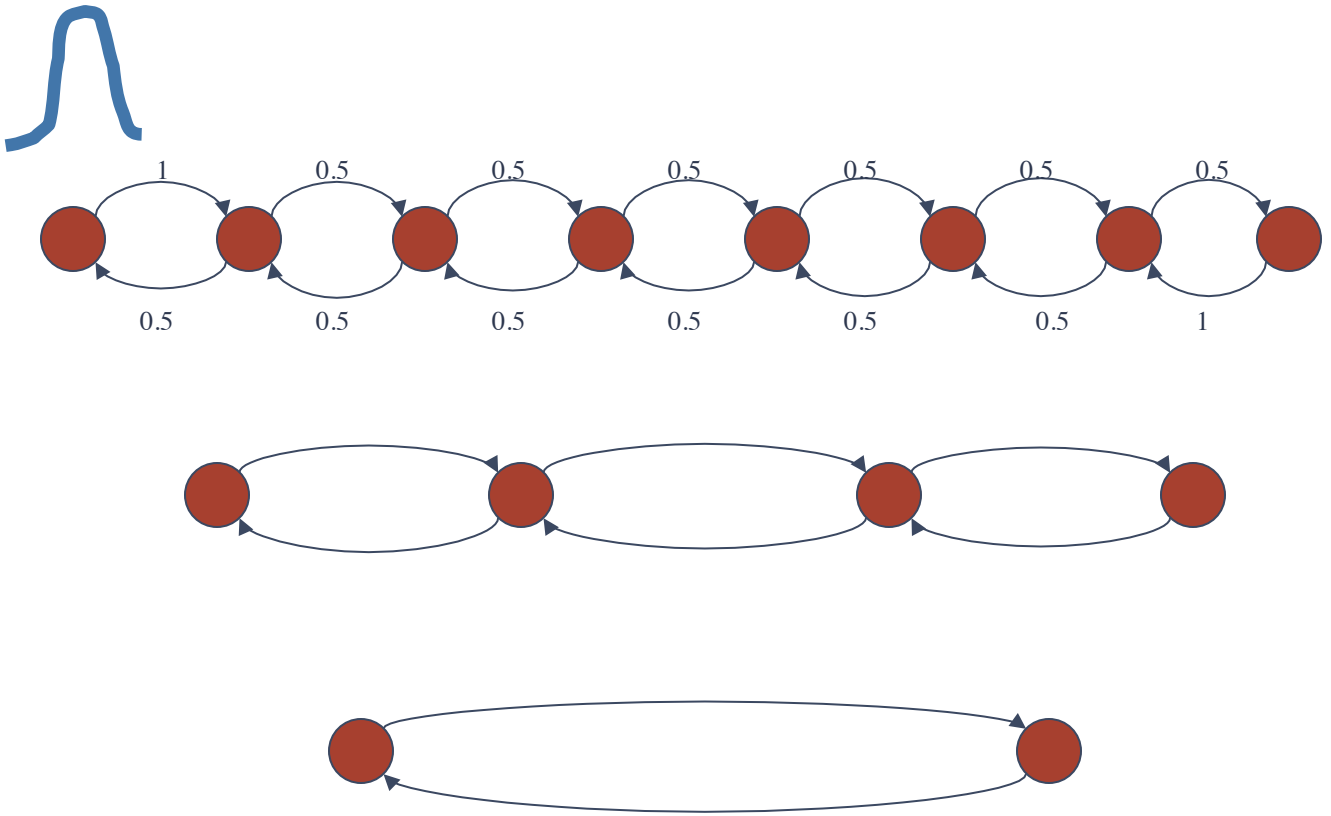


n	it
128	322
256	494
512	1010
1024	1768

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

why/when is power method slow?
why multilevel methods?

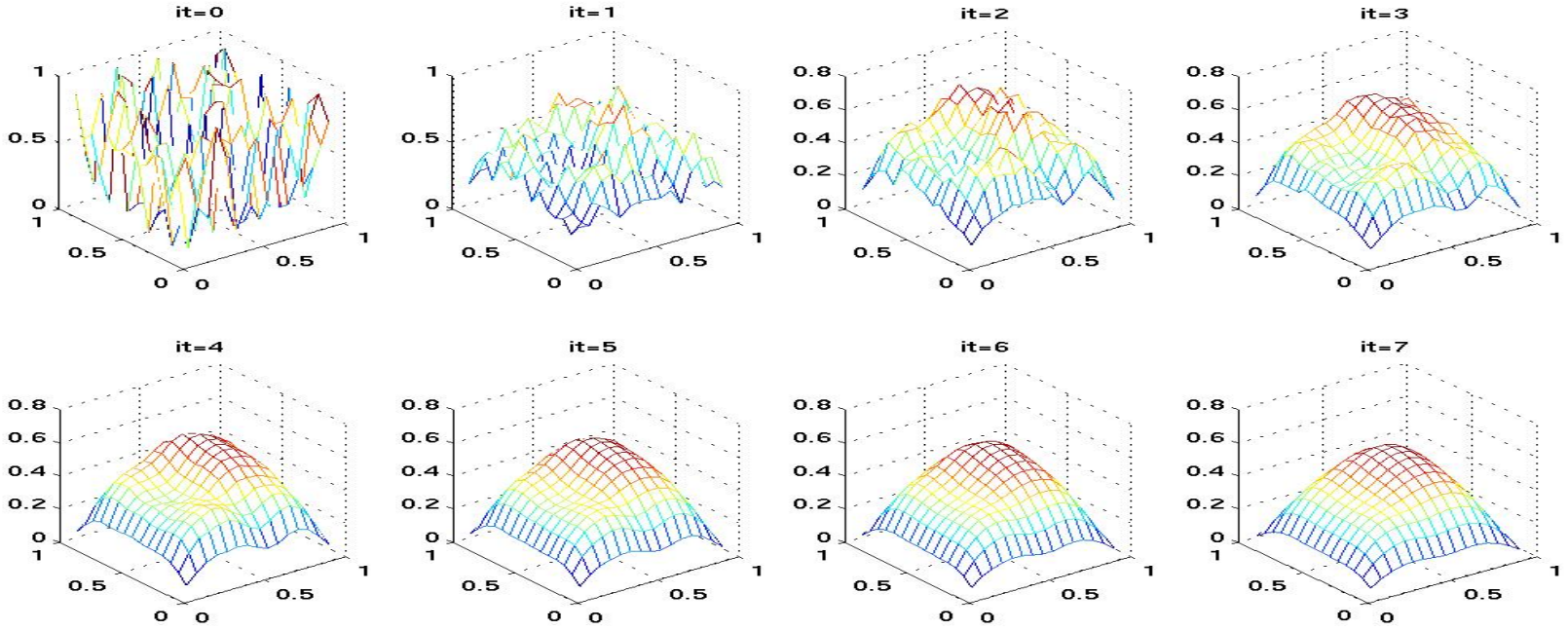


University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

principle of multigrid (for PDEs)

$$-u_{xx} - u_{yy} = f(x,y) \qquad Ax = b$$

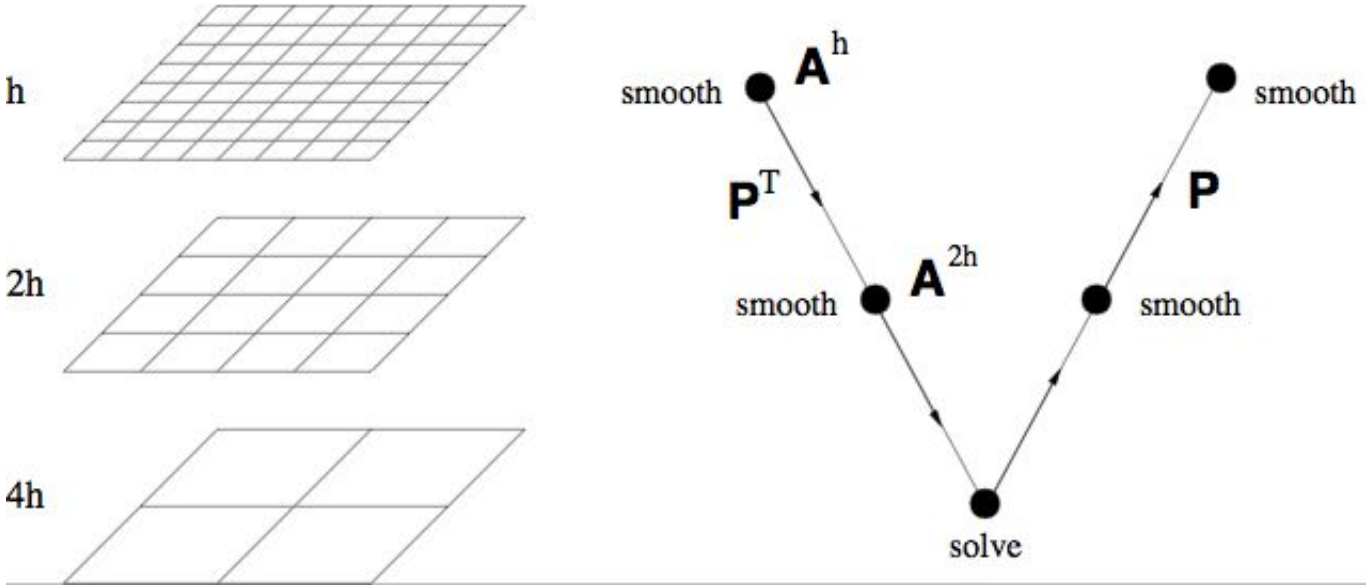


- high-frequency error is removed by relaxation (weighted Jacobi, Gauss-Seidel, ... power method)
- low-frequency-error needs to be removed by coarse-grid correction

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

multigrid hierarchy: V-cycle



- multigrid V-cycle:
  - relax (=smooth) on successively coarser grids
  - transfer error using restriction ($R=P^T$) and interpolation (P)
- W=O(n) : (optimally) scalable method

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

4. aggregation for Markov chains

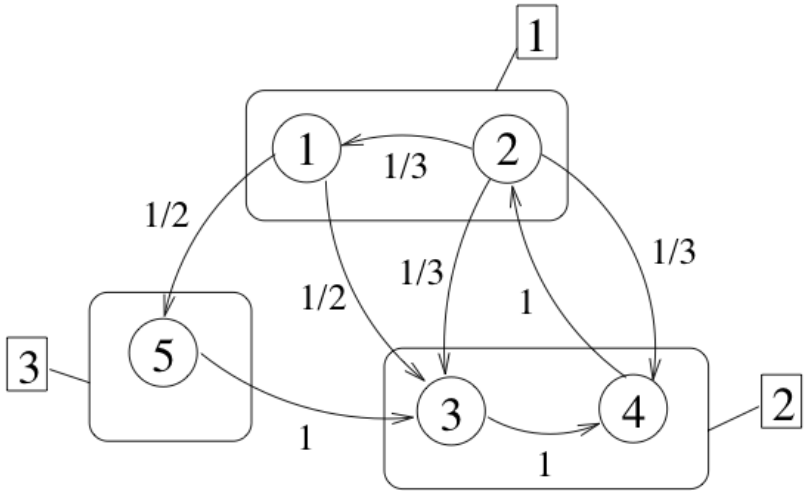
- form three coarse, aggregated states

$$x_{c,I} = \sum_{i \in I} x_i$$

$$\mathbf{x}_c^T = [8/19 \;\; 10/19 \;\; 1/19]$$

$$B_c \mathbf{x}_c = \mathbf{x}_c$$

$$b_{c,IJ} = \frac{\sum_{j \in J} x_j \left(\sum_{i \in I} b_{ij} \right)}{\sum_{j \in J} x_j}$$



$$B_c = \begin{bmatrix} 1/4 & 3/5 & 0 \\ 5/8 & 2/5 & 1 \\ 1/8 & 0 & 0 \end{bmatrix}$$

(Simon and Ando, 1961)

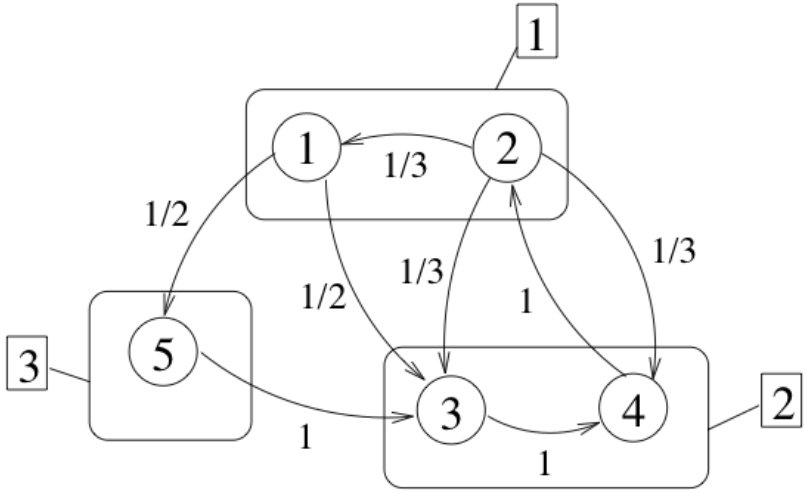
University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

aggregation for Markov chains

$$B_c \, \mathbf{x}_c = \mathbf{x}_c$$

$$b_{c,IJ} = \frac{\sum_{j \in J} x_j \left(\sum_{i \in I} b_{ij} \right)}{\sum_{j \in J} x_j}$$



$$B_c = Q^T \, B \, \mathrm{diag}(\mathbf{x}) \, Q \, \mathrm{diag}(Q^T \, \mathbf{x})^{-1}$$

$$x_{c,I} = \sum_{i \in I} x_i$$

$$\mathbf{x}_c = Q^T \, \mathbf{x}$$

$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

two-level aggregation method

repeat

$$\text{fine-level relaxation: } \mathbf{x}^* = B\,\mathbf{x}_i$$
$$\text{build } Q$$
$$\text{build } B_c = Q^T B\,\mathrm{diag}(\mathbf{x}^*)\,Q\,(\mathrm{diag}(Q^T\,\mathbf{x}^*))^{-1}$$
$$\text{coarse-level solve: } B_c\,\mathbf{x}_c = \mathbf{x}_c$$
$$\text{fine-level update: } \mathbf{x}_{i+1} = \mathrm{diag}(\mathbf{x}^*)\,Q\,(\mathrm{diag}(Q^T\,\mathbf{x}^*))^{-1}\mathbf{x}_c$$

(similar to lumping method from Takahashi, 1975)
('iterative aggregation/disaggregation')
(note: there is a convergence proof for this two-level method, Marek and Mayer 1998, 2003)

NII Tokyo
hdesterck@uwaterloo.ca

A LUMPING METHOD FOR NUMERICAL CALCULATIONS OF STATIONARY DISTRIBUTIONS OF MARKOV CHAINS

Yukio TAKAHASHI

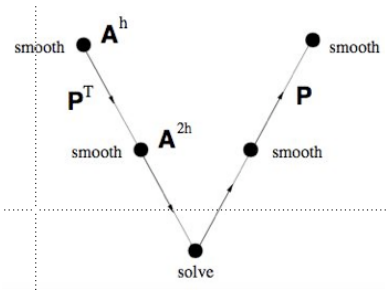
Classification: Numerical Methods in Markov Chains

June 1975, No. B - 18

Department of Information Sciences
Tokyo Institute of Technology

multilevel aggregation algorithm

Algorithm: Multilevel Adaptive Aggregation method (V-cycle)



$\mathbf{x} = \text{AM\_V}(A, \mathbf{x}, \nu_1, \nu_2)$

begin

$\mathbf{x} \leftarrow \text{Relax}(A, \mathbf{x})$ ν_1 times

build Q based on $\mathbf{x}$ and A (Q is rebuilt every level and cycle)

$R = Q^T$ and $P = \text{diag}(\mathbf{x})\, Q$

$A_c = R\, A\, P$

$\mathbf{x}_c = \text{AM\_V}(A_c\, \text{diag}(P^T 1)^{-1}, P^T 1, \nu_1, \nu_2)$ (coarse-level solve)

$\mathbf{x} = P\, (\text{diag}(P^T 1))^{-1} \mathbf{x}_c$ (coarse-level correction)

$\mathbf{x} \leftarrow \text{Relax}(A, \mathbf{x})$ ν_2 times

end

(Krieger, Horton 1994, but no good way to build Q, convergence not good)

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

well-posedness: singular M-matrices

- singular M-matrix:

$$A = \begin{bmatrix} + & - & - & - & - \\ - & + & - & - & - \\ - & - & + & - & - \\ - & - & - & + & - \\ - & - & - & - & + \end{bmatrix}$$

$$A \in \mathbb{R}^{n \times n} \text{ is a singular M-matrix} \Leftrightarrow$$

$$\exists\, B \in \mathbb{R}^{n \times n},\ b_{ij} \geq 0 \,\forall i, j\ :\ A = \rho(B)\, I - B$$

- our *A=I-B* is a singular M-matrix on all levels

(1) Irreducible singular M-matrices have a unique solution to the problem $A\mathbf{x} = 0$, *up to scaling. All components of* $\mathbf{x}$ *have strictly the same sign (i.e., scaling can be chosen s.t.* $x_i > 0\,\forall i$*). (This follows directly from the Perron-Frobenius theorem.)*

(3) Irreducible singular M-matrices have nonpositive off-diagonal elements, and strictly positive diagonal elements $(n > 1)$.

(4) If A has a strictly positive element in its left or right nullspace and the off-diagonal elements of A are nonpositive, then A is a singular M-matrix (see also [21]).

NII Tokyo
hdesterck@uwaterloo.ca

well-posedness: multilevel method

THEOREM 3.1 (Singular M-matrix property of AM coarse-level operators). *A_c is an irreducible singular M-matrix on all coarse levels, and thus has a unique right kernel vector $\mathbf{e}_c$ with strictly positive components (up to scaling) on all levels.*

(1) $\mathbf{1}_c^T A_c = 0 \quad \forall\, \mathbf{x}_i$
(since $\mathbf{1}_c^T R = \mathbf{1}^T$ and $\mathbf{1}^T A = 0$)

$$A = \begin{bmatrix} + & - & - & - & - \\ - & + & - & - & - \\ - & - & + & - & - \\ - & - & - & + & - \\ - & - & - & - & + \end{bmatrix}$$

THEOREM 3.2 (Fixed-point property of AM V-cycle). *Exact solution $\mathbf{x}$ is a fixed point of the AM V-cycle.*

(2) $A_c\, \mathbf{1}_c = 0 \quad$ for $\mathbf{x}_i = \mathbf{x}$

$$A_c\, \mathbf{e}_c = 0$$
$$\mathbf{x}_{i+1} = P\, \mathbf{e}_c$$

NII Tokyo
hdesterck@uwaterloo.ca

aggregation strategy

- fine-level relaxation should efficiently distribute probability within aggregates (smooth out local, high-frequency errors)
- coarse-level update will efficiently distribute probability between aggregates (smooth out global, low-frequency errors)
- base aggregates on ‘strong connections’ in $A\,\mathrm{diag}(\mathbf{x}_i)$

NII Tokyo
hdesterck@uwaterloo.ca

aggregation strategy

scaled problem matrix:

$$\hat{A} = A\,\mathrm{diag}(\mathbf{x}_i)$$

strong connection: coefficient is large in either of rows *i* or *j*

$$-\hat{a}_{ij} \geq \theta \max_{k \neq i}\{-\hat{a}_{ik}\} \quad \text{or} \quad -\hat{a}_{ji} \geq \theta \max_{k \neq j}\{-\hat{a}_{jk}\}$$

($\theta \in (0,1)$, θ=0.25)

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

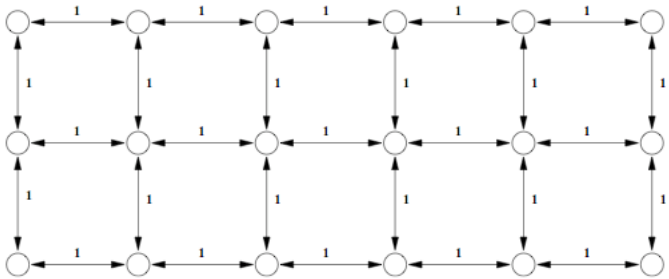
‘neighbourhood’ aggregation strategy

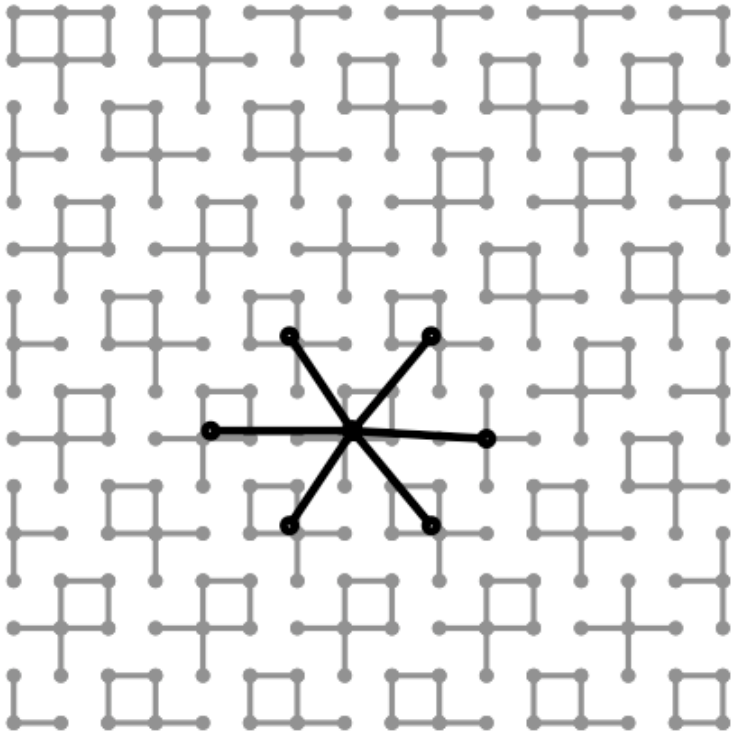
Algorithm 2: neighborhood-based aggregation, $\{Q_J\}_{J=1}^{m} \longleftarrow$ **NeighbourhoodAgg** $(A\,\mathrm{diag}(\mathbf{x}),\, \theta)$

For all points i, build strong neighbourhoods $\mathcal{N}_i$ based on $A\,\mathrm{diag}(\mathbf{x})$ and θ.
Set $\mathcal{R} \leftarrow \{1, ..., n\}$ and $J \leftarrow 0$.
`/* 1st pass: assign entire neighborhoods to aggregates */`
for $i \in \{1, ..., n\}$ **do**
  if $(\mathcal{R} \cap \mathcal{N}_i) = \mathcal{N}_i$ **then**
    $J \leftarrow J + 1$.
    $Q_J \leftarrow \mathcal{N}_i$, $\hat{Q}_J \leftarrow \mathcal{N}_i$.
    $\mathcal{R} \leftarrow \mathcal{R} \setminus \mathcal{N}_i$.
  end
end
$m \leftarrow J$.
`/* 2nd pass: put remaining points in aggregates they are most connected to */`
while $\mathcal{R} \neq \emptyset$ **do**
  Pick $i \in \mathcal{R}$ and set $J \leftarrow \mathrm{argmax}_{K=1,...,m}\, \mathrm{card}\,(\mathcal{N}_i \cap Q_K)$.
  Set $\hat{Q}_J \leftarrow Q_J \cup \{i\}$ and $\mathcal{R} \leftarrow \mathcal{R} \setminus \{i\}$.
end
for $J \in \{1, ..., m\}$ **do** $Q_J \leftarrow \hat{Q}_J$.

NII Tokyo
hdesterck@uwaterloo.ca

aggregation: periodic 2D lattice



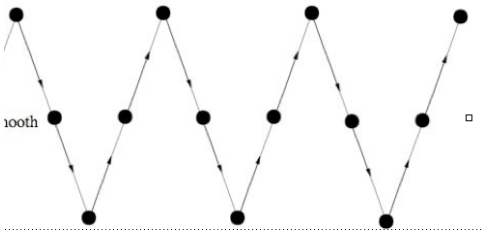


$$B_c = Q^T B \, \mathrm{diag}(\mathbf{x}^*) \, Q \, (\mathrm{diag}(Q^T \mathbf{x}^*))^{-1}$$

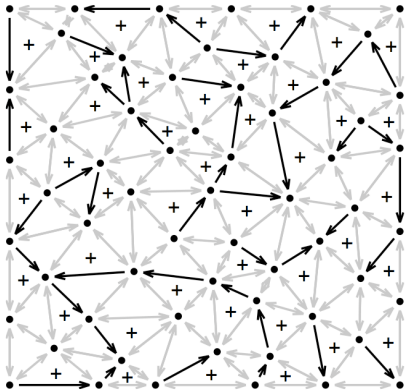
University of Waterloo

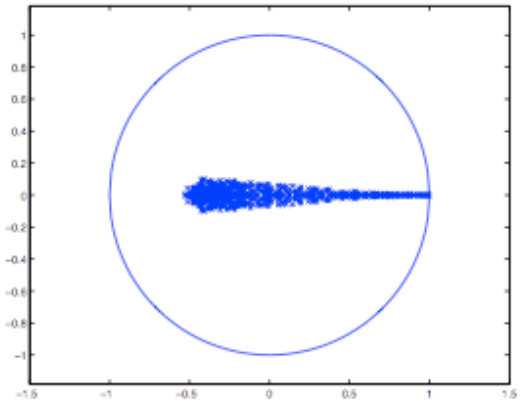
NII Tokyo
hdesterck@uwaterloo.ca

numerical results: aggregation multigrid for random walk problem



	1-level	aggregation		
n	iterations	iterations	C_{op}	levels
128	322	95	1.12	3
256	494	107	1.13	3
512	1010	156	1.14	3
1024	1768	220	1.15	4
2048		352	1.15	4





$$C_{op} = \frac{\sum_{l=0} \text{nonzeros}(A_l)}{nonzeros(A_0)}$$

does not work so well yet (not O(n) ...)

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

our work since 2006: speed up the multilevel aggregation method

goal: W=O(n) (number of V-cycles independent of n)

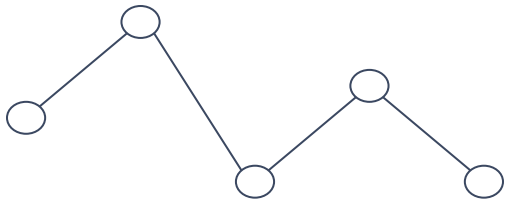
1. smoothed aggregation (SIAM J. Sc. Comp., submitted 2008)
2. build P by algebraic multigrid (SIAM J. Sc. Comp., submitted 2009)
3. recursive iterant recombination (SIAM J. Sc. Comp., submitted 2009)
4. overcorrection (NLAA, submitted 2010)

(inspired by algebraic multigrid for PDEs)

NII Tokyo
hdesterck@uwaterloo.ca

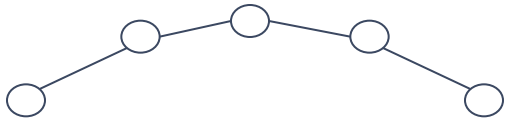
5. overlapping aggregates: we need ‘smoothed aggregation’...

(Vanek, Mandel, and Brezina, Computing, 1996)

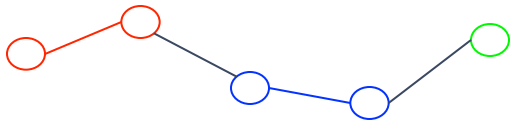


$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

after smoothing:

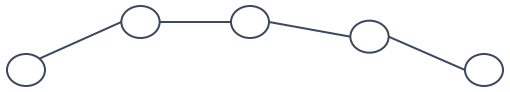


coarse grid correction with Q:



$$Q_s = \begin{bmatrix} \times & 0 & 0 \\ \times & \times & 0 \\ \times & \times & 0 \\ 0 & \times & \times \\ 0 & \times & \times \end{bmatrix}$$

coarse grid correction with Q_s:



University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

smoothed aggregation

$$A_c = Q^T A \, \mathrm{diag}(\mathbf{x}_i) \, Q = R A P$$

- smooth the columns of *P* with weighted Jacobi:

$$P_s = (I + w D^{-1} A) \, \mathrm{diag}(\mathbf{x}_i) \, Q$$

$$w = 0.7$$

- smooth the rows of *R* with weighted Jacobi:

$$R_s = Q^T (I + w A D^{-1})$$

NII Tokyo
hdesterck@uwaterloo.ca

smoothed aggregation: a problem with signs

- smoothed coarse level operator:

$$\begin{aligned} A_{cs} &= R_s \left(D - (L+U) \right) P_s \\ &= R_s D P_s - R_s (L+U) P_s \end{aligned}$$

$$A = \begin{bmatrix} + & - & - & - & - \\ - & + & - & - & - \\ - & - & + & - & - \\ - & - & - & + & - \\ - & - & - & - & + \end{bmatrix}$$

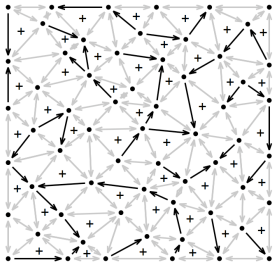
- problem: A_{cs} is not a singular M-matrix (signs wrong)
- solution:
  lumping approach
- well-posedness of
  this approach shown
  in our paper

$$S_{\{i,j\}} = \begin{array}{c} \\ i \\ \\ j \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccc} & i & & j & \end{array} \\ \begin{bmatrix} \ddots & \vdots & & \vdots & \\ \cdots & \beta_{\{i,j\}} & \cdots & -\beta_{\{i,j\}} & \cdots \\ & \vdots & & \vdots & \\ \cdots & -\beta_{\{i,j\}} & \cdots & \beta_{\{i,j\}} & \cdots \\ & \vdots & & \vdots & \end{bmatrix} \end{array}$$

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

numerical results: smoothed aggregation multigrid for random graph problem



	1-level	aggregation			smoothed aggregation			
n	iterations	iterations	C_{op}	levels	iterations	C_{op}	levels	R_{lump}
128	322	95	1.12	3				
256	494	107	1.13	3				
512	1010	156	1.14	3	36	1.28	3	2.5e-4
1024	1768	220	1.15	4	39	1.31	4	1.2e-4
2048		352	1.15	4	33	1.31	4	6.0e-5
4096					46	1.35	4	2.3e-4
8192					35	1.37	4	2.0e-4
16384					51	1.36	5	9.4e-5
32768					43	1.38	5	1.6e-4

$$C_{op} = \frac{\sum_{l=0} \text{nonzeros}(A_l)}{nonzeros(A_0)}$$

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

6. algebraic multigrid for Markov chains

- scaled problem matrix: $\bar{A} := A\,\mathrm{diag}(\mathbf{x}_i)$
- multiplicative error equation: $\bar{A}\,\mathbf{e}_i = \mathbf{0}$

$$\text{At convergence, } \mathbf{1} \text{ lies in the nullspace of } \bar{A}$$

- we can use ‘standard’ AMG on $\bar{A}\,\mathbf{e}_i = \mathbf{0}$
- define AMG coarsening and interpolation

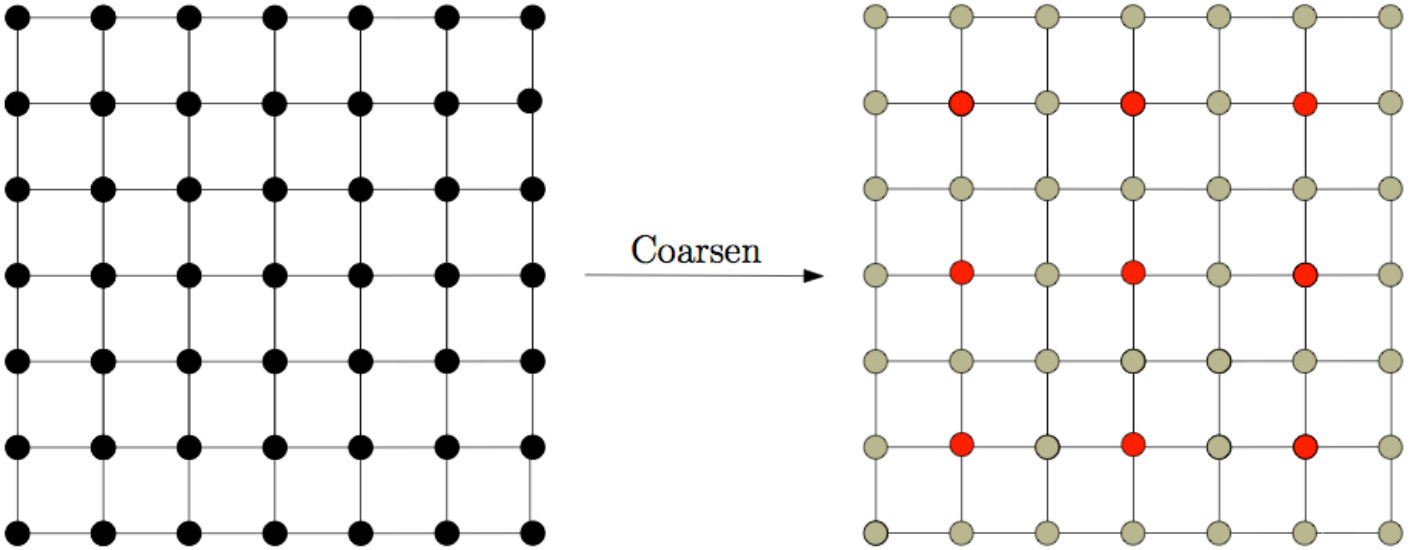
$$R\,\bar{A}\,P\,\mathbf{e}_c = \mathbf{0} \quad \text{or} \quad \bar{A}_c\,\mathbf{e}_c = \mathbf{0} \qquad R = P^T$$

- lumping can be done as for smoothed aggregation

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

AMG (two-pass) coarsening and interpolation



$$(P\,\mathbf{e}_c)_i = \begin{cases} (\mathbf{e}_c)_i & \text{if } i \in C, \\ \sum_{j \in C_i} w_{ij} (\mathbf{e}_c)_j & \text{if } i \in F. \end{cases}$$

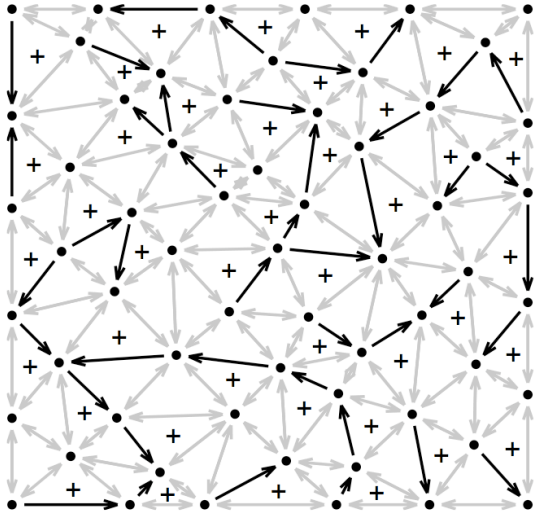
$$\bar{w}_{ij} = \frac{\bar{a}_{ij} + \sum_{m \in D_i^s} \left(\frac{\bar{a}_{im} \bar{a}_{mj}}{\sum_{k \in C_i} \bar{a}_{mk}} \right)}{\sum_{j \in C_i} \bar{a}_{ij} + \sum_{r \in D_i^s} \bar{a}_{ir}}$$

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

unstructured planar graph

	MCAMG					
n	γ	it	C_{op}	γ_{eff}	lev	R_l
1024	0.40	15	2.13	0.65	6	0
2048	0.33	14	2.22	0.61	7	6.3e-5
4096	0.40	15	2.19	0.66	7	6.3e-5
8192	0.40	15	2.25	0.66	8	9.3e-5
16384	0.37	14	2.26	0.65	9	7.0e-5
32768	0.37	14	2.28	0.65	9	1.3e-4



University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

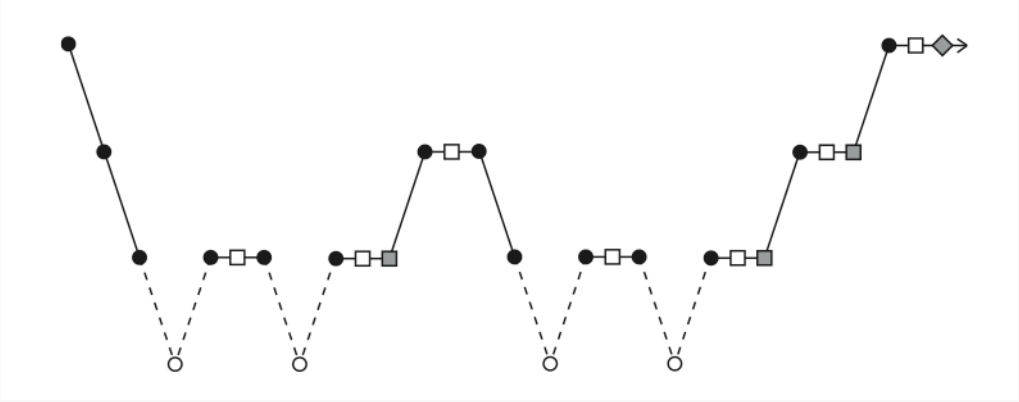
7. recursively accelerated (pure) aggregation

- idea: recombine iterates at all levels in W cycle

[27] T. WASHIO AND C.W. OOSTERLEE, *Krylov subspace acceleration for nonlinear multigrid schemes*, Electronic Transactions on Numerical Analysis 6:271-290, 1997.

[19] Y. NOTAY AND P.S. VASSILEVSKI, *Recursive Krylov-based multigrid cycles*, Numer. Lin. Alg. Appl. 15:473-487, 2008.

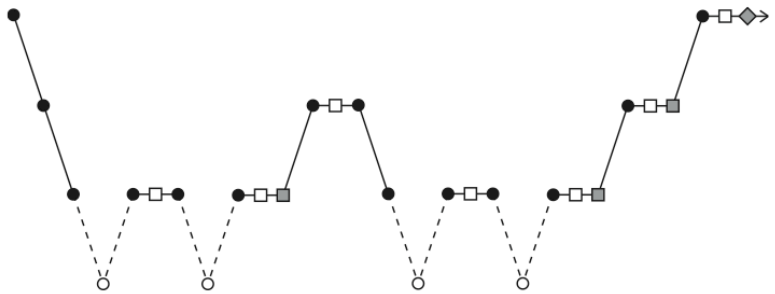
[20] Y. NOTAY, *An aggregation-based algebraic multigrid method*, Report GANMN 08-02, Universit Libre de Bruxelles, Brussels, Belgium, 2009.



University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

recursively accelerated (pure) aggregation



- for *Ax=b*, use recursive Krylov acceleration
- for Markov: need to impose probability constraints

$$\mathbf{w} = z_1\,\mathbf{x}_1 + z_2\,\mathbf{x}_2 = \hat{X}\,\mathbf{z}$$

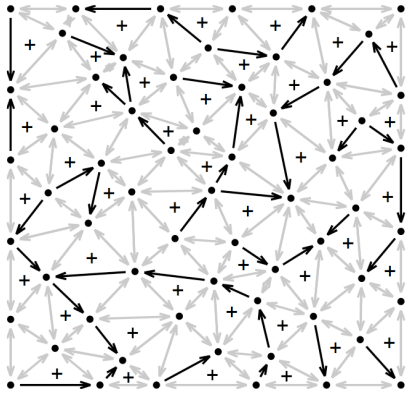
$$\begin{aligned} &\mathbf{w}^* = \operatorname{argmin}_{\mathbf{w}} \ \|A\mathbf{w}\|_2 \\ \text{subject to:}\quad &\mathbf{w} \geq 0 \quad \text{and} \\ &\|\mathbf{w}\|_1 = 1, \end{aligned} \qquad \begin{aligned} &\mathbf{z}^* = \operatorname{argmin}_{\mathbf{z}} \ \|(A\,\hat{X})\mathbf{z}\|_2 \\ \text{subject to:}\quad &\hat{X}\,\mathbf{z} \geq 0 \quad \text{and} \\ &\mathbf{1}^T\,\mathbf{z} = 1. \end{aligned}$$

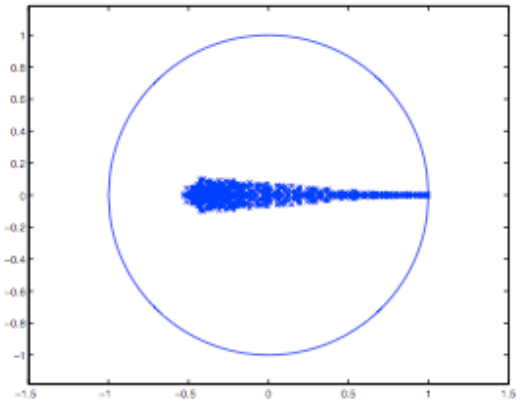
- standard quadratic programming problem

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

unstructured planar graph





	W cycles		RAMA cycles		W+ cycles		RAMA+ cycles	
n	it	C_{op}	it	C_{op}	it	C_{op}	it	C_{op}
1024	113	1.32	61	1.32	38	1.32	28	1.32
2048	152	1.33	70	1.33	35	1.33	27	1.33
4096	180	1.35	75	1.35	52	1.35	31	1.35
8192	201	1.36	78	1.36	39	1.36	26	1.36
16384	214	1.36	67	1.36	43	1.36	27	1.36
32768	301	1.37	87	1.37	47	1.37	28	1.37

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

8. over-correction, and ‘frozen’ additive cycles

- (with Eran Treister and Irad Yavneh)
- idea: ‘shape’ of correction is often good, but ‘amplitude’ may be too small

  therefore, overcorrect with factor α

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha P \mathbf{e}_c$$

- determine optimal α automatically

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

‘frozen’ additive cycles

- idea: replace expensive ‘multiplicative’ cycles by cheap ‘frozen’ additive cycles (as soon as good convergence)

build Q based on $\mathbf{x}$ and A

$R = Q^T$ and $P = \mathrm{diag}(\mathbf{x})\, Q$

$A_c = R\, A\, P$

- can do this ‘on-the-fly’ (OTF)
- can lead to large speed gains

NII Tokyo
hdesterck@uwaterloo.ca

‘frozen’ additive cycles

multiplicative formulation: $\mathbf{e}_c$

$$RAP\,\mathbf{e}_c = 0$$

$$\mathbf{x}_{i+1} = P\,\mathbf{e}_c$$

$$\mathbf{x}_i = P\,\mathbf{e}_{c,i}$$

additive formulation: $\hat{\mathbf{e}}_c$

$$RAP\,\hat{\mathbf{e}}_c = R\,\mathbf{r}_i$$

$$\mathbf{x}_{i+1} = \mathbf{x}_i + P\,\hat{\mathbf{e}}_c$$

$$\mathbf{r}_i = -A\,\mathbf{x}_i$$

equivalent via: $\hat{\mathbf{e}}_c = \mathbf{e}_c - \mathbf{e}_{c,i}$

$$RAP\,\hat{\mathbf{e}}_c = RAP\,(\mathbf{e}_c - \mathbf{e}_{c,i}) = -RAP\,\mathbf{e}_{c,i} = -RA\,\mathbf{x}_i = R\,\mathbf{r}_i$$

$$\mathbf{x}_{i+1} = P\,\mathbf{e}_c = P\,\hat{\mathbf{e}}_c + P\,\mathbf{e}_{c,i} = \mathbf{x}_i + P\,\hat{\mathbf{e}}_c$$

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

tandem queuing network

Table I. Tandem queueing network.

	one-level ($\omega = 0.99$)	two-level	Arnoldi		GMRES	
n	it	it	it	WU	it	WU
1024	5582	902	25	911	25	932
4096	12005	3383	25	880	25	900
16384			25	1231	25	1185
65536			25	1494	25	1484
262144			50	2313	50	2320

FIG. 5.6. *Tandem queueing network.*

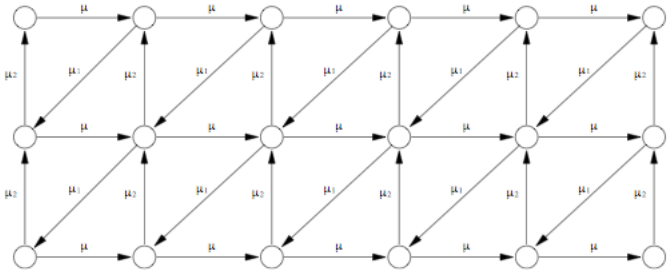


FIG. 5.7. *Graph for tandem queueing network.*

Table II. Tandem queueing network.

	AGG			OTF-AGG			MCAMG			OTF-MCAMG		
n	it	(C_{op})	WU	it	(C_{op})	WU	it	(C_{op})	WU	it	(C_{op})	WU
4096	159	(1.48)	2589	169	(1.48)	1897	11	(4.52)	5113	13	(4.53)	1478
16384	272	(1.49)	3257	325	(1.49)	2314	13	(4.56)	6194	15	(4.57)	1689
65536	465	(1.50)	6578	719	(1.50)	5795	13	(4.61)	4755	20	(4.65)	1411
262144							14	(4.65)	4224	24	(4.67)	1270

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

tandem queuing network

Table II. Tandem queueing network.

	AGG			OTF-AGG			MCAMG			OTF-MCAMG		
n	it	(C_{op})	WU	it	(C_{op})	WU	it	(C_{op})	WU	it	(C_{op})	WU
4096	159	(1.48)	2589	169	(1.48)	1897	11	(4.52)	5113	13	(4.53)	1478
16384	272	(1.49)	3257	325	(1.49)	2314	13	(4.56)	6194	15	(4.57)	1689
65536	465	(1.50)	6578	719	(1.50)	5795	13	(4.61)	4755	20	(4.65)	1411
262144							14	(4.65)	4224	24	(4.67)	1270

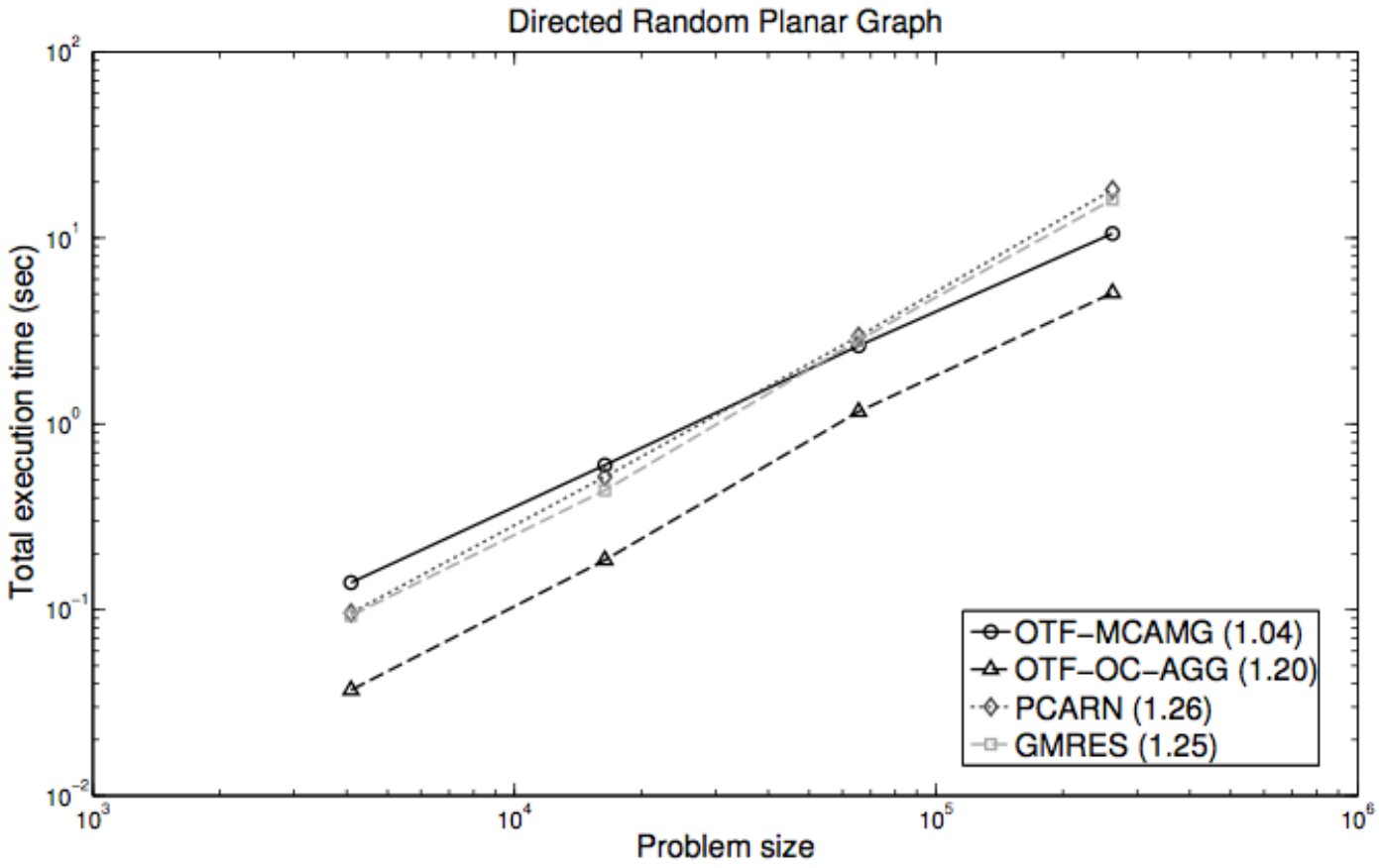
Table III. Tandem queueing network.

	OC-AGG (automatic)			OC-AGG (fixed - 1.9)			OTF-OC-AGG (automatic)			OTF-OC-AGG (fixed - 1.9)		
n	it	(C_{op})	WU	it	(C_{op})	WU	it	(C_{op})	WU	it	(C_{op})	WU
4096	16	(1.48)	260	17	(1.48)	277	36	(1.48)	297	40	(1.48)	292
16384	19	(1.49)	225	22	(1.49)	261	44	(1.49)	285	43	(1.49)	246
65536	23	(1.50)	339	23	(1.50)	321	60	(1.50)	428	59	(1.50)	392
262144	26	(1.50)	409	26	(1.50)	393	–	–	–	–	–	–

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

unstructured planar graph



University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

9. conclusions

- algebraic multilevel methods can lead to W=O(n) solvers for slowly mixing Markov chains
- we have developed several ways to accelerate ‘pure’ aggregation methods such that W=O(n) is reached
  - smoothed aggregation
  - algebraic multigrid
  - recursive iterant recombination
  - over-correction (and frozen additive cycles)

  (these approaches are inspired on multigrid for PDEs)

NII Tokyo
hdesterck@uwaterloo.ca

conclusions

- theory is very hard because the systems are non-symmetric
- our methods will not be fast for:
  - high-dimensional lattices, queues, tensor-product structure
  - fast mixing Markov chains

  (but we can handle unstructured Markov chains)
- good results are obtained for many slowly mixing Markov chains
- multilevel methods can be very powerful

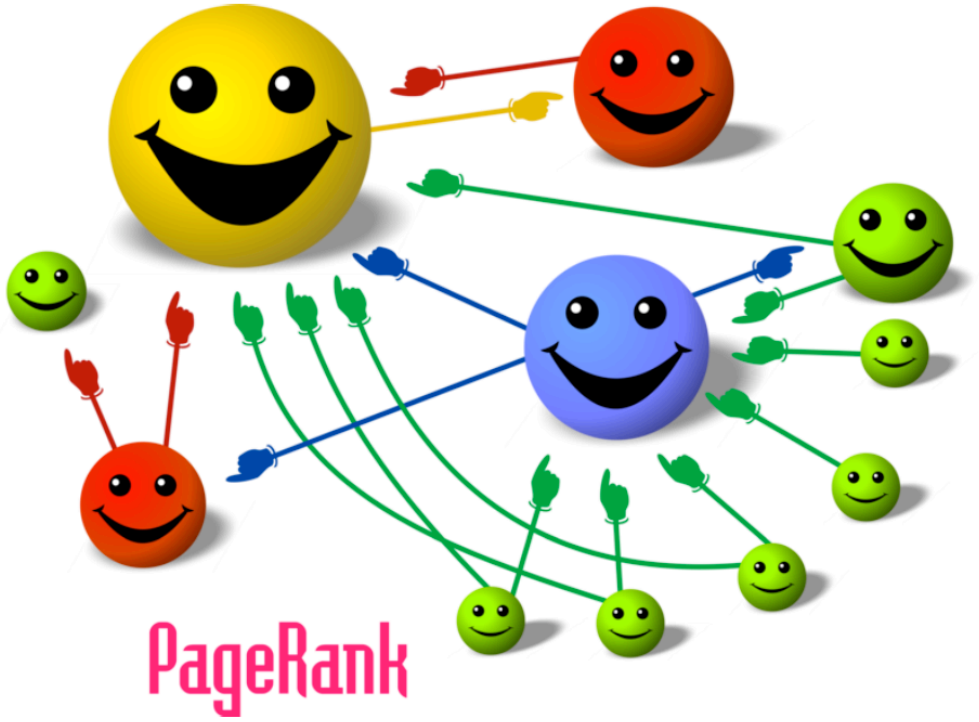
NII Tokyo
hdesterck@uwaterloo.ca

thank you
questions?

University of Waterloo

NII Tokyo
hdesterck@uwaterloo.ca

6. Test Problems

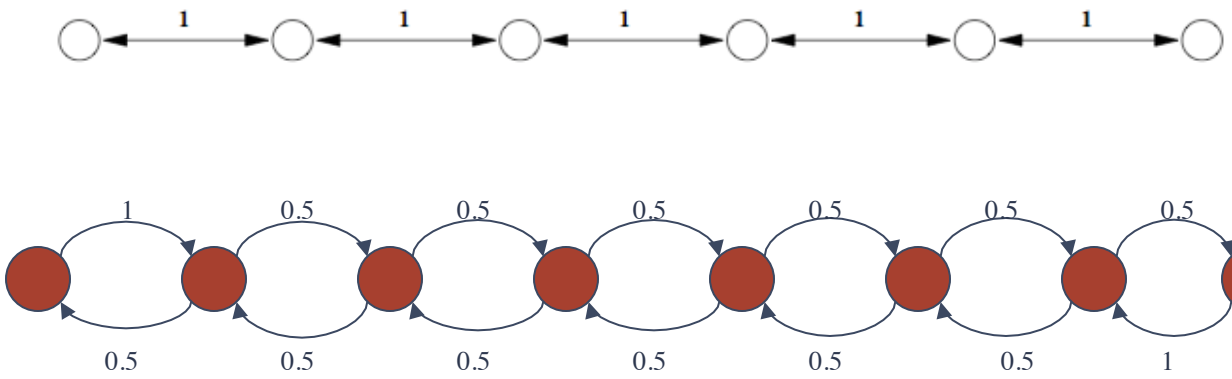


(De Sterck et al., SISC, 2008, ‘Multilevel adaptive aggregation for Markov chains, with application to web ranking’)

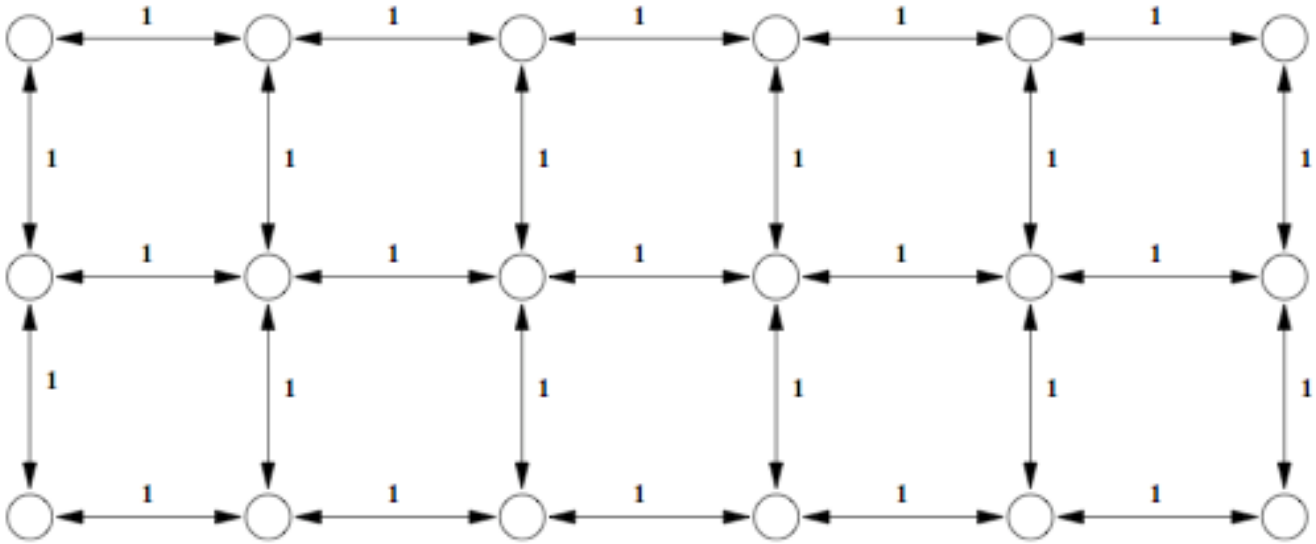
hdesterck@uwaterloo.ca

6.1 Uniform 1D Chain

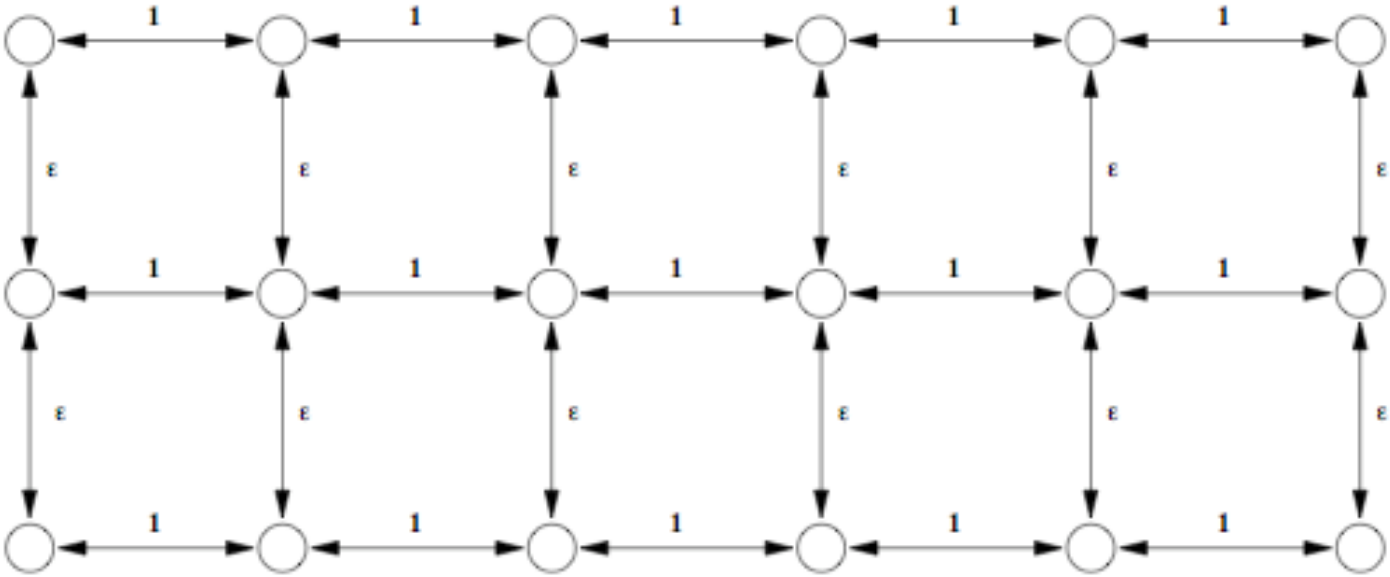
- random walk on (undirected) graph
- all edges have the same weight
- transition probability for directed edge =
  weight of edge / sum of weights of outgoing edges
- solution trivial - test problem
- random walk on undirected graph gives real-spectrum *B*



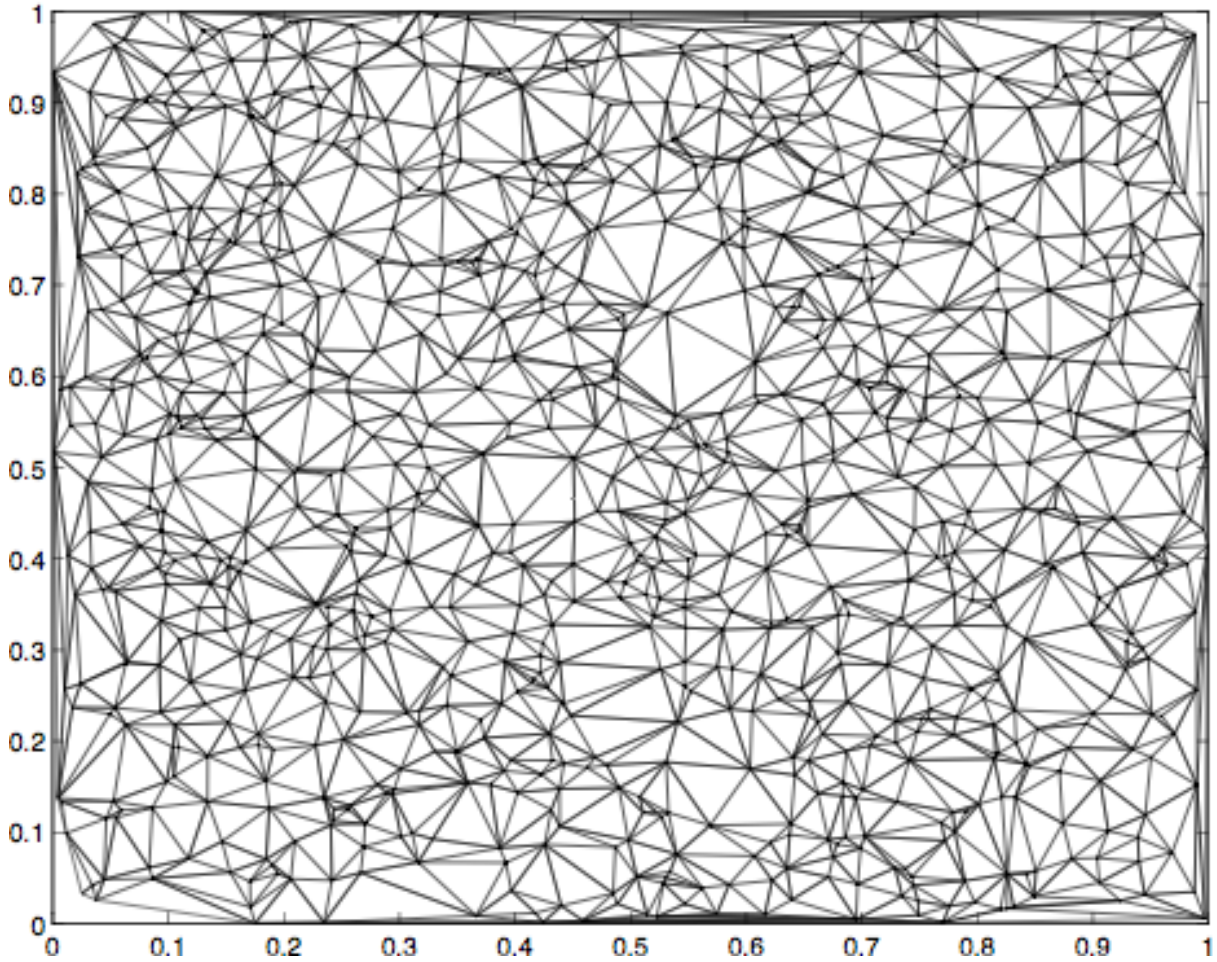
6.2 Uniform 2D Lattice



6.3 Anisotropic 2D Lattice



6.4 Unstructured Planar Graph



University of Waterloo

hdesterck@uwaterloo.ca

Size of Subdominant Eigenvalue

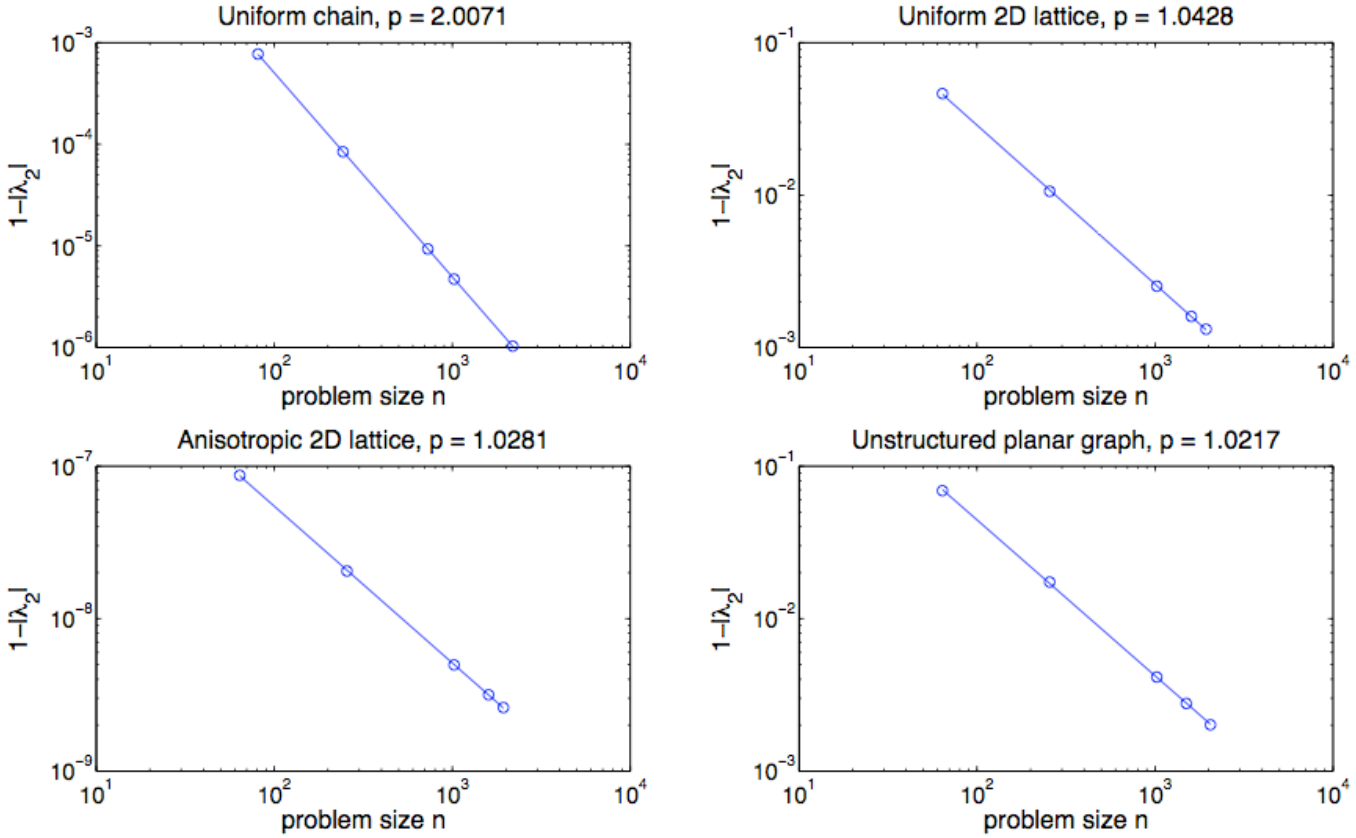


FIG. 5.1. *Magnitude of subdominant eigenvalue as a function of problem size.*

6.5 Tandem Queueing Network

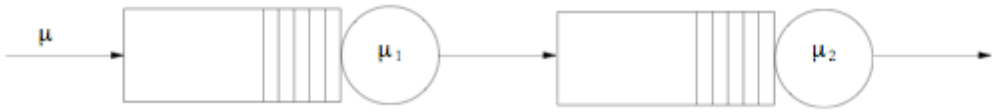


FIG. 5.6. *Tandem queueing network.*

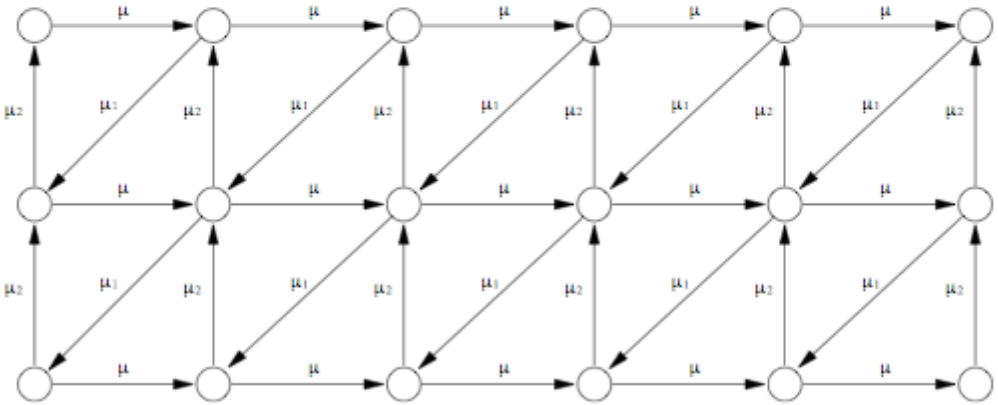
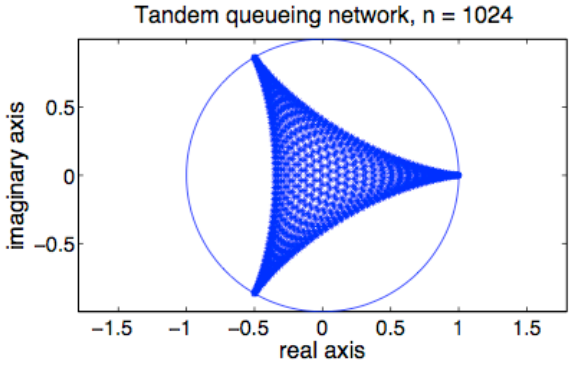
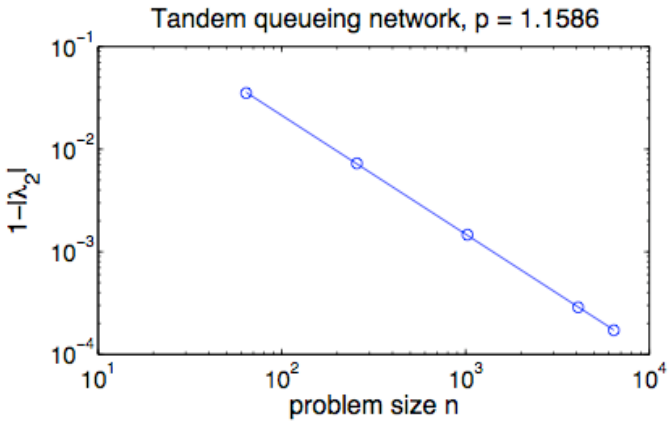


FIG. 5.7. *Graph for tandem queueing network.*

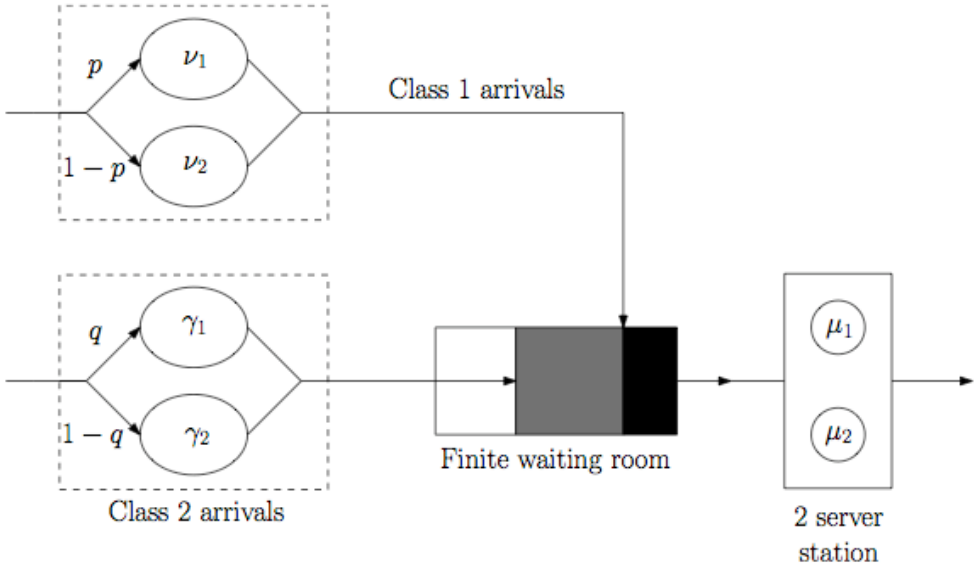


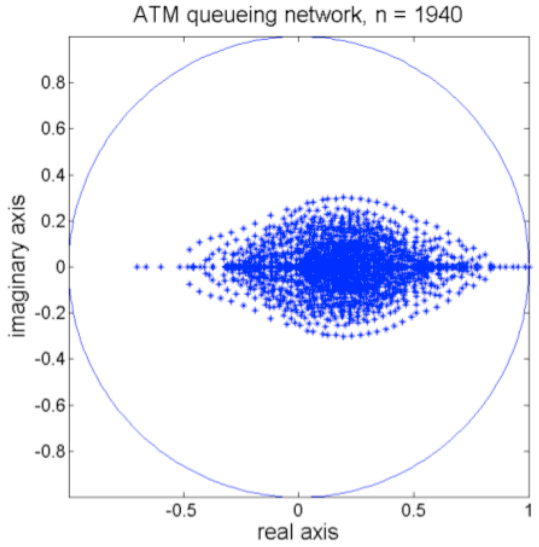


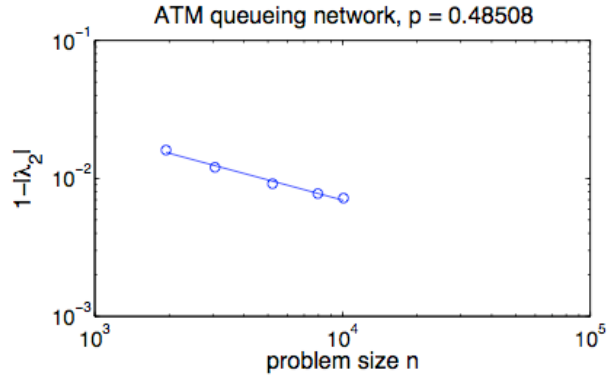
University of Waterloo

hdesterck@uwaterloo.ca

6.6 ATM Queueing Network







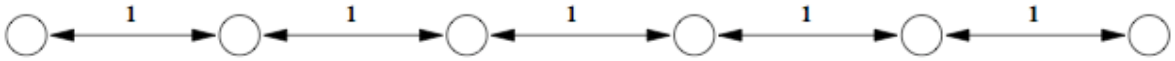
University of Waterloo

7. Numerical Results

7.1 Uniform 1D Chain

	MCAMG						A-SAM [8] distance-two					
n	γ	it	C_{op}	γ_{eff}	lev	R_l	γ	it	C_{op}	γ_{eff}	lev	R_l
2187	0.18	11	1.99	0.43	9	0	0.31	12	1.49	0.46	6	0
6561	0.18	11	2.00	0.43	11	0	0.31	12	1.49	0.46	7	0
19683	0.18	11	2.00	0.43	12	0	0.32	12	1.49	0.47	8	0
59049	0.18	11	2.00	0.43	14	0	0.32	12	1.50	0.47	9	0

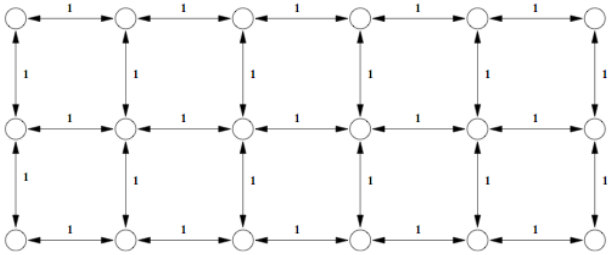
TABLE 5.1
Uniform chain.



7.2 Uniform 2D Lattice

	MCAMG						A-SAM [8] distance-two					
n	γ	it	C_{op}	γ_{eff}	lev	R_l	γ	it	C_{op}	γ_{eff}	lev	R_l
1024	0.23	11	2.20	0.51	6	0	0.49	20	1.42	0.60	4	4.5e-3
4096	0.23	11	2.20	0.52	7	0	0.49	20	1.47	0.62	4	1.7e-3
16384	0.24	11	2.20	0.52	8	0	0.59	20	1.56	0.72	5	1.4e-3
65536	0.24	11	2.20	0.52	9	0	0.66	21	1.59	0.77	6	1.3e-3

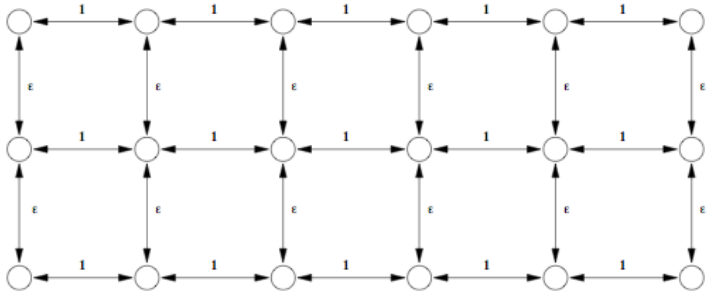
TABLE 5.2
Uniform 2D lattice.



7.3 Anisotropic 2D Lattice

	MCAMG						A-SAM [8] distance-two					
n	γ	it	C_{op}	γ_{eff}	lev	R_l	γ	it	C_{op}	γ_{eff}	lev	R_l
1024	0.18	11	2.58	0.52	8	0	0.33	14	2.81	0.68	5	1.6e-3
4096	0.18	11	2.67	0.53	10	0	0.33	14	3.43	0.73	7	4.9e-4
16384	0.18	11	2.73	0.54	12	0	0.33	13	4.17	0.77	7	2.5e-4
65536	0.18	11	2.76	0.54	14	0	0.32	13	4.80	0.79	9	7.6e-5

TABLE 5.3
Anisotropic 2D lattice ($\varepsilon = 1e-6$).



7.5 Tandem Queueing Network

	MCAMG						A-SAM [8] distance-two					
n	γ	it	C_{op}	γ_{eff}	lev	R_l	γ	it	C_{op}	γ_{eff}	lev	R_l
1024	0.33	16	4.41	0.78	7	1.4e-1	0.41	20	2.04	0.64	4	7.6e-2
4096	0.32	15	4.54	0.78	8	1.2e-1	0.45	24	2.12	0.69	5	5.5e-2
16384	0.33	16	4.59	0.78	10	1.6e-1	0.56	30	2.18	0.77	6	5.3e-2
65536	0.33	15	4.61	0.79	11	7.0e-2	0.71	37	2.37	0.86	6	1.3e-1

TABLE 5.5
Tandem queueing network.

FIG. 5.6. *Tandem queueing network.*

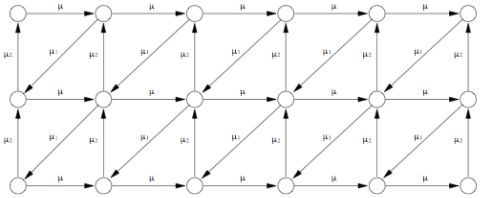
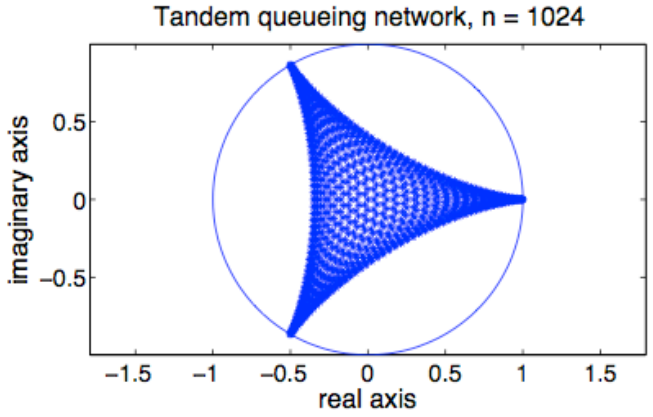


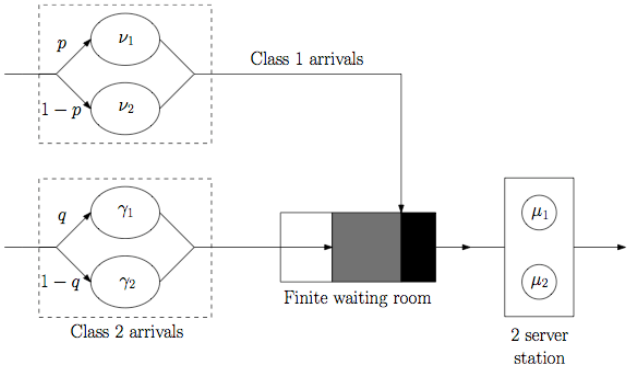
FIG. 5.7. *Graph for tandem queueing network.*

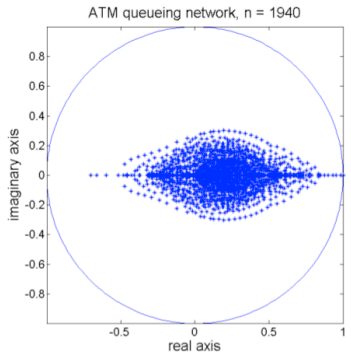


7.6 ATM Queueing Network (MCAMG)

n	γ	it	C_{op}	γ_{eff}	lev	R_l
1940	0.37	19	7.06	0.87	9	3.65e-2
3060	0.43	19	7.46	0.89	12	3.29e-2
5220	0.44	21	7.62	0.90	15	3.11e-2
10100	0.46	20	7.64	0.90	18	2.87e-2
13796	0.47	21	8.08	0.91	22	2.68e-2
19620	0.48	21	8.12	0.91	27	2.58e-2
32276	0.45	21	8.58	0.91	29	2.35e-2

TABLE 5.7
ATM queueing network.





University of Waterloo

hdesterck@uwaterloo.ca

8. Conclusions

- A-SAM (Smoothed Aggregation for Markov Chains) and MCAMG (Algebraic Multigrid for Markov Chains) are scalable: they are algorithms for calculating the stationary vector of slowly mixing Markov chains with near-optimal complexity
- smoothing is essential for aggregation for many problems
- appropriate theoretical framework (well-posedness)
- no theory yet on (optimal) convergence (non-symmetric matrices)
- this can be done in parallel
- other presentations in this mini-symposium: other multilevel methods for the stationary Markov problem

- Questions?

Algebraic Aggregation Mechanism

$$\bar{A} = A\,\mathrm{diag}(\mathbf{x}_i)$$ (scaled problem matrix)

$$S_{jk} = \begin{cases} 1 & \text{if } j \neq k \text{ and } -\bar{a}_{jk} \geq \theta \max_{l \neq j}(-\bar{a}_{jl}) \\ 0 & \text{otherwise,} \end{cases}$$ (strength matrix)

ALGORITHM. AGGREGATION BASED ON STRENGTH MATRIX S

repeat

- among the unassigned states, choose state j which has the largest value in current iterate $\mathbf{x}_i$ as the seed point of a new aggregate
- add all unassigned states k that are strongly influenced by seed point j (i.e., $S_{kj} = 1$) to the new aggregate

until all states are assigned to aggregates

Error Equation

- multiplicative error:
- error equation:

$$\mathbf{x} = \text{diag}(\mathbf{x}_i)\,\mathbf{e}_i$$
$$A\,\text{diag}(\mathbf{x}_i)\,\mathbf{e}_i = 0$$

- coarse grid equation:

$$Q^T A\,\text{diag}(\mathbf{x}_i)\,Q\,\mathbf{e}_c = 0$$
$$A_c\,\mathbf{e}_c = 0$$

- restriction and interpolation:

$$R = Q^T \qquad P = \text{diag}(\mathbf{x}_i)\,Q$$
$$A_c = R\,A\,P$$

- coarse grid correction: $\mathbf{x}_{i+1} = P\,\mathbf{e}_c$

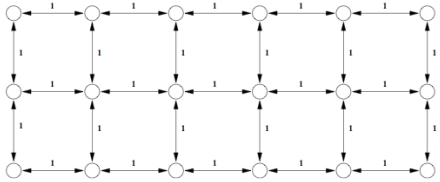
Error Equation

- important properties of A_c:

$$(1)\ \mathbf{1}_c^T A_c = 0 \quad \forall\, \mathbf{x}_i$$
$$(\text{since } \mathbf{1}_c^T R = \mathbf{1}^T \text{ and } \mathbf{1}^T A = 0)$$

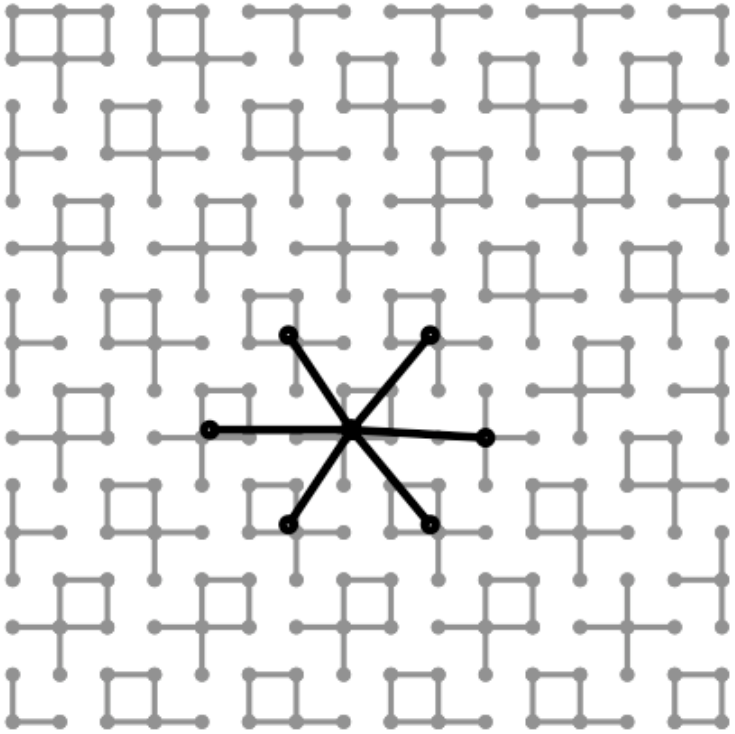
$$(2)\ A_c \mathbf{1}_c = 0 \quad \text{for } \mathbf{x}_i = \mathbf{x}$$

smoothed aggregation: periodic 2D lattice

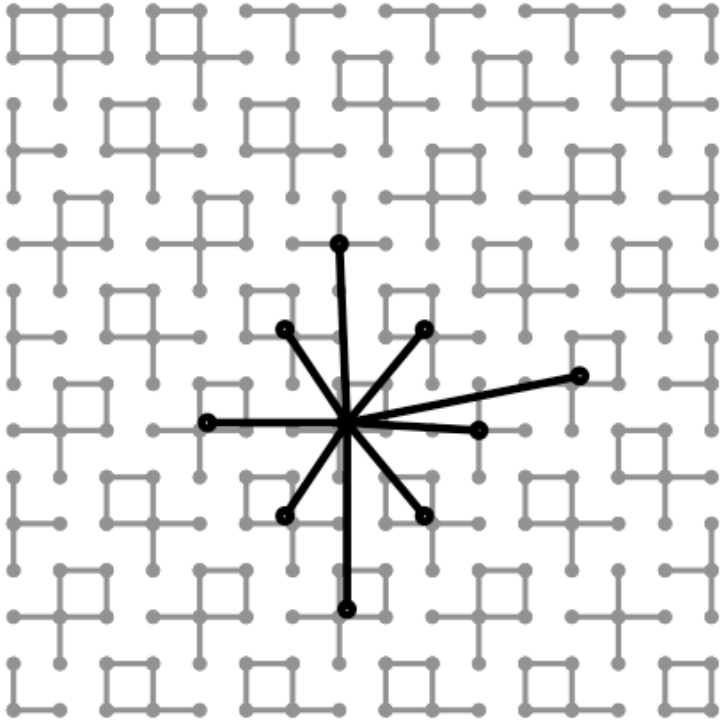


$$A_c = R_s \, A \, P_s$$

unsmoothed



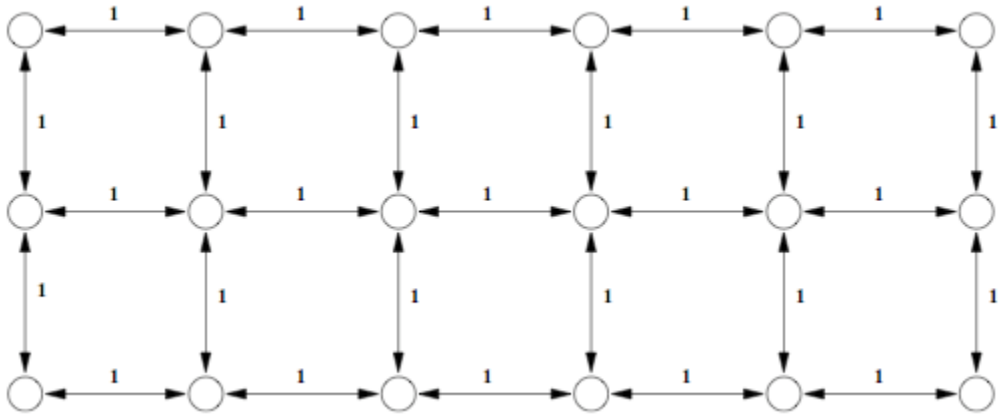
smoothed



University of Waterloo

RSA 2009
hdesterck@uwaterloo.ca

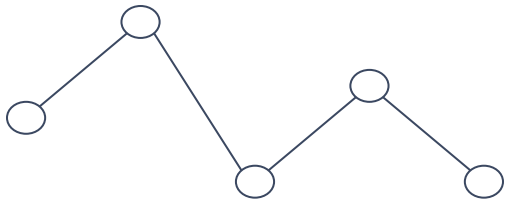
numerical results: smoothed aggregation multigrid for periodic 2D lattice problem



	1-level	aggregation			smoothed aggregation			
n	iterations	iterations	C_{op}	levels	iterations	C_{op}	levels	R_{lump}
64	197	47	1.23	3	16	1.26	3	0
256	760	96	1.26	3	17	1.34	3	0
1024	2411	242	1.25	4	17	1.32	4	0
4096		328	1.26	5	18	1.34	5	0
16384					18	1.33	5	0
32768					19	1.34	6	0

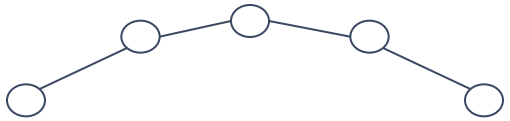
We Need ‘Smoothed Aggregation’...

(Vanek, Mandel, and Brezina, Computing, 1996)

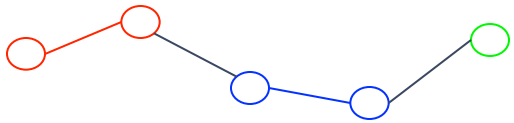


$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

after smoothing:

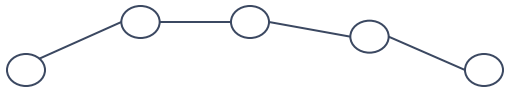


coarse grid correction with Q:



$$Q_s = \begin{bmatrix} \times & 0 & 0 \\ \times & \times & 0 \\ \times & \times & 0 \\ 0 & \times & \times \\ 0 & \times & \times \end{bmatrix}$$

coarse grid correction with Q_s:



Smoothed Aggregation

$$A = D - (L + U)$$

- smooth the columns of *P* with weighted Jacobi:

$$P_s = (1 - w)\,\mathrm{diag}(\mathbf{x}_i)\,Q + w\,D^{-1}\,(L + U)\,\mathrm{diag}(\mathbf{x}_i)\,Q$$

- smooth the rows of *R* with weighted Jacobi:

$$R_s = R\,(1 - w) + R\,w\,(L + U)\,D^{-1}$$

Smoothed Aggregation

- smoothed coarse level operator:

$$A_{cs} = R_s \left(D - (L+U)\right) P_s$$
$$= R_s D P_s - R_s (L+U) P_s$$

$$\mathbf{1}_c^T A_{cs} = 0 \quad \forall \mathbf{x}_i,$$
$$A_{cs} \mathbf{1}_c = 0 \quad \text{for } \mathbf{x}_i = \mathbf{x}$$

- problem: A_{cs} is not a singular M-matrix (signs wrong)

- solution: lumping approach on S in

$$A_{cs} = S - G \qquad \hat{A}_{cs} = \hat{S} - G$$

$$A = \begin{bmatrix} + & - & - & - & - \\ - & + & - & - & - \\ - & - & + & - & - \\ - & - & - & + & - \\ - & - & - & - & + \end{bmatrix}$$

Smoothed Aggregation

$$A_{cs} = S - G \qquad \hat{A}_{cs} = \hat{S} - G$$

$$A = \begin{bmatrix} + & - & - & - & - \\ - & + & - & - & - \\ - & - & + & - & - \\ - & - & - & + & - \\ - & - & - & - & + \end{bmatrix}$$

- we want as little lumping as possible
- only lump 'offending' elements *(i,j):*

$$s_{ij} \neq 0,\ i \neq j \text{ and } s_{ij} - g_{ij} \geq 0$$

$$\mathbf{1}_c^T \hat{A}_{cs} = 0 \quad \forall \mathbf{x}_i,$$
$$\hat{A}_{cs}\, \mathbf{1}_c = 0 \quad \text{for } \mathbf{x}_i = \mathbf{x}$$

(we consider both off-diagonal signs and reducibility here!)

- for 'offending' elements *(i,j)*, add $S_{\{i,j\}}$ to S:

$$S_{\{i,j\}} = \begin{array}{c} \\ i \\ \\ j \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccc} & i & & j & \end{array} \\ \begin{bmatrix} \ddots & \vdots & & \vdots & \\ \cdots & \beta_{\{i,j\}} & \cdots & -\beta_{\{i,j\}} & \cdots \\ & \vdots & & \vdots & \\ \cdots & -\beta_{\{i,j\}} & \cdots & \beta_{\{i,j\}} & \cdots \\ & \vdots & & \vdots & \end{bmatrix} \end{array}$$

$$s_{ij} - g_{ij} - \beta_{\{i,j\}} < 0$$
$$s_{ji} - g_{ji} - \beta_{\{i,j\}} < 0$$

conserves both row and column sums

Lumped Smoothed Method is Well-posed

(A-SAM: Algebraic Smoothed Aggregation for Markov Chains)

THEOREM 4.1 (Singular M-matrix property of lumped SAM coarse-level operators). *$\hat{A}_{cs}$ is an irreducible singular M-matrix on all coarse levels, and thus has a unique right kernel vector $\mathbf{e}_c$ with strictly positive components (up to scaling) on all levels.*

THEOREM 4.2 (Fixed-point property of lumped SAM V-cycle). *Exact solution $\mathbf{x}$ is a fixed point of the SAM V-cycle (with lumping).*

$$\mathbf{1}_c^T \hat{A}_{cs} = 0 \quad \forall \mathbf{x}_i,$$
$$\hat{A}_{cs} \mathbf{1}_c = 0 \quad \text{for } \mathbf{x}_i = \mathbf{x}$$

$$A = \begin{bmatrix} + & - & - & - & - \\ - & + & - & - & - \\ - & - & + & - & - \\ - & - & - & + & - \\ - & - & - & - & + \end{bmatrix}$$

(De Sterck et al., SISC (accepted, 2009), 'Smoothed aggregation multigrid for Markov chains')

AMG Properties

- we can show: all elements of *P* >= *0*
- lumping can be done as in the Smoothed Aggregation case:

$$A = \begin{bmatrix} + & - & - & - & - \\ - & + & - & - & - \\ - & - & + & - & - \\ - & - & - & + & - \\ - & - & - & - & + \end{bmatrix}$$

$$\bar{A}_c = P^T \bar{A}\, P = P^T D\, P - P^T (L+U)\, P = S - G$$

$$\hat{A}_c = \hat{S} - G$$

- lumping conserves row and column sums:

$$\mathbf{1}_c^T \hat{A}_c = \mathbf{1}_c^T \bar{A}_c + \mathbf{1}_c^T (\hat{S} - S) = \mathbf{1}_c^T \bar{A}_c = \mathbf{0} \quad \forall\, \mathbf{x}_i,$$

$$\hat{A}_c\, \mathbf{1}_c = \bar{A}_c\, \mathbf{1}_c + (\hat{S} - S)\, \mathbf{1}_c = \bar{A}_c\, \mathbf{1}_c = \mathbf{0} \quad \text{for } \mathbf{x}_i = \mathbf{x}$$

Algebraic Multigrid for Markov Chains (MCAMG)

Algorithm 1: MCAMG$(A, \mathbf{x}, \nu_1, \nu_2)$, AMG for Markov chains (V-cycle)

if *not at the coarsest level* **then**
  $\mathbf{x} \leftarrow \text{Relax}(A, \mathbf{x})$ ν_1 times
  $\bar{A} \leftarrow A\,\text{diag}(\mathbf{x})$
  Compute the set of coarse-level points C
  Construct the interpolation operator P
  Construct the coarse-level operator $\bar{A}_c \leftarrow P^T \bar{A}\, P$
  Obtain the lumped coarse-level operator $\hat{A}_c \leftarrow \text{Lump}(\bar{A}_c, \eta)$
  $\mathbf{e}_c \leftarrow \text{MCAMG}(\hat{A}_c, \mathbf{1}_c, \nu_1, \nu_2)$ `/* coarse-level solve */`
  $\mathbf{x} \leftarrow \text{diag}(\mathbf{x})\, P\, \mathbf{e}_c$ `/* coarse-level correction */`
  $\mathbf{x} \leftarrow \text{Relax}(A, \mathbf{x})$ ν_2 times
else
  $\mathbf{x} \leftarrow$ direct solve of $K\,\mathbf{x} = \mathbf{z}$ `/* see Section 4.4 */`
end

MCAMG Properties

THEOREM 4.2 (Singular M-matrix property of lumped coarse-level operator). *$\hat{A}_c$ is an irreducible singular M-matrix on all coarse levels and, thus, has a unique right-kernel vector with positive components (up to scaling) on all levels.*

$$A = \begin{bmatrix} + & - & - & - & - \\ - & + & - & - & - \\ - & - & + & - & - \\ - & - & - & + & - \\ - & - & - & - & + \end{bmatrix}$$

THEOREM 4.3 (Fixed-point property of MCAMG V-cycle). *The exact solution, $\mathbf{x}$, is a fixed point of the MCAMG V-cycle.*

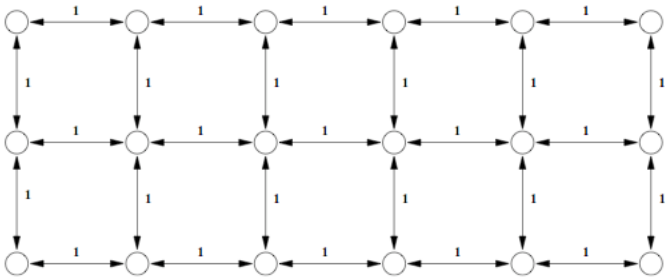
(De Sterck et al., 'Algebraic Multigrid for Markov Chains', preprint)

8. numerical results

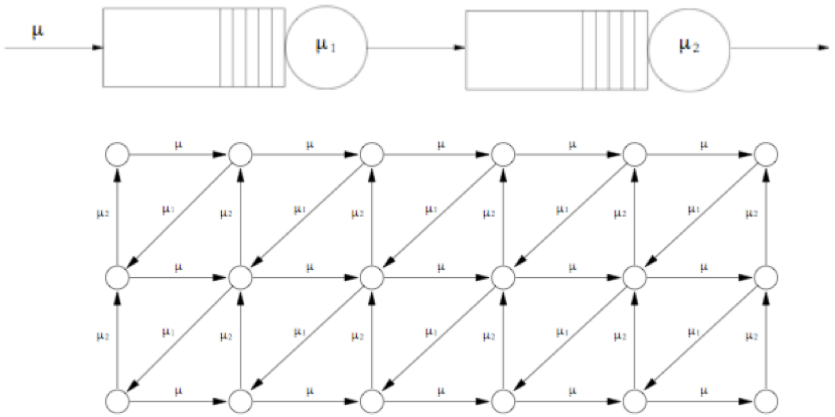
1) random walk on 2D lattice

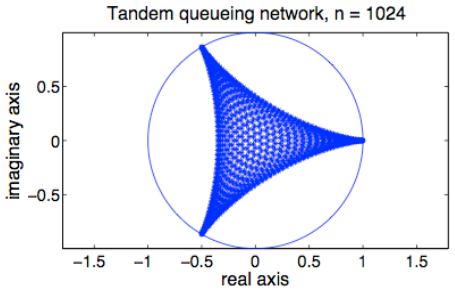
	W cycles		RAMA cycles		W+ cycles		RAMA+ cycles	
n	it	C_{op}	it	C_{op}	it	C_{op}	it	C_{op}
64	39	1.47	35	1.47	18	1.47	18	1.47
256	62	1.51	40	1.51	26	1.51	20	1.51
1024	106	1.57	41	1.57	36	1.57	22	1.57
4096	104	1.60	41	1.61	36	1.60	21	1.61
16384	166	1.60	42	1.60	47	1.60	21	1.60
65536	187	1.60	68	1.60	50	1.60	28	1.61

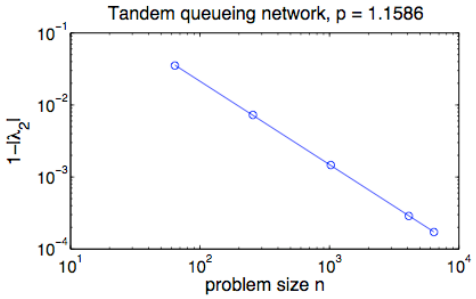
note: ‘+’ means additional top-level acceleration with window size 3



2) tandem queue





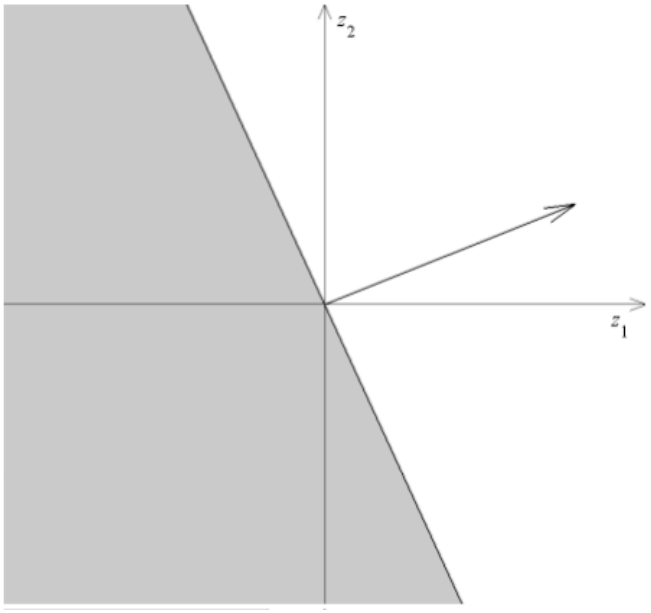


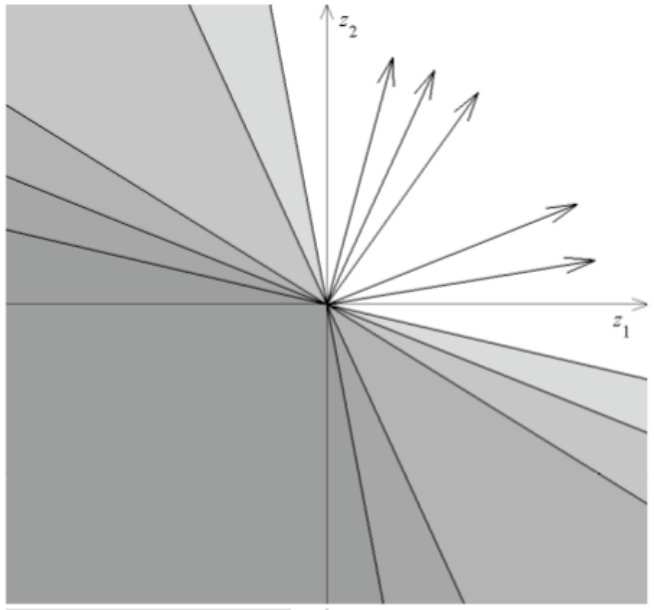
	W cycles		RAMA cycles		W+ cycles		RAMA+ cycles	
n	it	C_{op}	it	C_{op}	it	C_{op}	it	C_{op}
256	67	1.47	42	1.47	33	1.47	27	1.47
1024	121	1.47	48	1.47	53	1.47	31	1.47
4096	142	1.50	50	1.50	50	1.50	33	1.50
16384	212	1.51	57	1.51	63	1.51	32	1.51
65536	229	1.50	63	1.50	78	1.50	37	1.50

quadratic programming problem

$$\begin{aligned} & \mathbf{z}^* = \text{argmin}_{\mathbf{z}} \ \|(A\hat{X})\mathbf{z}\|_2 \\ \text{subject to:} \quad & \hat{X}\,\mathbf{z} \geq 0 \quad \text{and} \\ & \mathbf{1}^T\mathbf{z} = 1. \end{aligned}$$

efficient explicit solution for recombination of two iterates





University of Waterloo

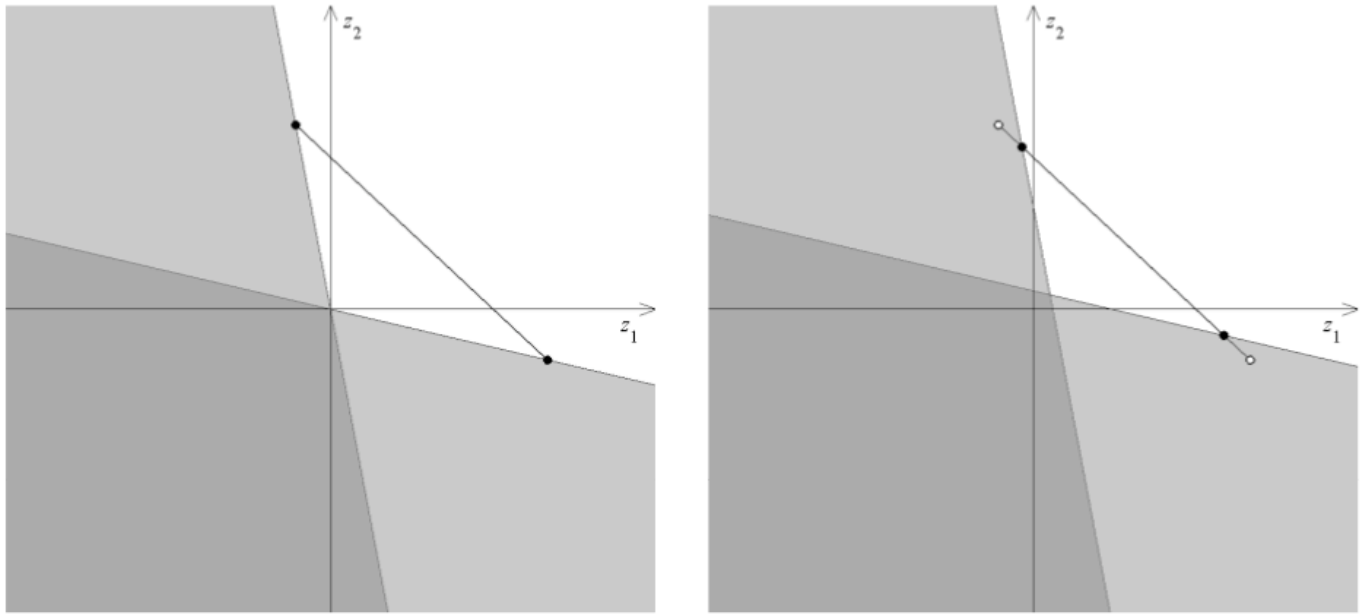
Copper 2010

quadratic programming problem

$$\begin{aligned} & \mathbf{z}^* = \operatorname{argmin}_{\mathbf{z}} \ \|(A\,\hat{X})\mathbf{z}\|_2 \\ \text{subject to:} \quad & \hat{X}\,\mathbf{z} \geq 0 \quad \text{and} \\ & \mathbf{1}^T\,\mathbf{z} = 1. \end{aligned}$$

$$\begin{aligned} & \mathbf{z}^* = \operatorname{argmin}_{\mathbf{z}} \ \|(A\,\hat{X})\mathbf{z}\|_2 \\ \text{subject to:} \quad & \hat{X}\,\mathbf{z} \geq \delta \, \min_{i,j}(\hat{x}_{i,j}) \quad \text{and} \\ & \mathbf{1}^T\,\mathbf{z} = 1, \end{aligned}$$

efficient explicit solution for recombination of two iterates



$$z_1^* = \frac{\langle A\mathbf{x}_2, A\mathbf{x}_2\rangle - \langle A\mathbf{x}_1, A\mathbf{x}_2\rangle}{\langle A\mathbf{x}_1, A\mathbf{x}_1\rangle - 2\,\langle A\mathbf{x}_1, A\mathbf{x}_2\rangle + \langle A\mathbf{x}_2, A\mathbf{x}_2\rangle}$$

Copper 2010

University of Waterloo