Multiscale Methods

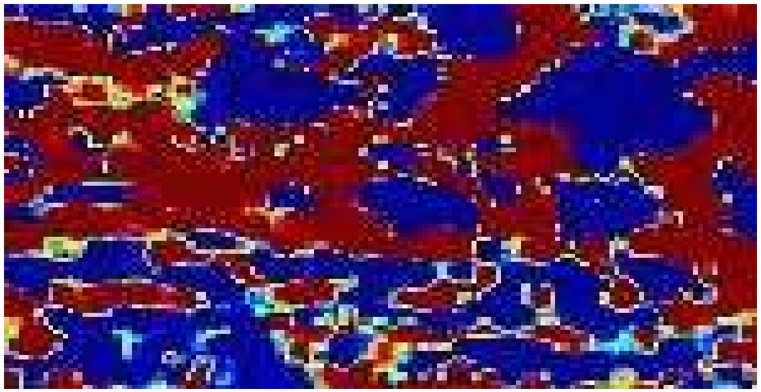
In many complex systems a big scale gap can be observed between micro- and macroscopic scales because of the difference in physical (social, biological, mathematical, etc.) models and/or laws at different scales.

"Visions of Quixote," oil on canvas, 1989

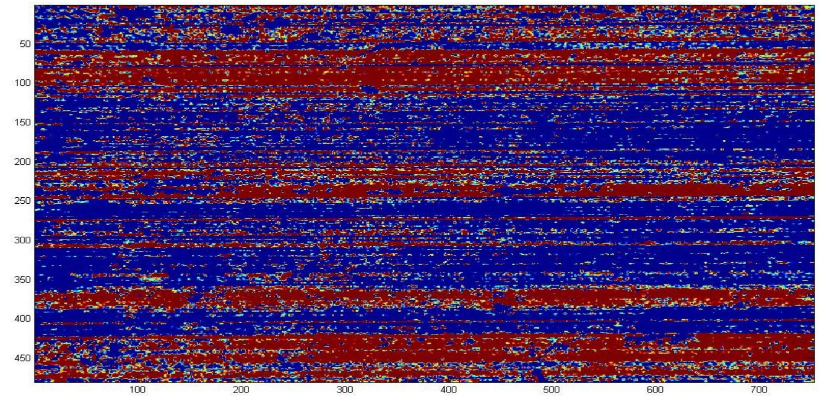
"Visions of Quixote," oil on canvas, 1989

Even if elementary objects of the system have a complicated (and even nondeterministic) behavior, their ensembles can be more structured .

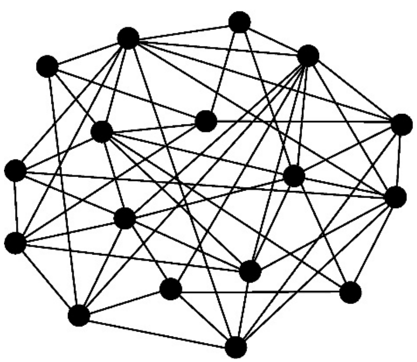
High resolution

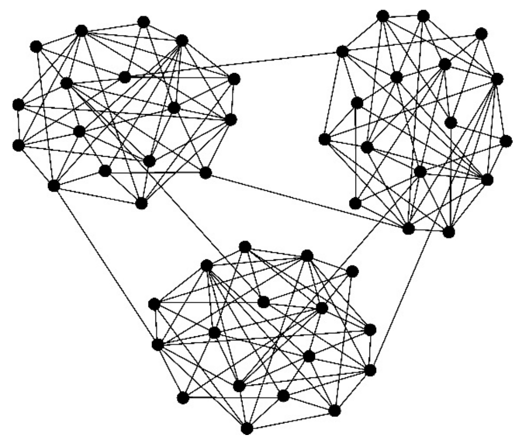


Low resolution



Two body scratch model





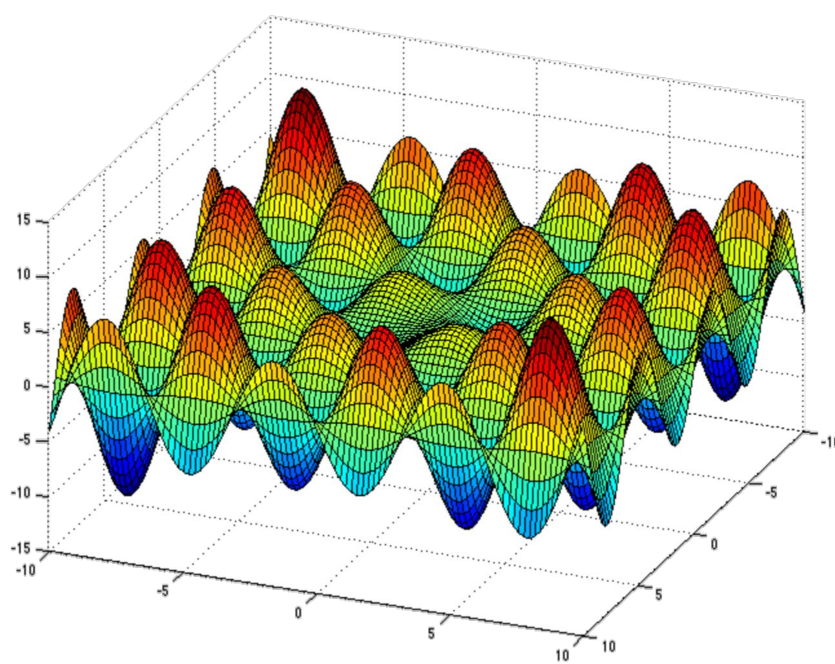
Social networks

Even when the difference between models at different scales is not observed, an efficient approximation of the microscopic scale can be achieved by looking at the macroscopic scale **with its substantially smaller number of elementary objects**.

Motivation

There exist tens (if not hundreds) of different classes of algorithms for large-scale combinatorial optimization that eventually ensure the solution or a small gap.

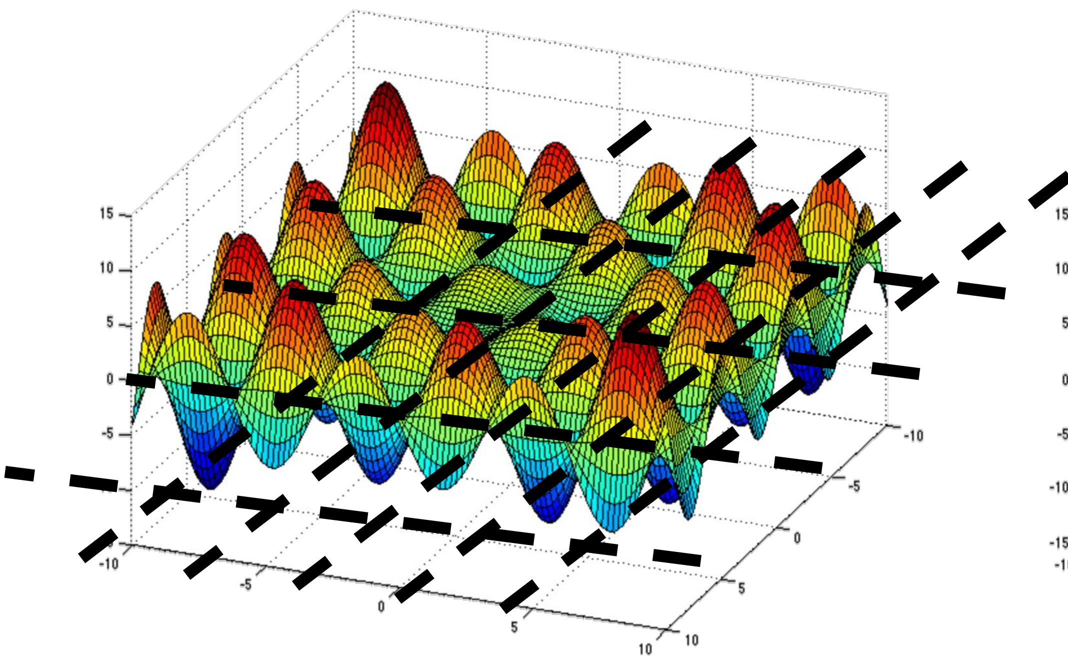
However, despite of their effectiveness and availability of computational resources, there are always certain barriers in the problem size and algorithm complexity.



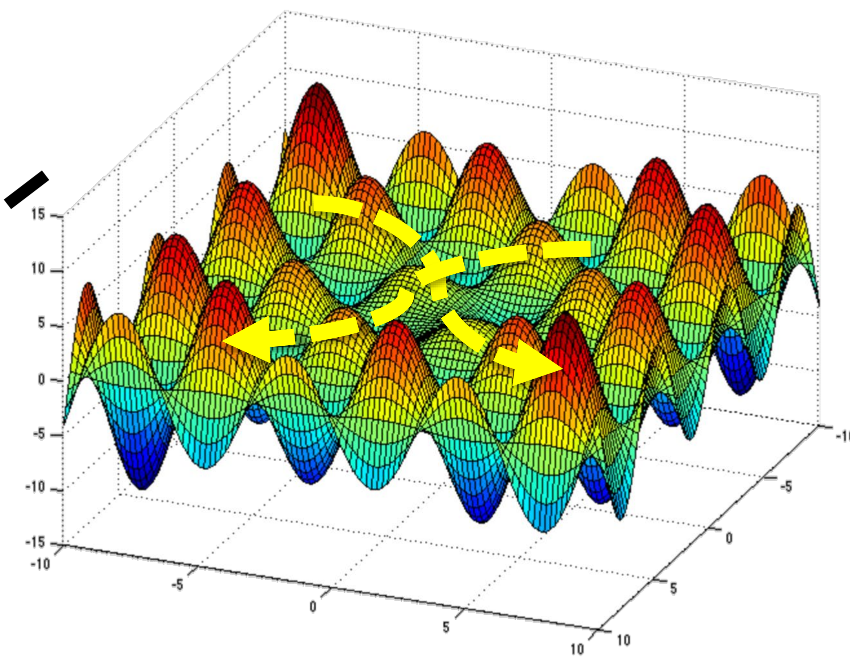
Motivation

When these barriers are met, typical ways to continue tackling the problem are by using

Decomposition



Smart but "blind" search



However, there is another barrier that these methods typically do not overcome: *we solve "one variable at a time"*.

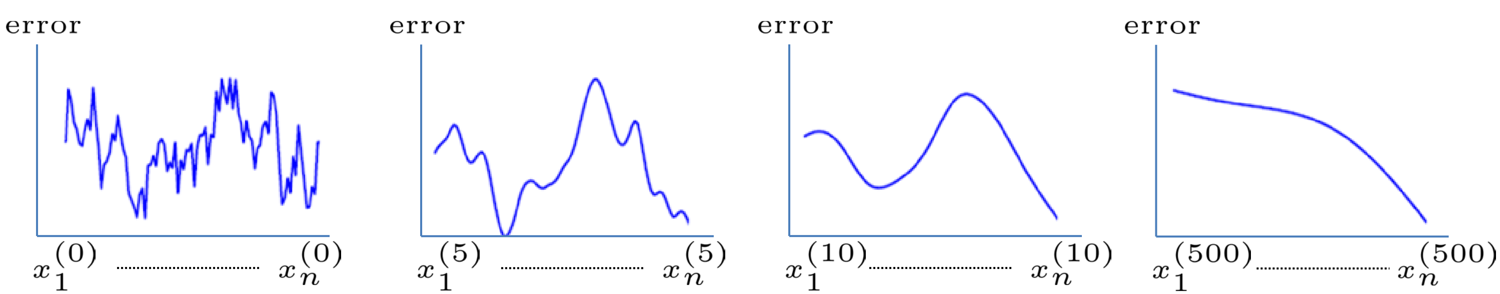
The Multiscale Method
Multiscale ≈ Multilevel ≈ Multigrid ≈ Multiresolutional

- The **Multiscale method is a class of algorithmic techniques for solving efficiently large-scale computational and optimization problems**.
- **A multivariable problem defined in some space can have an approximate description at any given length scale of that space** (a continuum problem can be discretized at any given resolution, multiparticle system can be represented at any given characteristic length, etc).
- **The multiscale algorithm recursively constructs a sequence of such descriptions at increasingly larger (coarser) scales**, and combines *local* processing at each scale with *inter-scale* interactions.

Algebraic multigrid in three slides

- We need to solve a large-scale system $Ax = b$, SPD
- ~~Gaussian elimination~~, ~~LU~~, ~~Cholesky~~, ~~QR~~, ... 

- Iterative methods $x^{(k+1)} = Tx^{(k)} + v$, e.g., Gauss-Seidel stationary iterative relaxation



Aha! A suitable relaxation can reduce the information content of the error, and quickly make it approximable by far fewer variables.

Algebraic multigrid in three slides

Brandt, McCormick, Rudge, "Algebraic Multigrid (AMG) for automatic multi-grid solution with application to geodetic computations", 1982

- Given: $A \in \mathbb{R}^{n \times n}$ positive definite, symmetric.
- Goal: solve $Ax = b$.
- Claim: If A is positive definite, then

$$x \text{ minimizes } P(x) = \frac{1}{2}x^T Ax - x^T b \textbf{ iff } Ax = b.$$

- $\tilde{x}$ - current approximation
- $e(rror) = x - \tilde{x}$ (hard to estimate)
- $b - A\tilde{x} = r(\text{esidual}) = A(x - \tilde{x}) = Ae$

Algebraic multigrid in three slides

At all levels: solve $Ae = r$, where $e(rror) = x - \tilde{x}$ and $r(esidual) = b - A\tilde{x}$

$$min\ \frac{1}{2}e^T Ae - e^T r \ =$$

$$min\ \frac{1}{2}(\tilde{e}+\uparrow_c^f e^c)^T A(\tilde{e}+\uparrow_c^f e^c) - (\tilde{e}+\uparrow_c^f e^c)^T r \ \leftrightarrow\ \ldots\ \leftrightarrow$$

$$min\ \frac{1}{2}(e^c)^T \left[(\uparrow_c^f)^T A \uparrow_c^f\right] e^c - (e^c)^T(\uparrow_c^f)^T(r - A\tilde{e}) \ =$$

$$min\ \frac{1}{2}(e^c)^T A^c e^c - (e^c)^T r^c$$

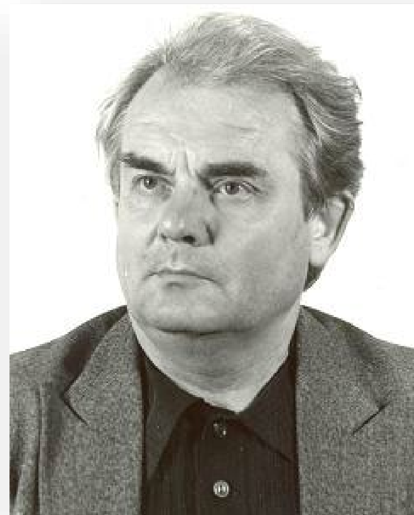
- $\tilde{e}$ - initial fine level error
- e^c - coarse level error
- $\uparrow_c^f$ - coarse-to-fine interpolation operator

History of Multiscale Methods

Joseph Fourier

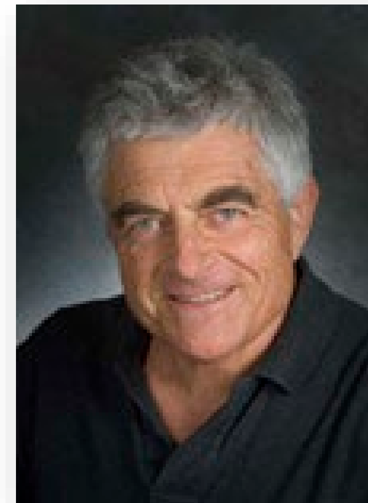
Functional analysis at multiple resolutions (1768-1830)

Radiy Fedorenko



Smoothing, finite elements, two-level multigrid (1930-2009)

Achi Brandt



Popularization, first basic research (1977), algebraic multigrid (1980), ...

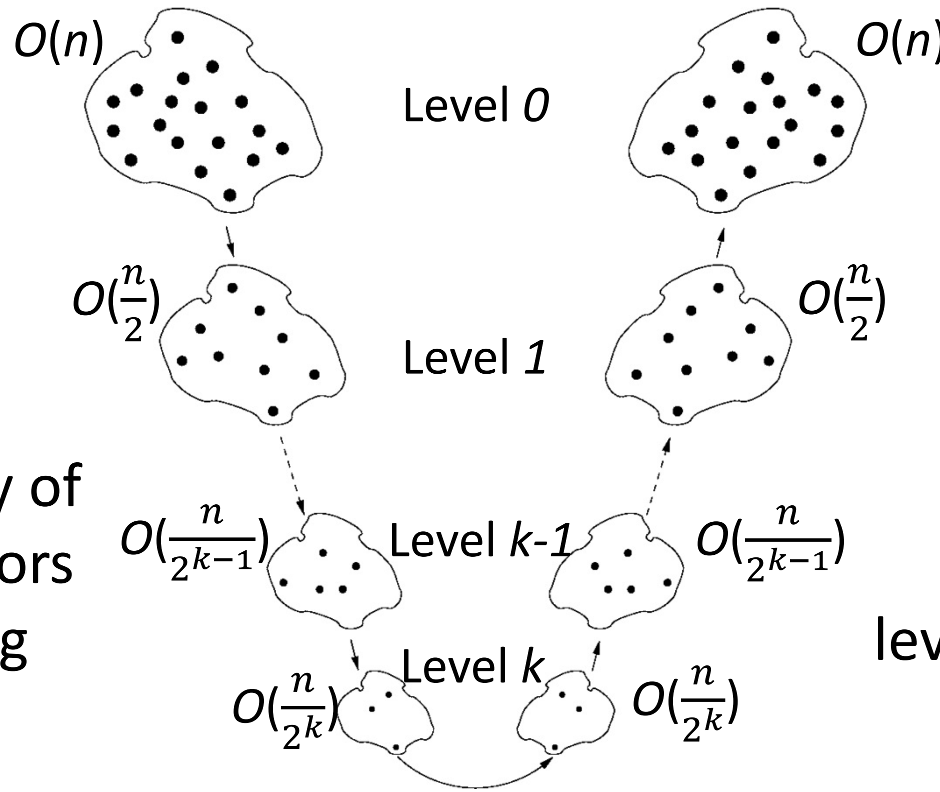
Examples of multilevel and multiscale classes of algorithms

- Line search multigrid for convex optimization (Goldfarb, Wen)
- PDE-constrained optimization (Borzi, Nash, Toint, ...)
- Multilevel trust-region methods (Gratton, Mouffe, Sartenaer, Toint, ...)
- Non-convex non-linear optimization for VLSI placement (Chan, Cong, Sze, ...)
- Linear programming - multilevel iterative methods (Gelman, Mandel, ...)
- Derivative-free multilevel optimization (Mendonca, Peckman, Toint, ...)

Examples of multilevel combinatorial optimization

- (Hyper)graph partitioning and clustering (see many references in "Recent advances in graph partitioning", 2016)
- Various graph/matrix arrangement problems such as the minimum linear arrangement, bandwidth, workbound, wavefront, fill-in (Brandt, Hu, Ron, Safro, …)
- Vertex separators (Karypis, Hager, Safro, Sanders, Schultz, …)
- Coloring (Walshaw)
- TSP (Walshaw, Ron, …)
- VLSI placement (Chan, Cong, Hu, Karypis, Brandt, Ron, Viswanathan, …)

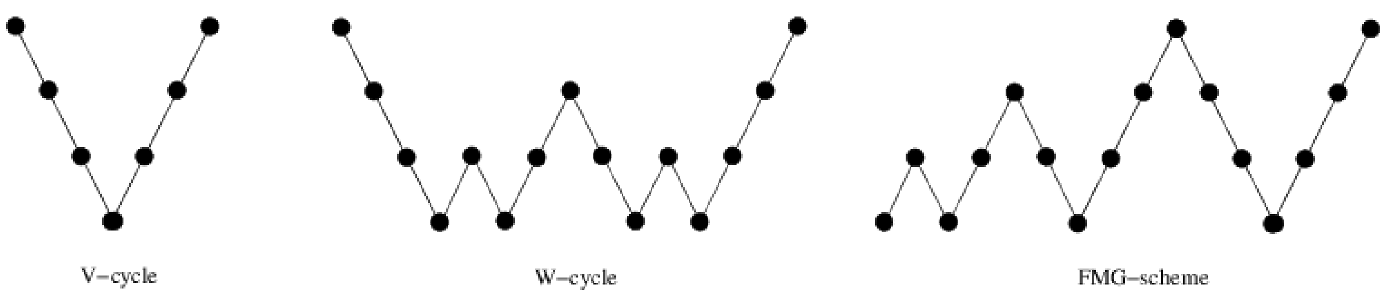
Cycles and complexity $\sum_{i=0}^{k} O(\frac{n}{2^i}) \rightarrow O(n)$

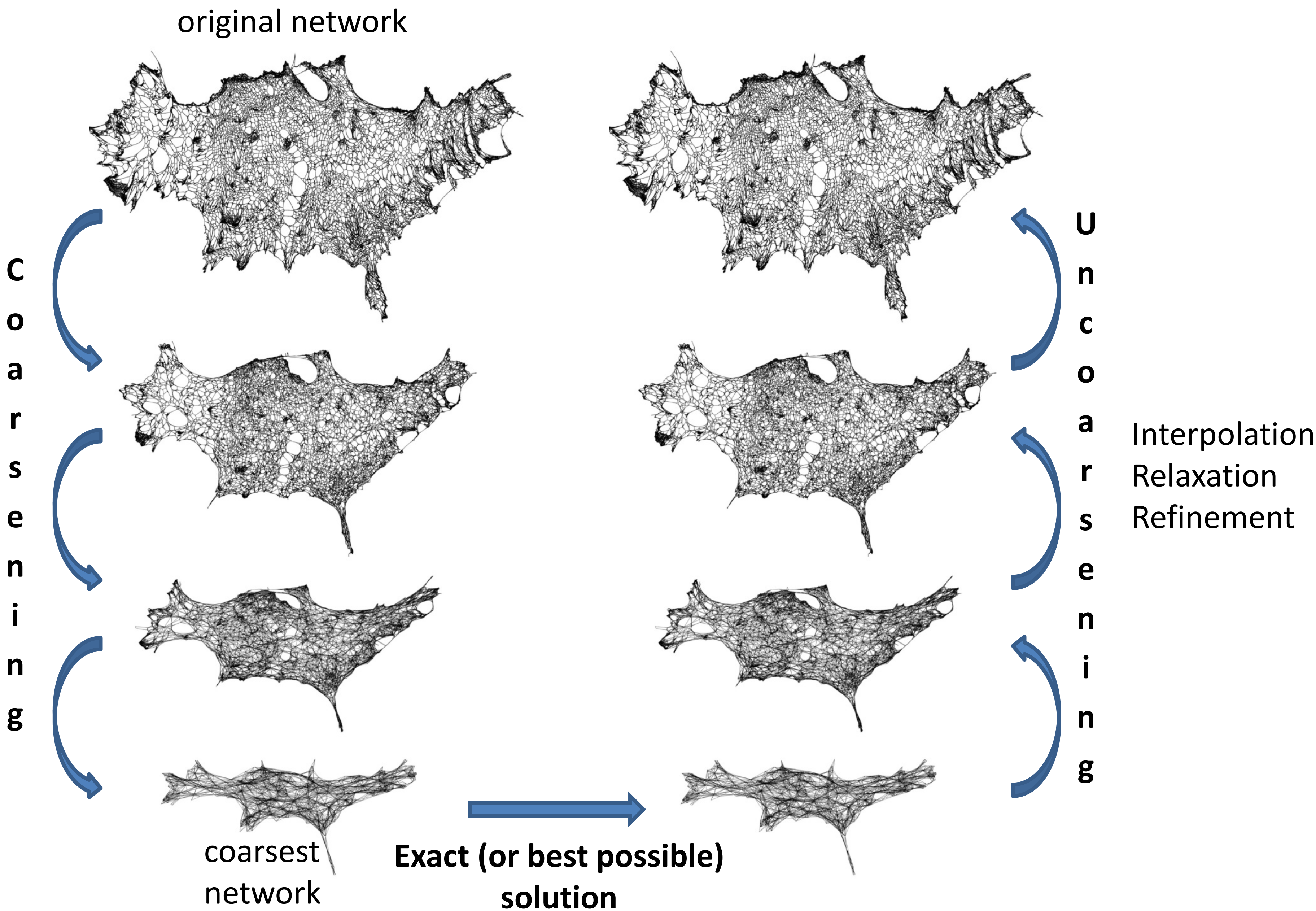


Exact solution

Coarsening
Create a hierarchy of restriction operators and corresponding coarse problems

Uncoarsening
Approximate solutions at each level by interpolation from coarser level, and further refinement




original network
C
o
a
r
s
e
n
i
n
g
coarsest
network
Exact (or best possible)
solution
U
n
c
o
a
r
s
e
n
i
n
g
Interpolation
Relaxation
Refinement

Multilevel Algorithms for Optimization Problems on Networks

- Examples: VLSI Placement, Partitioning, Minimum Linear Arrangement, Minimum Bandwidth, Clustering, TSP, Community Detection, Segmentation, Visualization, ...

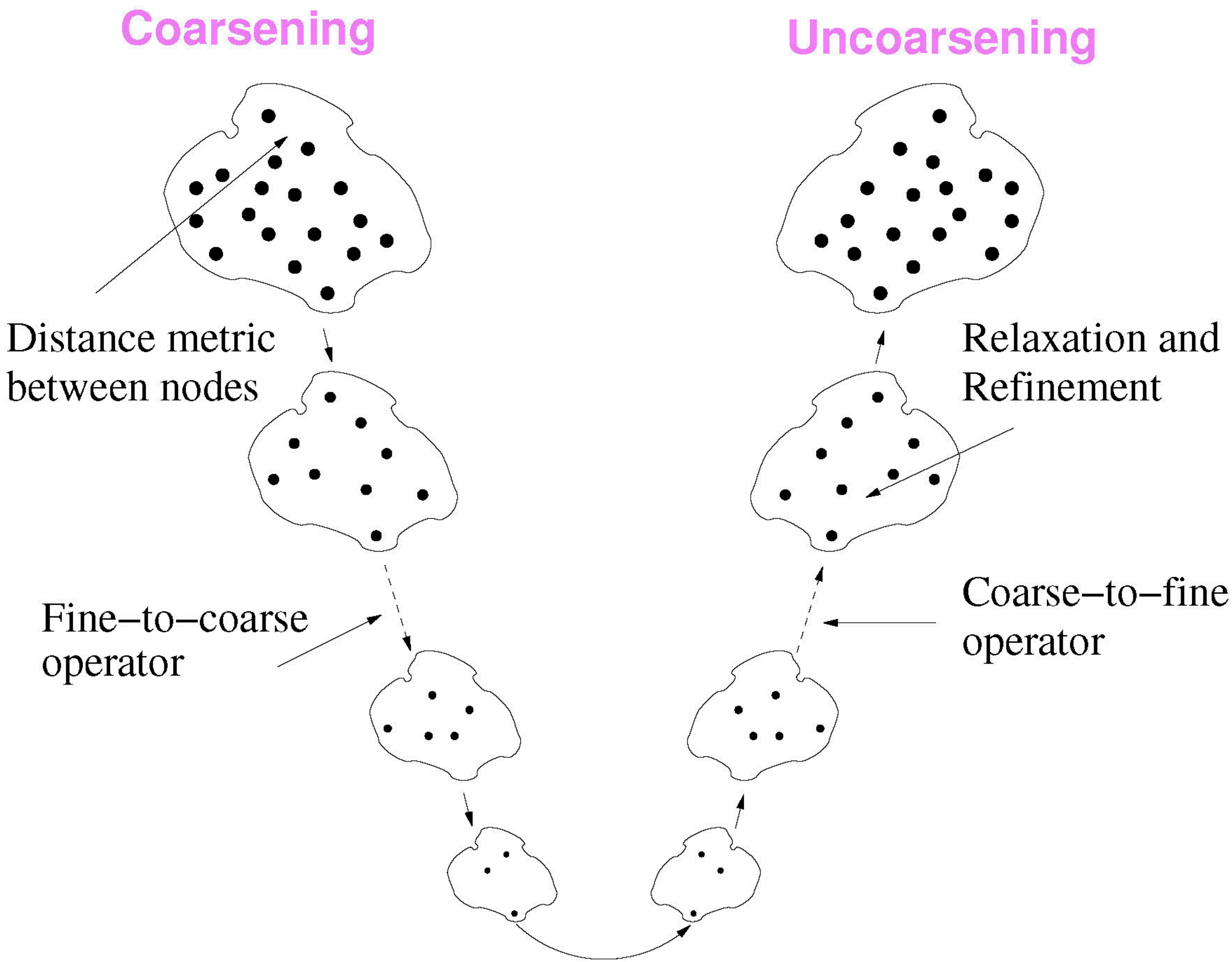
- Quality: Usually exhibit superior results to other methods on practical test suites.

- Time: Usually exhibit **linear** time complexity.

- Technical advantage: Admits parallelization. Suitable for various HPC configurations.

Four main questions

Think globally, act locally



Exact or best possible solution

Network Compression-friendly Ordering
(and Minimum Linear Arrangement Problems)

Compressed row representation

Node	Sorted list of neighbors (possibly with edge info)
1	2, 5, 6, 12, 18, 23, 103
…	…
1584	1585, 1592, 1600

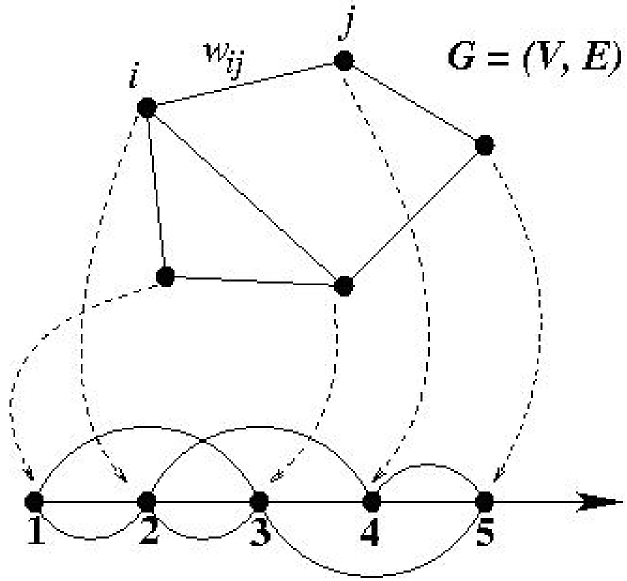
[KDD09 Chierichetti et al.] Given a sorted list of neighbours (x_1, x_2, x_3, ...), represent it by a list of differences (x_1, $x_2 - x_1$, $x_3 - x_1$, ...) or (x_1, $x_2 - x_1$, $x_3 - x_2$, ...)

Compressed row gap representation

Node	Sorted list of neighbors (possibly with edge info)
1	1, 4, 5, 11, 17, 22, 102
…	…
1584	1, 8, 16

… and then apply some compression algorithm (such as Boldi-Vigna scheme)

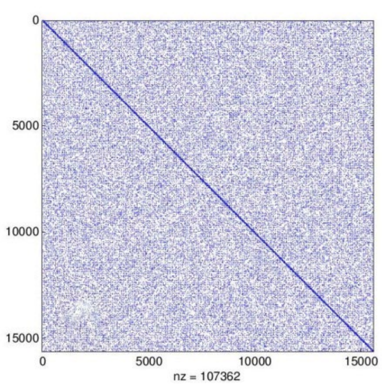
Network Compression-friendly Ordering

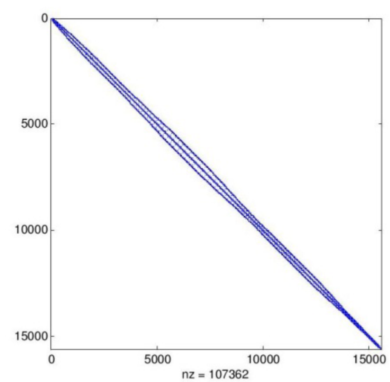


- Graph $G = (V, E)$
- Weighting function on edges $w : E \to \mathbb{R}_{\geq 0}$
- Permutation of vertices $\pi : V \to \{1, 2, ..., |V|\}$

The Minimum Logarithmic Arrangement Problem

$$\min_{\pi \in s(n)} \sum_{ij \in E} w_{ij} \lg |\pi(i) - \pi(j)|$$



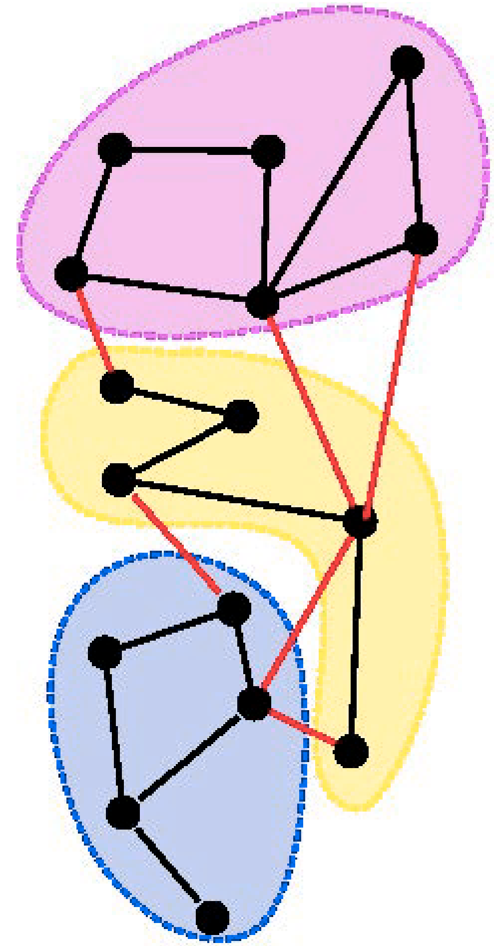


Network compression-friendly ordering, minimum linear arrangement, minimum 2-sum, minimum bandwidth, etc. are well known NP-complete problems.

Graph Minimum Partitioning/Clustering Problem

Given

- Graph $G = (V, E)$
- Weighting function on edges $w : E \to \mathbb{R}_{\geq 0}$
- Partitioning of vertices $\pi : V \to \{P_1, P_2, ..., P_k\}$
- Imbalance factor $0 \leq \alpha \leq 1$



The Minimum k-partitioning Problem

$$\min_{\pi \text{ is partitioning}} \sum_{u \in P_j, v \notin P_j} w_{uv}$$

$$\text{such that } |P_i| \leq (1 + \alpha) \cdot \frac{|V|}{k}$$

Applications: network analysis, machine learning, load-balancing, HPC, etc.

Results with up to 5% imbalance
S_max <= 1.05 x S_opt

graph	2	4	8	16	32	64
add20	536 (1256) [IPMNE2]	1120 (628) [*+IPMNE2]	1657 (314) [HYPAL]	2027 (157) [FSMAGP]	2341 (78) [FSMAGP]	2920 (39) [FSMAGP]
data	181 (1497) [SDP]	363 (745) [KaFFPa]	628 (374) [NW]	1076 (187) [FSMAGP]	1743 (94) [FSMAGP]	2747 (47) [KaBaPET]
3elt	87 (2398) [JE]	197 (1237) [NW]	329 (619) [KaFFPaE]	557 (309) [KaFFPaE]	930 (155) [FSMAGP]	1498 (77) [KaBaPET]
uk	18 (2455) [JE]	39 (1238) [*+KFFP]	75 (633) [KaFFPaE]	137 (315) [*+IPMNE2]	236 (158) [KaBaPE]	394 (79) [KaBaPE]
add32	10 (2481) [J2.2]	33 (1241) [JE]	63 (650) [KasPar]	117 (311) [JE]	212 (156) [JE]	476 (80) [*+IPMNE2]
bcsstk33	9914 (4554) [iJ]	20158 (2294) [FSMAGP]	33908 (1147) [FSMAGP]	54119 (574) [FSMAGP]	76070 (287) [*+ILP]	105297 (143) [*+ILP]
whitaker3	126 (4908) [JE]	376 (2546) [FSMAGP]	644 (1283) [*+IPMNE2]	1068 (643) [KaBaPET]	1632 (322) [KaBaPET]	2425 (161) [*+ILP]
crack	182 (5187) [NW]	360 (2606) [NW]	666 (1342) [FSMAGP]	1063 (671) [FSMAGP]	1655 (329) [FSMAGP]	2487 (164) [*+ILP]
wing_nodal	1668 (5742) [SDP]	3520 (2869) [FSMAGP]	5339 (1436) [FSMAGP]	8160 (718) [FSMAGP]	11533 (359) [*+ILP]	15514 (179) [*+ILP]
fe_4elt2	130 (5572) [MRSB]	335 (2918) [FSMAGP]	578 (1462) [KaFFPaE]	979 (731) [FSMAGP]	1571 (366) [*+IPMNE2]	2406 (183) [*+ILP]
vibrobox	10310 (6184) [JE]	18690 (3235) [FSMAGP]	23924 (1618) [KaFFPaE]	31216 (809) [*+ILP]	38823 (405) [*+ILP]	45987 (202) [*+ILP]
bcsstk29	2818 (7008) [GrPart]	7925 (3672) [KaFFPaE]	13540 (1830) [KaFFPaE]	20924 (918) [NW]	33450 (459) [FSMAGP]	53703 (229) [FSMAGP]
4elt	137 (8003) [NW]	315 (4090) [NW]	515 (2047) [FSMAGP]	887 (1024) [KaBaPE]	1493 (512) [KaBaPET]	2478 (256) [*+ILP]
fe_sphere	384 (8289) [JE]	762 (4257) [*+KFFP]	1152 (2060) [JE]	1678 (1076) [FSMAGP]	2427 (536) [FSMAGP]	3456 (269) [FSMAGP]
cti	318 (8480) [JE]	889 (4416) [FSMAGP]	1684 (2200) [*+KFFP]	2701 (1101) [KaBaPET]	3904 (553) [FSMAGP]	5460 (277) [*+ILP]
memplus	5253 (9322) [*+ILP]	9281 (4661) [*+ILP]	11543 (2330) [*+KFFP]	12799 (1165) [*+IPMNE2]	13857 (582) [*+ILP]	15875 (291) [*+ILP]
cs4	353 (11811) [KaFFPa]	908 (5906) [KaBaPE]	1420 (2946) [*+IPMNE2]	2042 (1477) [*+ILP]	2855 (739) [*+ILP]	3959 (369) [*+ILP]
bcsstk30	6251 (14679) [JE]	16165 (7590) [FSMAGP]	34068 (3796) [FSMAGP]	68323 (1898) [FSMAGP]	109368 (949) [FSMAGP]	166787 (474) [*+ILP]
bcsstk31	2660 (18683) [*+ILP]	7065 (9341) [FSMAGP]	12823 (4669) [*+ILP]	22718 (2336) [*+ILP]	36354 (1168) [*+ILP]	55250 (584) [*+ILP]
fe_pwt	340 (18260) [GrPart]	700 (9370) [KaFFPaE]	1405 (4744) [FSMAGP]	2737 (2396) [FSMAGP]	5305 (1199) [*+ILP]	7956 (599) [*+ILP]
bcsstk32	4622 (23319) [KasPar]	8441 (11706) [KaFFPa]	18955 (5855) [*+IPMNE2]	34374 (2928) [KaBaPE]	58352 (1464) [*+IPMNE2]	88595 (732) [*+ILP]
fe_body	262 (22544) [MQI]	588 (11835) [*+KFFP]	1012 (5916) [*+IPMNE2]	1683 (2958) [KaBaPE]	2677 (1479) [*+ILP]	4500 (740) [*+ILP]
t60k	65 (31437) [SDP]	195 (15719) [*+KFFP]	441 (7874) [*+IPMNE2]	787 (3938) [KaBaPE]	1289 (1969) [*+ILP]	2013 (984) [*+ILP]
wing	770 (32511) [*+KFFP]	1589 (16270) [*+ILP]	2440 (8114) [*+IPMNE2]	3775 (4068) [*+IPMNE2]	5512 (2035) [*+ILP]	7529 (1018) [*+ILP]
brack2	660 (32600) [SDP]	2731 (16438) [KaFFPa]	6592 (8219) [KaFFPaE]	11052 (4110) [*+ILP]	16765 (2055) [KaBaPET]	25100 (1027) [*+ILP]
finan512	162 (37376) [Ch2.0]	324 (18688) [Ch2.0]	648 (9344) [Ch2.0]	1296 (4672) [Ch2.0]	2592 (2336) [Ch2.0]	10560 (1168) [NW]
fe_tooth	3773 (40567) [SDP]	6687 (20508) [*+IPMNE2]	11147 (10255) [*+ILP]	16983 (5128) [*+ILP]	24270 (2564) [*+ILP]	33387 (1282) [*+ILP]
fe_rotor	1940 (52284) [KaFFPa]	6779 (26150) [KaBaPET]	12308 (13074) [*+ILP]	19677 (6538) [*+ILP]	30355 (3269) [*+ILP]	44368 (1634) [*+ILP]
598a	2336 (57855) [MQI]	7722 (29130) [*+ILP]	15413 (14565) [*+IPMNE2]	25198 (7282) [*+IPMNE2]	37632 (3641) [*+ILP]	54677 (1820) [*+ILP]
fe_ocean	311 (73322) [GrPart]	1686 (37274) [KaFFPa]	3886 (18811) [KaBaPE]	7338 (9413) [FSMAGP]	12033 (4707) [*+ILP]	19391 (2353) [*+ILP]
144	6345 (75941) [FSMAGP]	14978 (37971) [*+ILP]	24174 (18986) [*+ILP]	36607 (9493) [*+ILP]	54160 (4747) [*+ILP]	75753 (2374) [*+ILP]
wave	8524 (82064) [KaFFPaE]	16528 (41006) [*+ILP]	28489 (20183) [*+IPMNE2]	42024 (10258) [*+ILP]	59608 (5129) [*+ILP]	81989 (2565) [*+ILP]
m14b	3802 (112532) [MQI]	12858 (56374) [*+ILP]	25126 (28182) [*+IPMNE2]	41097 (14094) [*+ILP]	63397 (7047) [*+ILP]	94123 (3523) [*+ILP]
auto	9450 (235532) [MQI]	25271 (117782) [KaFFPaE]	44206 (58891) [KaFFPaE]	74266 (29446) [*+ILP]	118998 (14723) [*+ILP]	169260 (7361) [*+ILP]

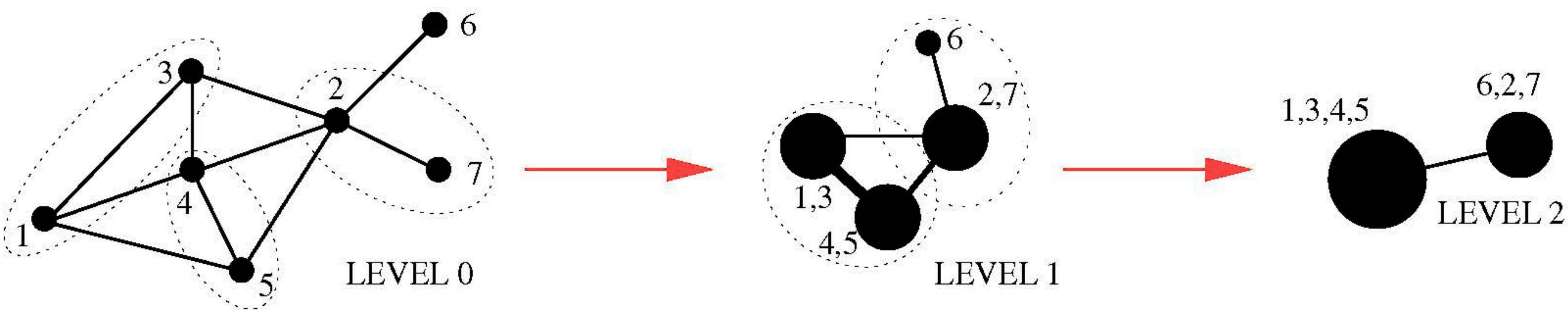
Simple Case: Coarsening by Contractions
(aka strict coarsening)

Intuitive explanation

Two or more vertices are merged if they have a good chance to share *common properties.*

Examples of common properties

- *k*-partitioning/clustering: *i* and *j* belong to the same part
- Network compression/linear arrangement: $|\pi(i) - \pi(j)|$is small



Simple Case: Coarsening by Contractions

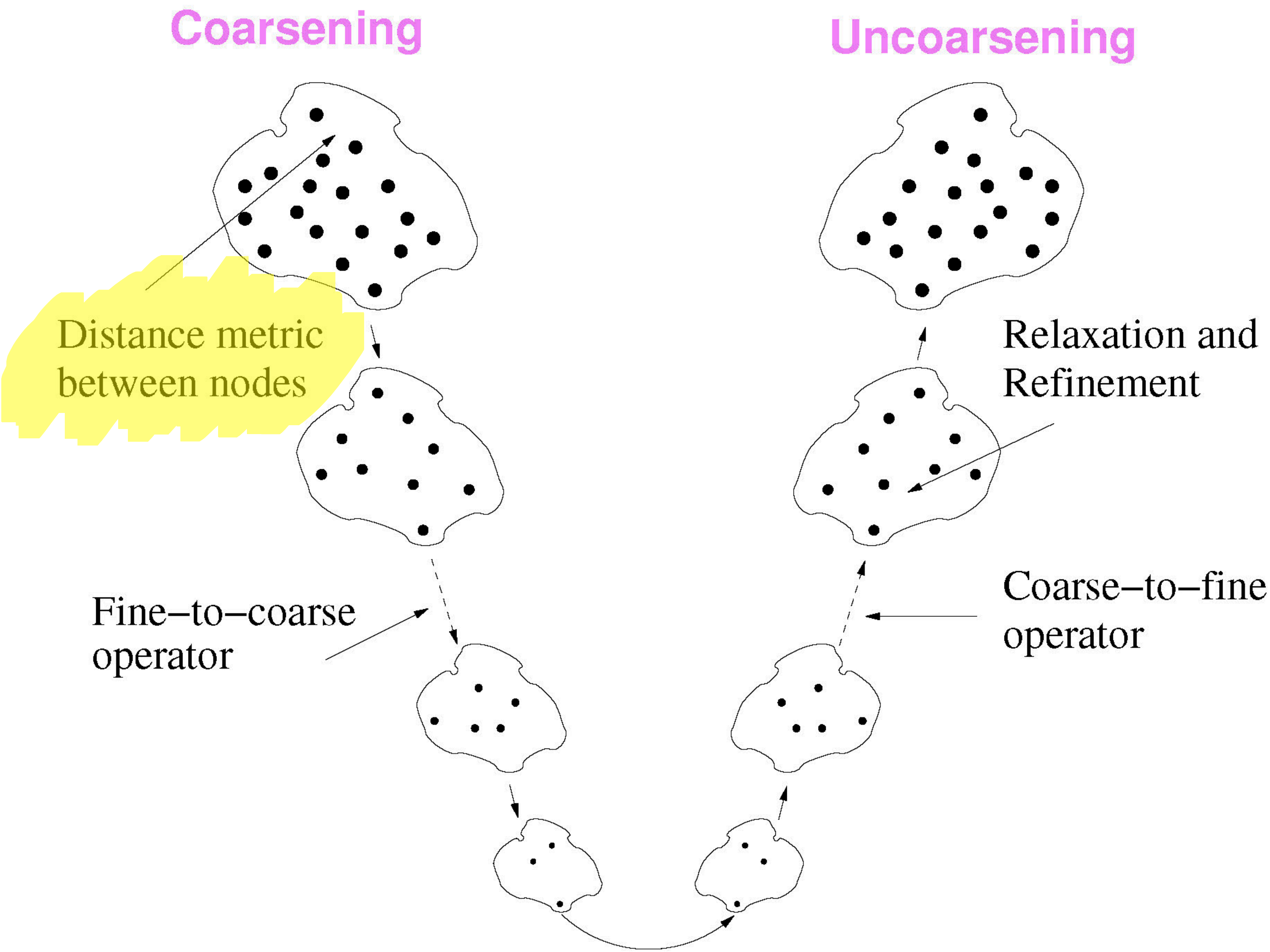
Common problem of strict coarsening methods

They make *local decisions* (i.e., merging) before accumulating the relevant global information. It creates additional difficulty for solving irregular instances when *local* decision contradicts *global* solution.

Existing multilevel solvers

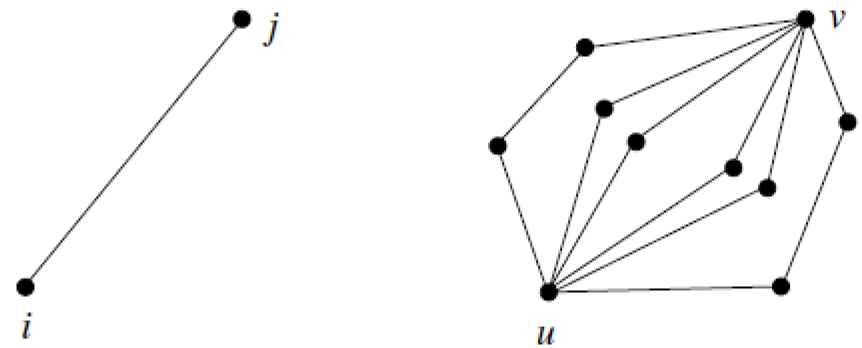
- CHACO by Hendrickson and Leland, since 1993
- METIS by Karypis and Kumar, since 1995
- SCOTCH by Pellegrini, since 1996
- JOSTLE by Walshaw, since 1995

Think globally, act locally



Exact or best possible solution

Models of Connectivity



- Shortest path; All/some indirect paths
- Spectral approaches
- Flow network capacity based approaches
- Random-walk approaches: commute time, first-passage time, etc. (Fouss, Pirotte, Renders, Saerens, ...)
- Interpretations of the diffusion (Lafon, Maggioni, Coifman, ...)
- Effective resistance of a graph (Boyd, Saberi, Spielman, ...)

Stationary Iterative Relaxation

Relaxation process that shows which pair of vertices tends to be 'more connected' than other.

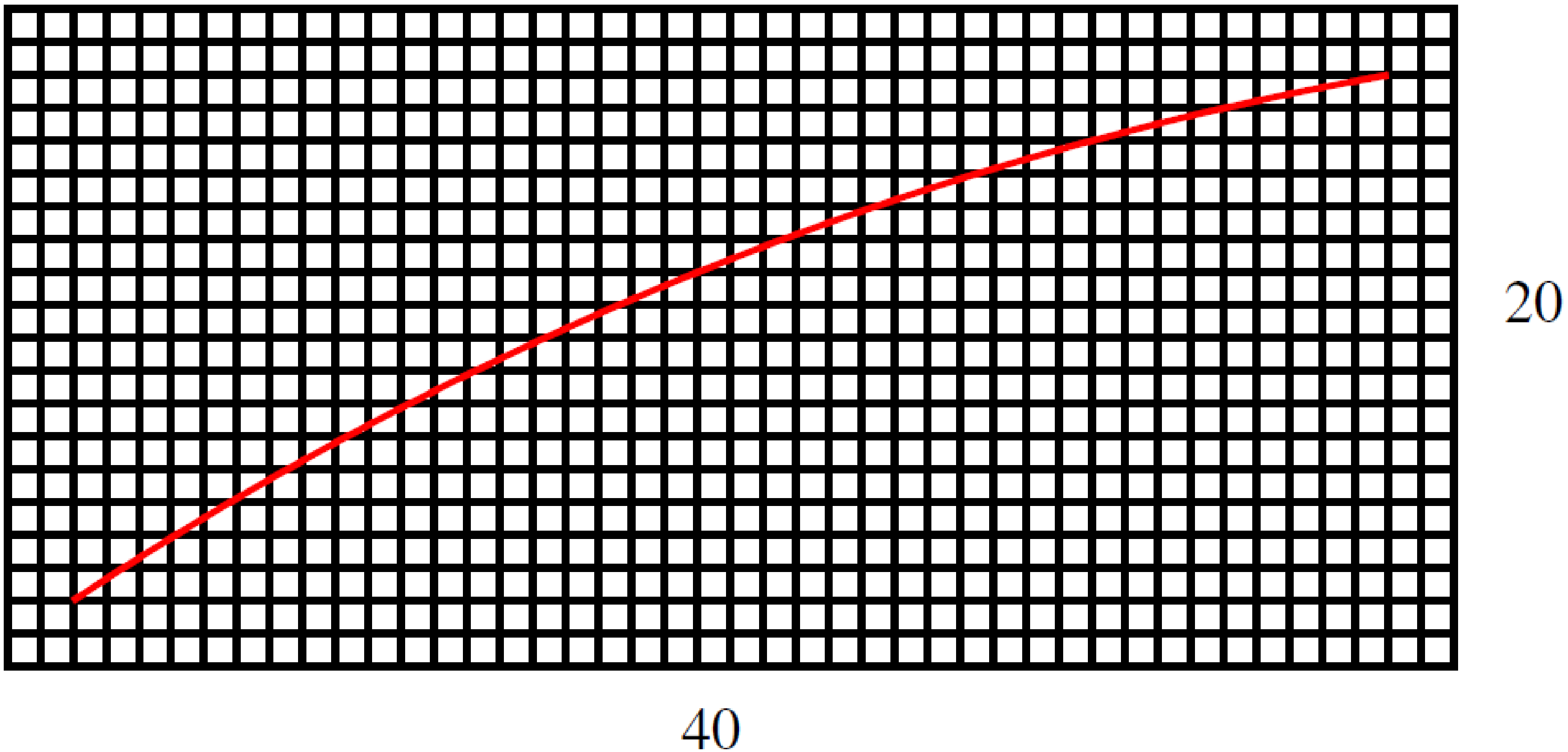
1. $\forall\, i \in V$ define $x_i = rand()$
2. Do k times step 3
3. $\forall\, i \in V\; x_i^k = (1-\omega)x_i^{k-1} + \omega \sum_j w_{ij} x_j^{k-1} / \sum_{ij} w_{ij}$

Conjecture

If $|x_i - x_j| > |x_u - x_v|$ then the local connectivity between u and v is **stronger** *than that between i and j.*

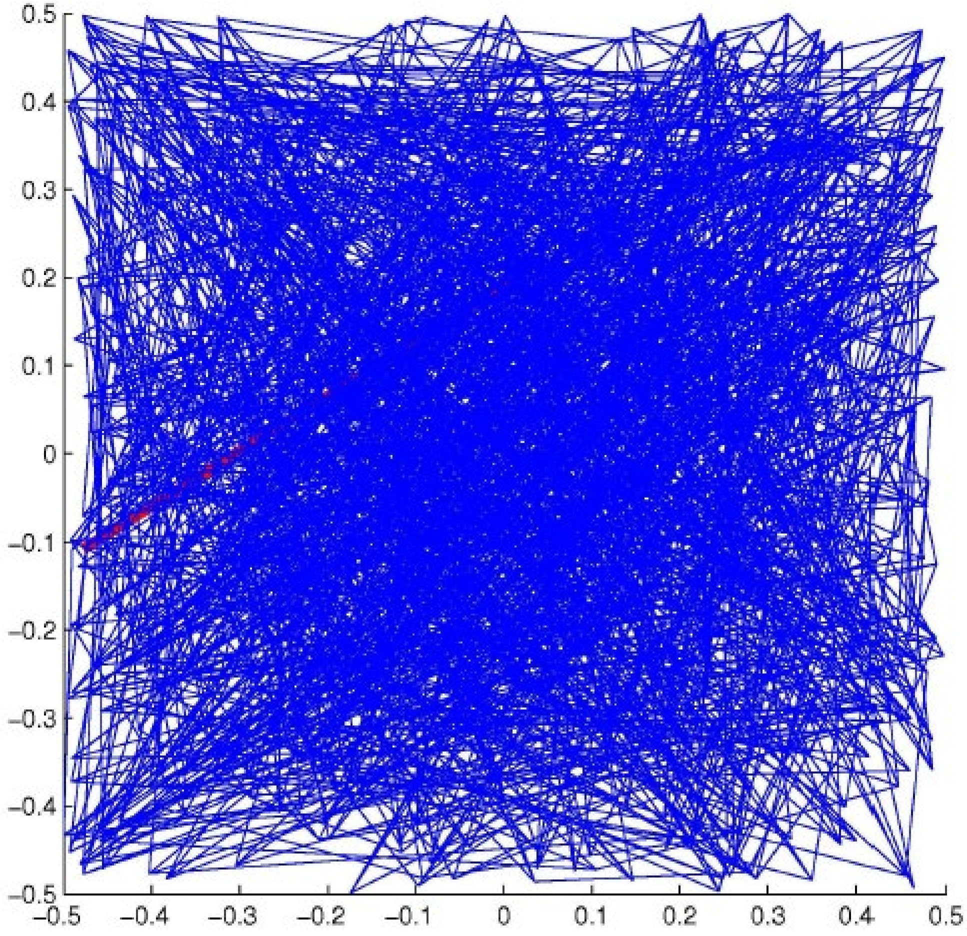
We will call $s_{ij}^{(k)} = |x_i - x_j|$ *the algebraic distance* between i and j after k iterations.

Toy Example: mesh 20x40 + diagonal

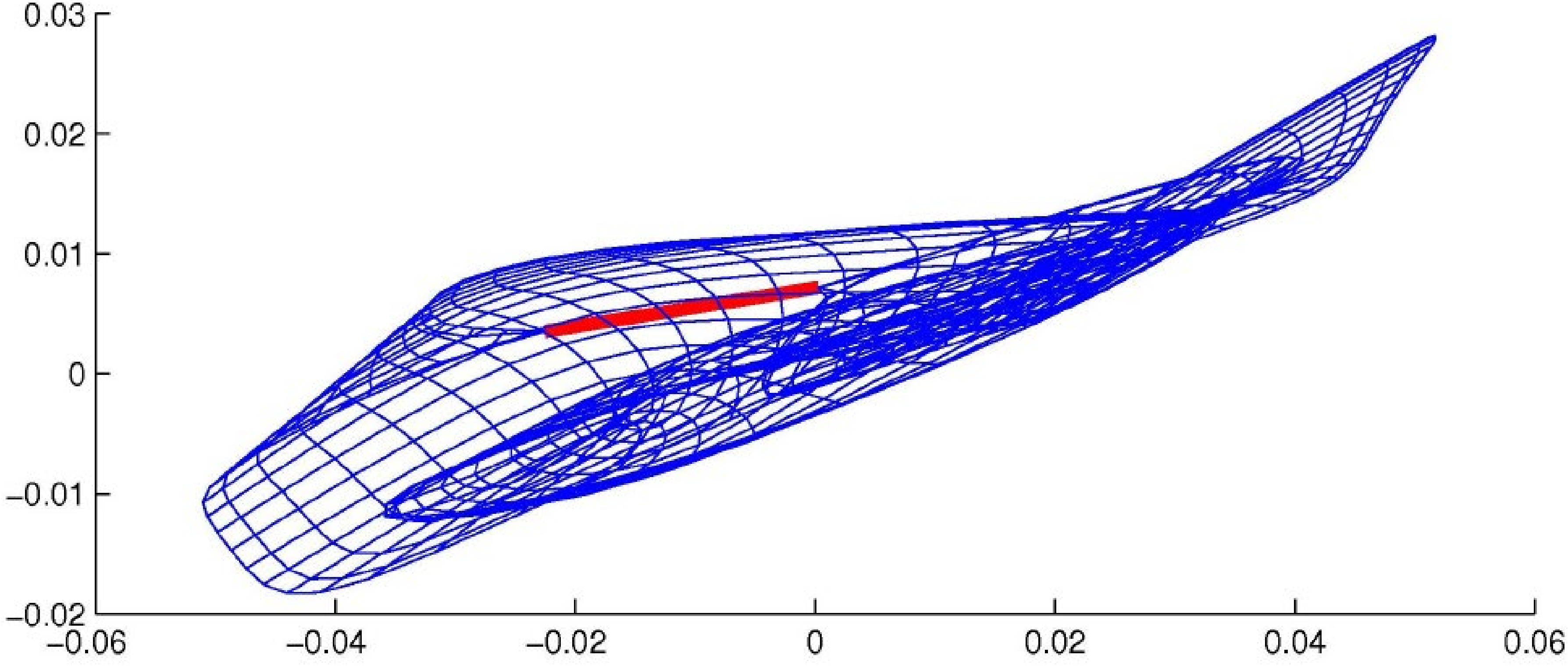


edge weights: red=2, black=1

Random Initialization



… after 10 iterations of Jacobi over-relaxation



Algebraic Distance

The iterators H_* for $x^{(k+1)} = H_* x^{(k)}$ are defined as

diagonal

lower triangular

upper triangular

$$+ U)H_{Gauss\ Seidel} = (D - L)^{-1}U \qquad H_{SOR} = (D/\omega - L)^{-1}((1/\omega - 1)D$$

$$D + L + U)H_{Jacobi} = D^{-1}(L + U) \qquad H_{Jacobi\ OR} = (D/\omega)^{-1}((1/\omega - 1)I$$

en nodes i and Extended p-normed algebraic distance betwee

random initializa- j after k iterations $x^{(k+1)} = H_* x^{(k)}$ on R

tions $x^{(0,r)}$

$$) - x_j^{(k,r)}|^p \Big)^{1/p} \qquad \rho_{ij}^{(k)} := \Big(\sum_{r=1}^{R} |x_i^{(k,r)}$$

- Ron, S, Brandt ``Relaxation-based coarsening and multiscale graph organization'', SIAM MMS, 2011
- Chen, S ``Algebraic distance on graphs'', SIAM J on SC, 2012
- Brandt, Brannick, Kahl, Livshits ``Bootstrap AMG'', SIAM J on SC, 2011
- Bolten et al. ``A Bootstrap Algebraic Multilevel Method for Markov Chains'', SIAM J on SC, 2011
- Shaydulin, Chen, S ``Relaxation-based coarsening for multilevel hypergraph partitioning'', SIAM MMS, 2019

Slow convergence …

Theorem

Given a connected graph, let $(\mu_i, \hat{v}_i)$ be the eigen-pairs of (L, D), labeled in nondecreasing order of the eigenvalues, and assume that $\mu_2 \neq \mu_3 \neq \mu_{n-1} \neq \mu_n$. Unless $\omega = 2/(\mu_2 + \mu_n)$, $\hat{s}_{ij}^{(k)}$ will always converge to a limit $|(e_i - e_j)^T \xi|$ in the order $O(\theta^k)$, for some ξ and $0 < \theta < 1$.

(i) *If* $0 < \omega < \frac{2}{(\mu_3+\mu_n)}$, *then* $\xi \in \mathrm{span}\{\hat{v}_2\}$ *and* $\theta = \frac{1-\omega\mu_3}{1-\omega\mu_2}$;

(ii) *If* $\frac{2}{(\mu_3+\mu_n)} \leq \omega < \frac{2}{(\mu_2+\mu_n)}$, *then* $\xi \in \mathrm{span}\{\hat{v}_2\}$ *and* $\theta = -\frac{1-\omega\mu_n}{1-\omega\mu_2}$;

(iii) *If* $\frac{2}{(\mu_2+\mu_n)} < \omega < \min\{\frac{2}{(\mu_2+\mu_{n-1})}, \frac{2}{\mu_n}\}$, *then* $\xi \in \mathrm{span}\{\hat{v}_n\}$ *and* $\theta = -\frac{1-\omega\mu_2}{1-\omega\mu_n}$;

(iv) *If* $\frac{2}{(\mu_2+\mu_{n-1})}) \leq \omega < \frac{2}{\mu_n}$, *then* $\xi \in \mathrm{span}\{\hat{v}_n\}$ *and* $\theta = \frac{1-\omega\mu_{n-1}}{1-\omega\mu_n}$.

Theorem

Given a graph, let $(\mu_i, \hat{v}_i)$ be the eigen-pairs of (L, D), labeled in nondecreasing order of the eigenvalues. Denote $\hat{V} = [\hat{v}_1, \ldots, \hat{v}_n]$. Let $x^{(0)}$ be the initial vector of the JOR process, and let $a = \hat{V}^{-1} x^{(0)}$ with $a_1 \neq 0$. If the following two conditions are satisfied:

$$1 - \omega\mu_n \geq 0 \quad \textit{and} \quad f_k := \frac{\alpha {r_k}^{2k}(1-r_k)^2}{1 + \alpha {r_k}^{2k}(1+r_k)^2} \leq \frac{1}{\kappa},$$

where $\alpha = \left(\sum_{i\neq 1} a_i^2\right) / (4a_1^2)$, r_k *is the unique root at* $[0, 1]$ *of*

$$2\alpha r^{2k+2} + 2\alpha r^{2k+1} + (k+1)r - k = 0\,,$$

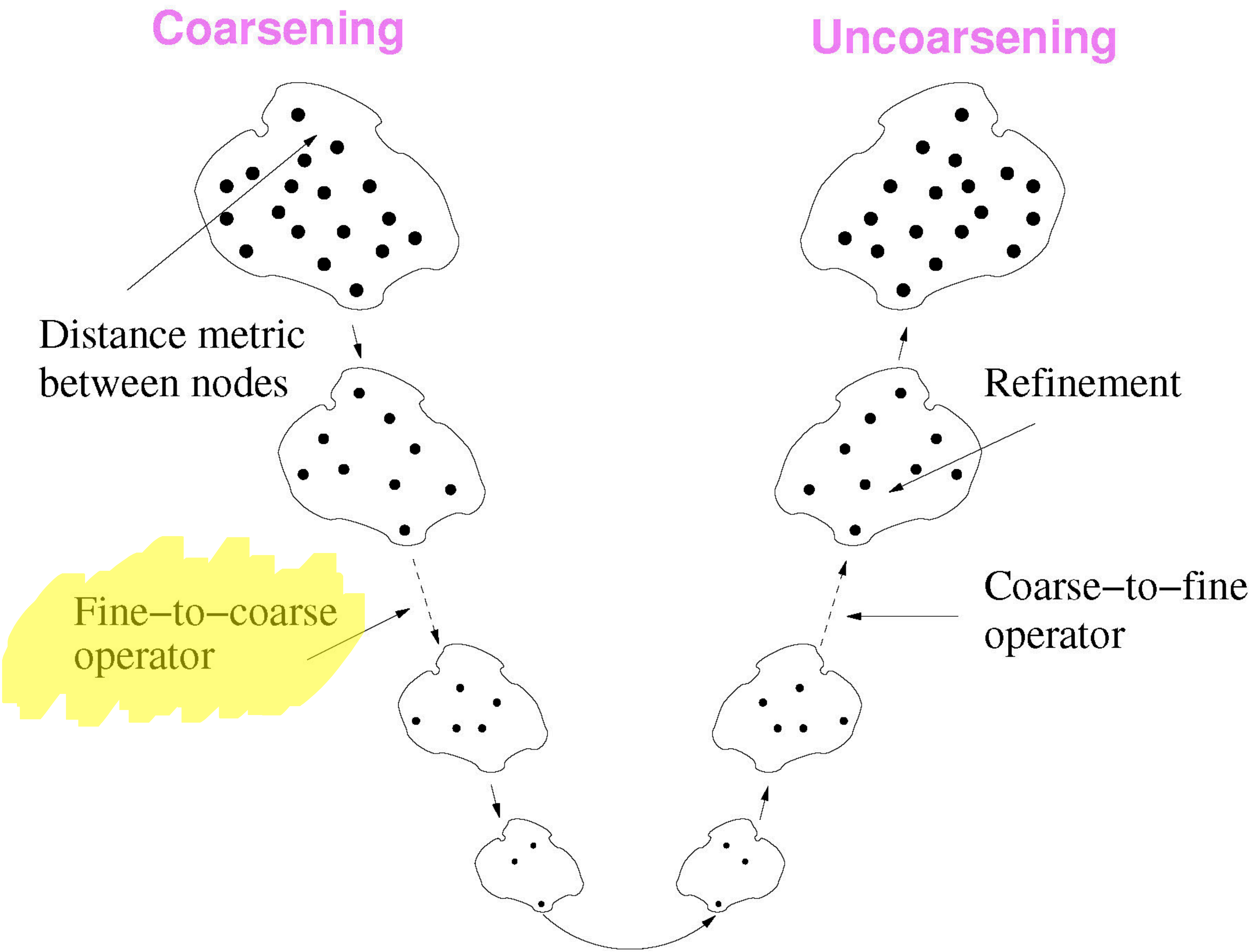
then $1 - \left\langle \frac{x^{(k)}}{\|x^{(k)}\|}, \frac{x^{(k+1)}}{\|x^{(k+1)}\|} \right\rangle^2 \leq \frac{4 cond(D) f_k}{(1 + cond(D) f_k)^2}$.

but fast stabilization which is what we need in multilevel framework

- Ron, S, Brandt ``Relaxation-based coarsening and multiscale graph organization'', SIAM MMS, 2011
- Chen, S ``Algebraic distance on graphs'', SIAM J on SC, 2012

Four main questions

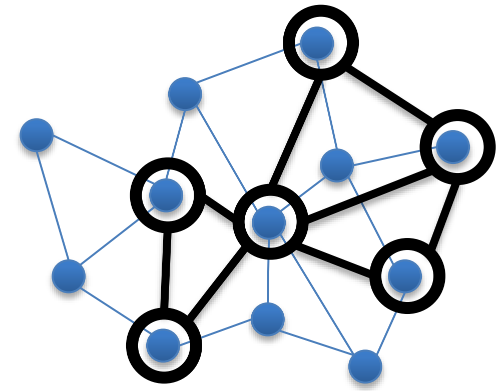
Think globally, act locally



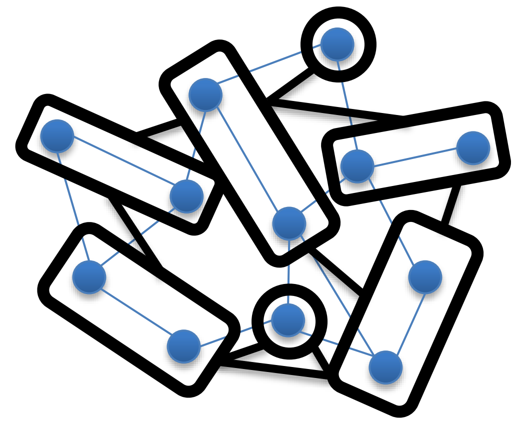
Exact or best possible solution

Types of Coarsening

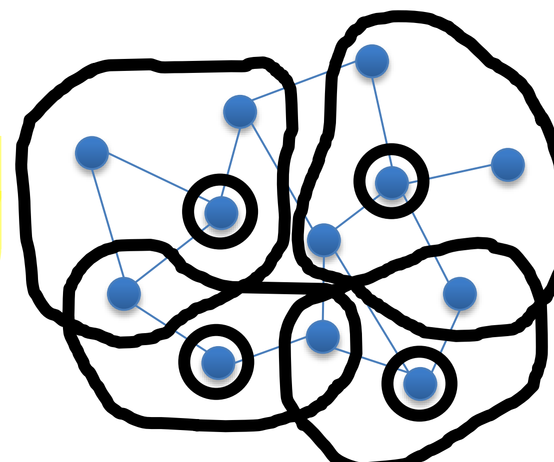
1. Iterative selection of some variables to the coarse level (e.g., independent sets)



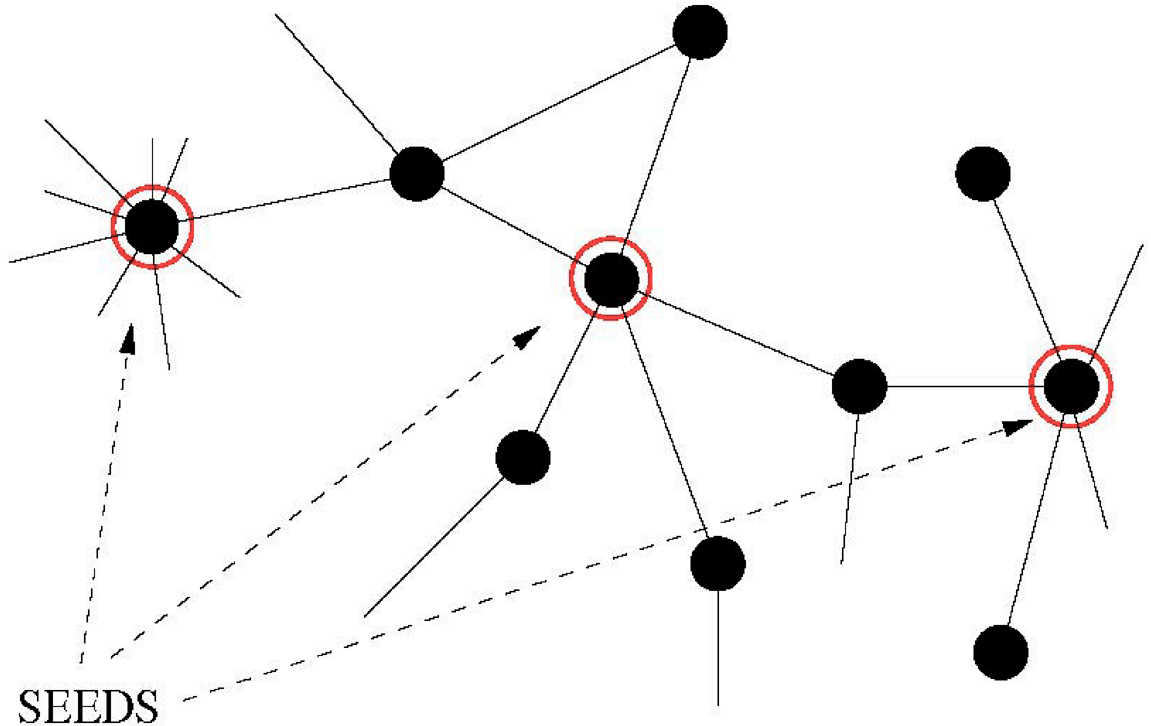
2. Strict coarsening (merging pairs) with some smart distance function (similar to some graph partitioning multilevel techniques)



3. AMG algebraic distance based coarsening of graph Laplacian



AMG: coarse variables

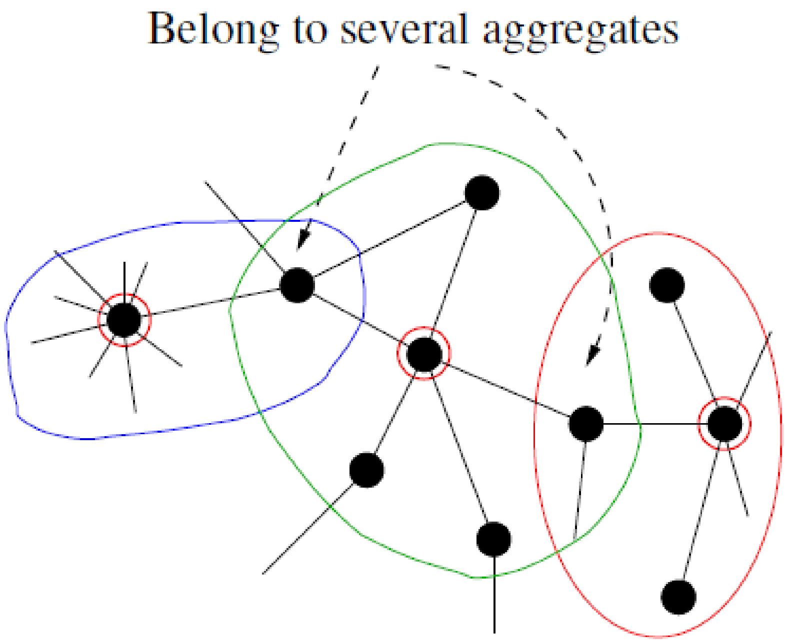


- Choose a dominating set $C \subset V$ s.t. all others from $F = V \setminus C$ are "strongly coupled" to C
- "Strongly coupled" $=$ Kernel coupling $\cdot$ algebraic distances ρ_{ij}

- ➢ Ron, S, Brandt ``Relaxation-based coarsening and multiscale graph organization'', SIAM MMS, 2011
- ➢ Chen, S ``Algebraic distance on graphs'', SIAM J on SC, 2012
- ➢ Brandt, Brannick, Kahl, Livshits ``Bootstrap AMG'', SIAM J on SC, 2011
- ➢ Bolten et al. ``A Bootstrap Algebraic Multilevel Method for Markov Chains'', SIAM J on SC, 2011
- ➢ Shaydulin, Chen, S ``Relaxation-based coarsening for multilevel hypergraph partitioning'', SIAM MMS, 2019

Interpolation weights

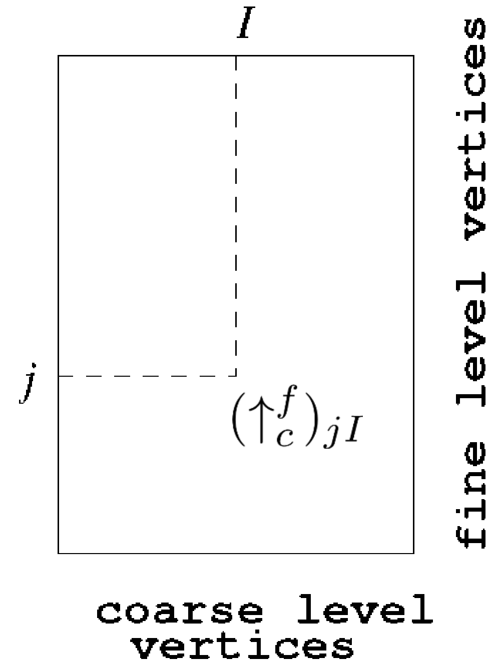
$$(\uparrow_c^f)_{iJ} = \begin{cases} \frac{\rho_{ij}^{-1}}{\left(\sum_{k \in N(i)} \rho_{ik}^{-1}\right)} & i \in F,\ j \in N(i) \\ 1 & i \in C,\ j = i \\ 0 & \text{otherwise} \end{cases}$$



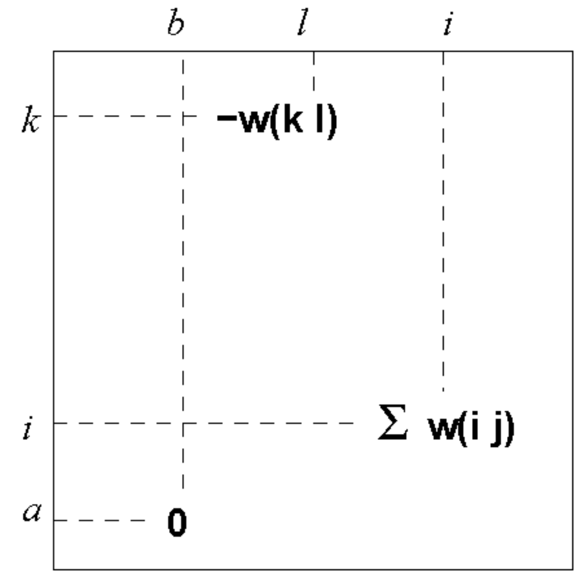
- Define the interpolation weights of all vertices
- In some sense, the interpolation weights (iw) are the probabilities of a vertex to share a common property with the aggregates it belongs to.

Coarse Graph

$\uparrow_c^f$ - restriction operator



L_f - weighted Laplacian at level f

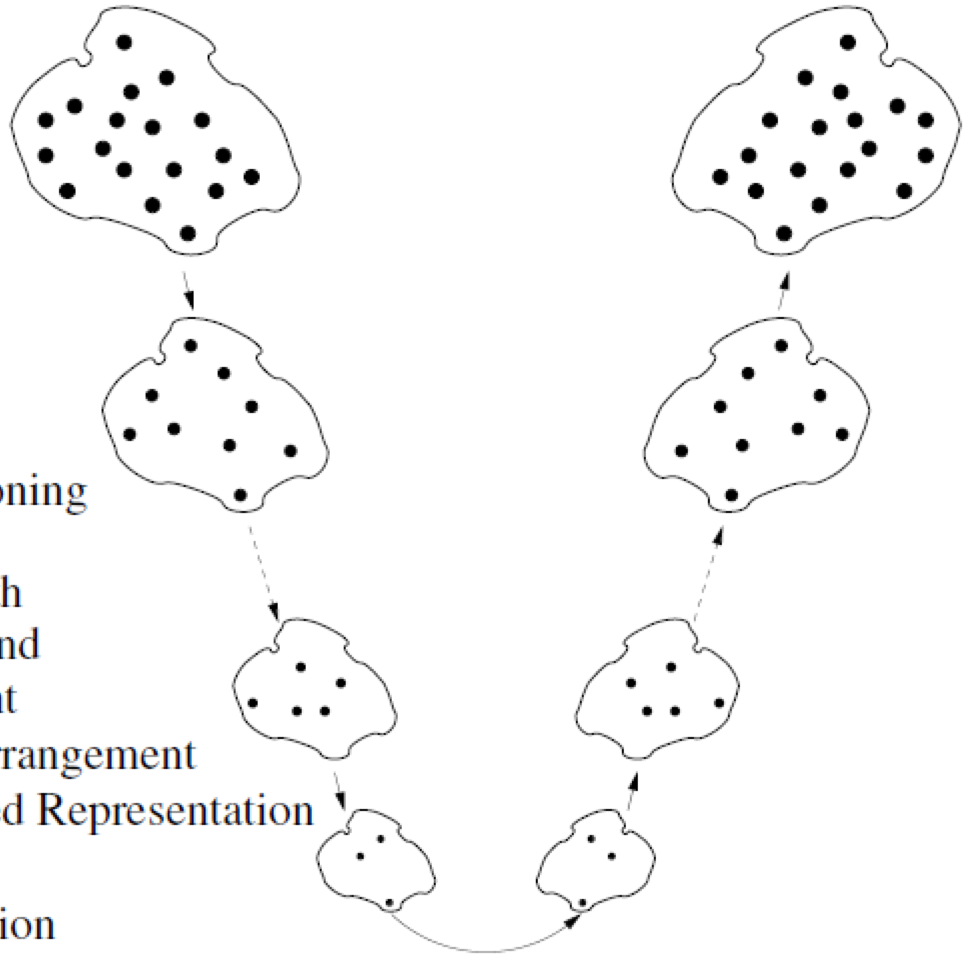


Coarse graph Laplacian

$$L_c = (\uparrow_c^f)^T L_f \uparrow_c^f$$

$$w_{IJ} = \sum_{l,k} (\uparrow_c^f)_{Il} \cdot w_{lk} \cdot (\uparrow_c^f)_{kJ}$$

Coarsening is problem independent

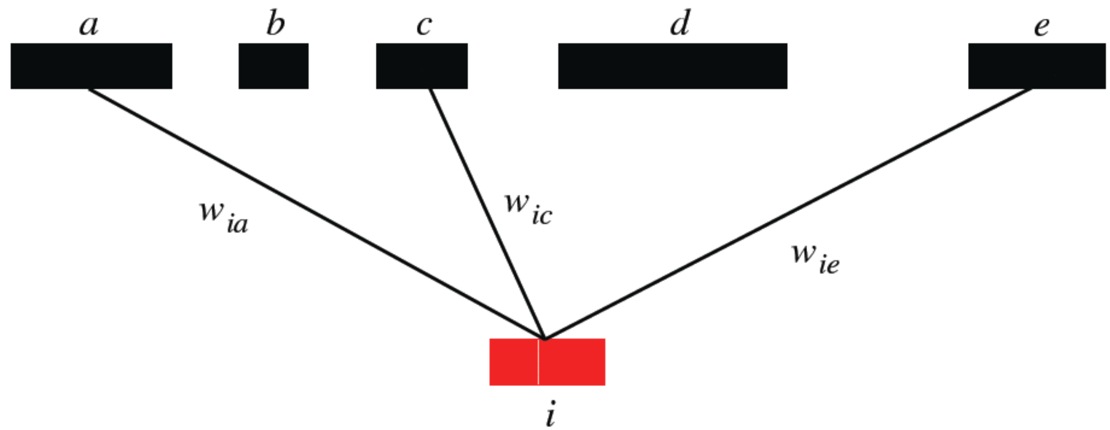


Interpolation
Relaxation
Refinement

Minimum k–Partitioning
Minimum 2–sum
Minimum Bandwidth
Minimum Workbound
Minimum Wavefront
Minimum Linear Arrangement
Network Compressed Representation
Clustering
Manifold Identification

Exact solution

MLogA Uncoarsening: Minimizing the Contribution of One Node



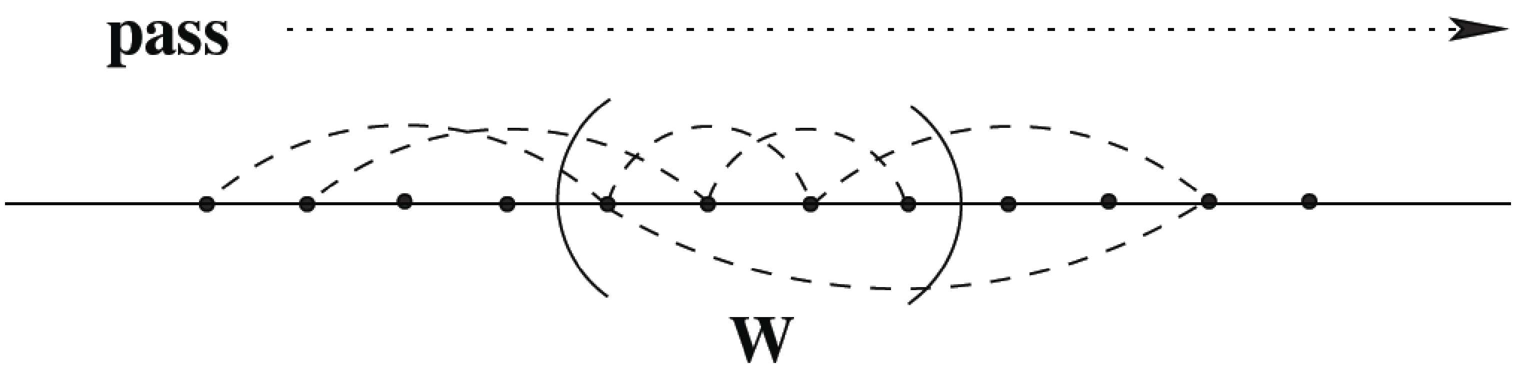
N_i – the set of ith neighbors with assigned coordinates $\tilde{x}_j$. To minimize the local contribution of i to the total energy, we have to assign to it a coordinate x_i that minimizes

$$\sum_{j \in N_i} w_{ij} \lg |x_i - \tilde{x}_j| \ . \tag{1}$$

$\forall j \in N_i,\ x_i = \tilde{x}_j \ \Rightarrow$ (1) is $-\infty$, we resolve this by setting

$$x_i = \tilde{x}_t \iff t = \arg\min_{k \in N_i} \sum_{k \neq j \in N_i} w_{kj} \lg |\tilde{x}_k - \tilde{x}_j| \ .$$

MLogA Uncoarsening: Refinement



Find π of W that

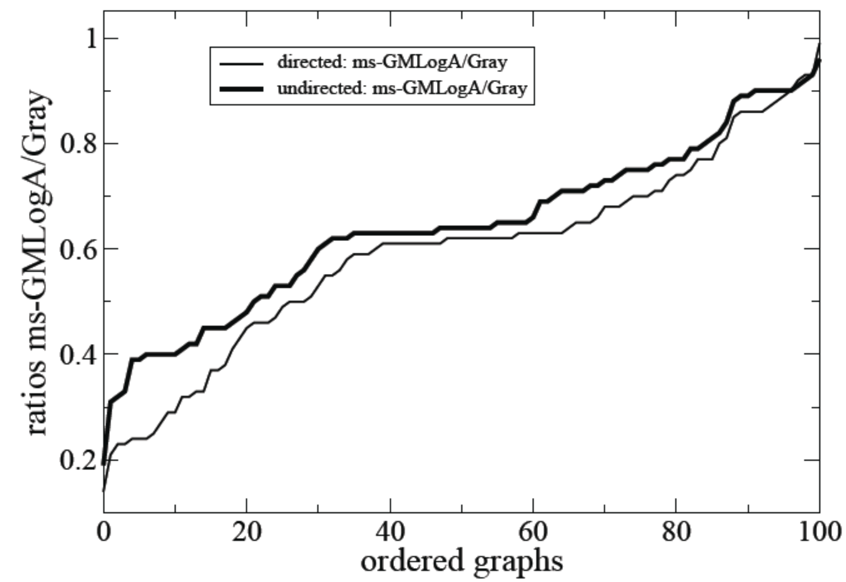
minimizes $$\sum_{ij \in W} w_{ij} \lg |x_i - x_j| + \sum_{i \in W, j \notin W} w_{ij} \lg |x_i - \tilde{x}_j|$$

subject to $$x_i = v_i/2 + \sum_{k, \pi(k) < \pi(i)} v_k$$

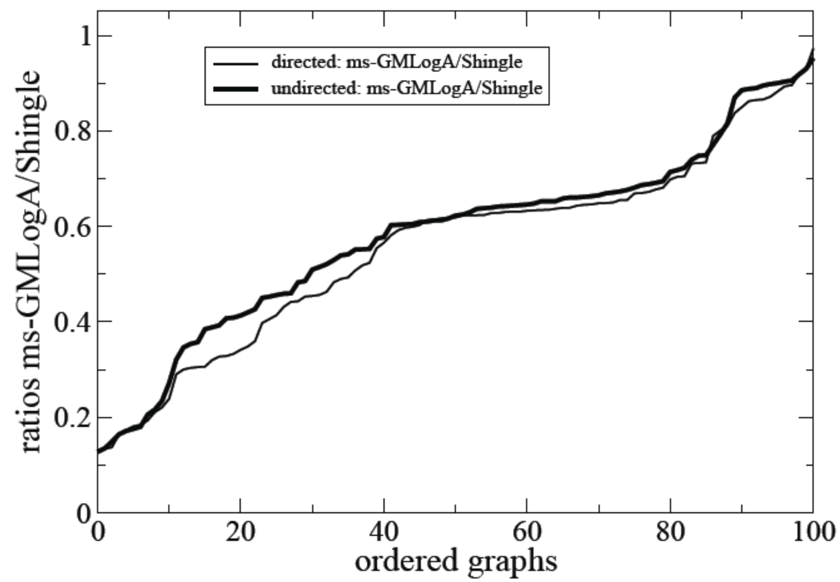
What are the most competitive algorithms?

- Randomized ordering - usually comes from parallel network crawling (fast to obtain, bad for performance)
- Lexicographical - network traversal for some order of neighbours such as BFS and DFS (easy to calculate, can be good for networks with excellent locality)
- Gray ordering - inspired by Gray coding when two successive vectors differ by exactly one bit (easy to calculate, good for Web-like (or good locality) networks)
- Shingle ordering - brings nodes with similar neighborhoods together, uses Jaccard coefficient $J(A, B) = |A \bigcap B| / |A \bigcup B|$ to measure the similarity (works good in ”preferential attachment models” when rich gets richer).
- LayeredLPA - label propagation algorithm is similar to the algebraic distance (usually better than previous methods)

Computational Results: Multiscale MLogA vs Gray/Shingle

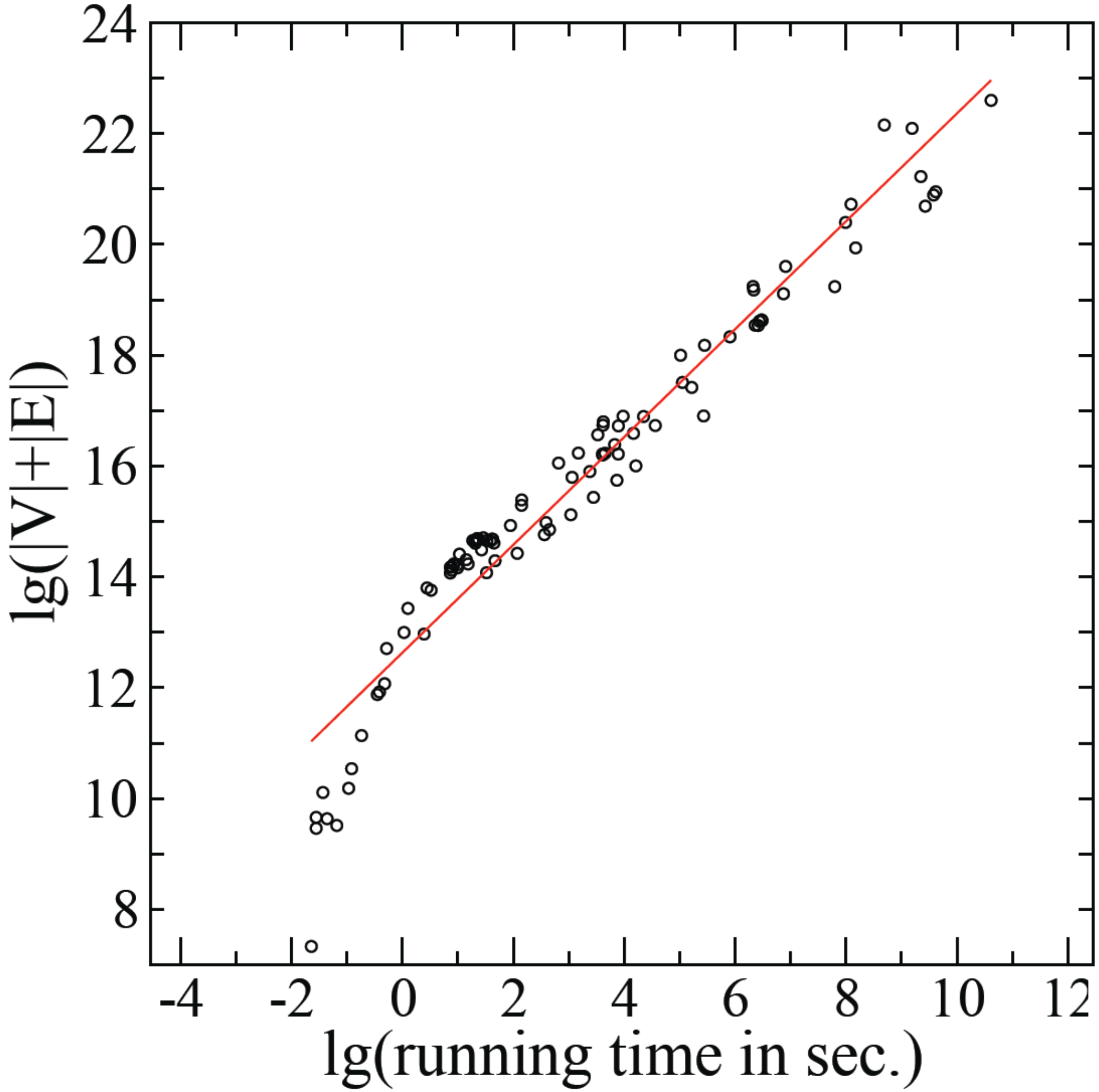


(a) Gray ordering vs ms-GMLogA

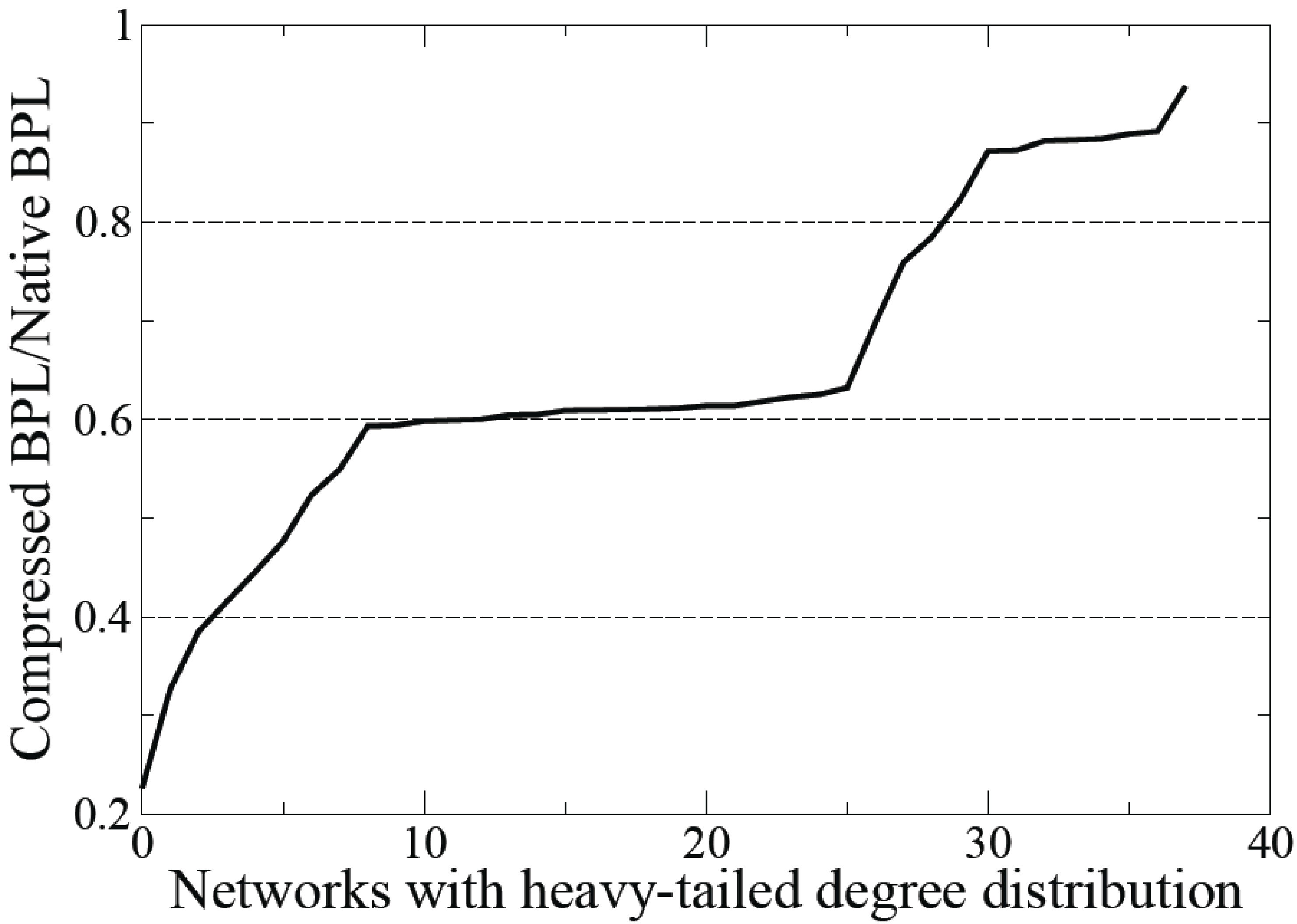


(b) Double shingle vs ms-GMLogA

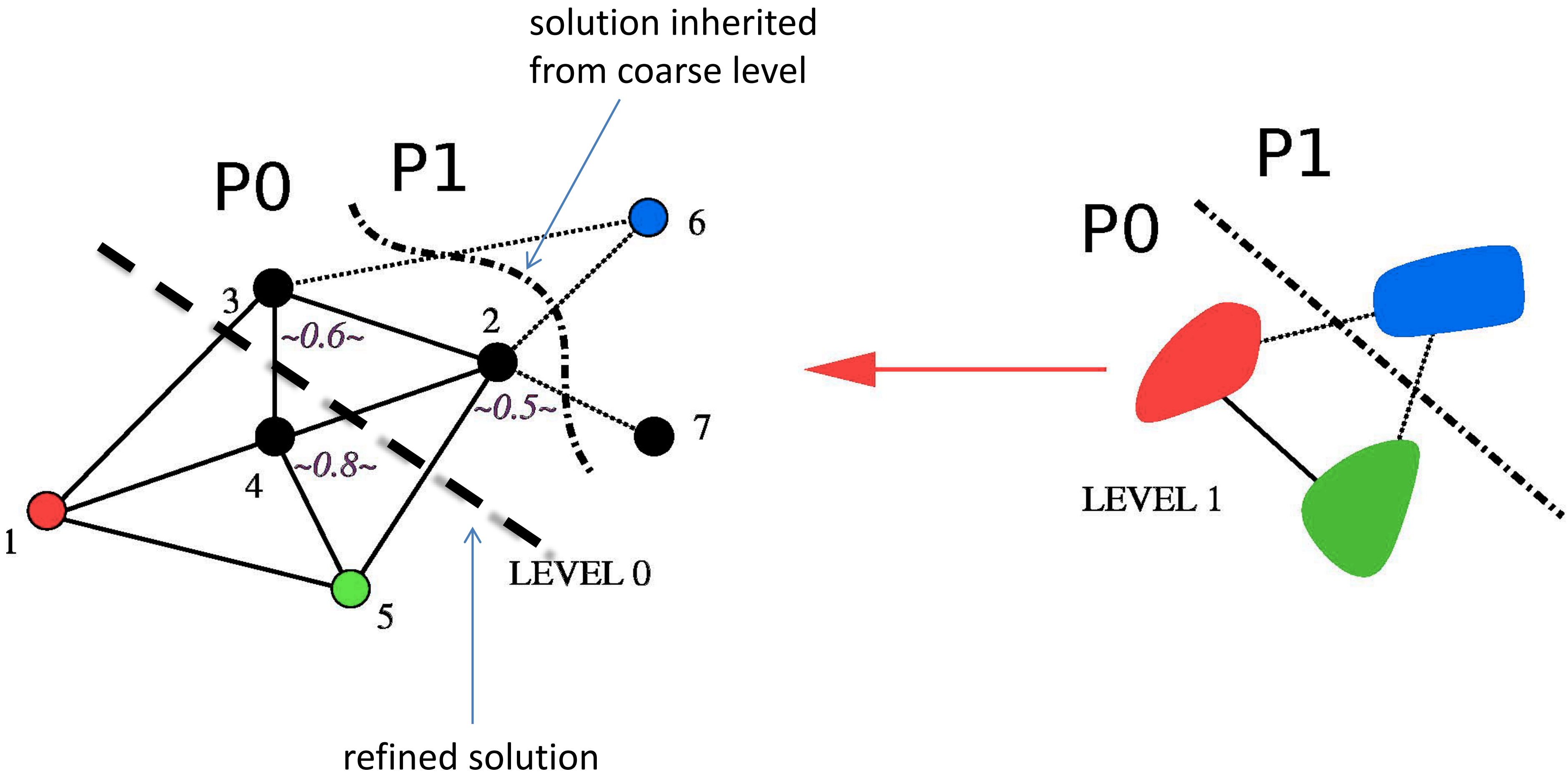
Scalability



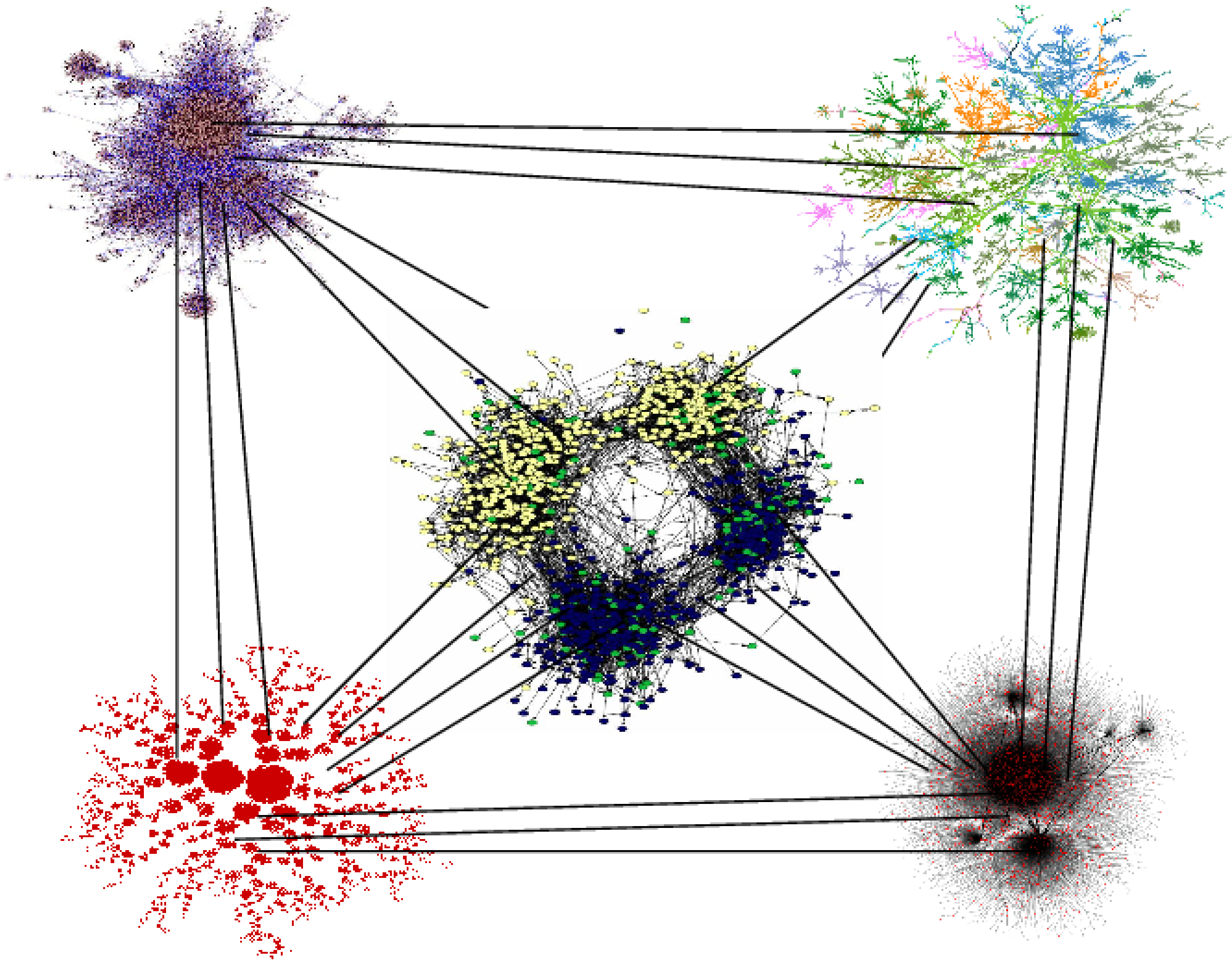
Heavy-tailed degree distributions; Are they compressible?



Refinement for *k*-partitioning

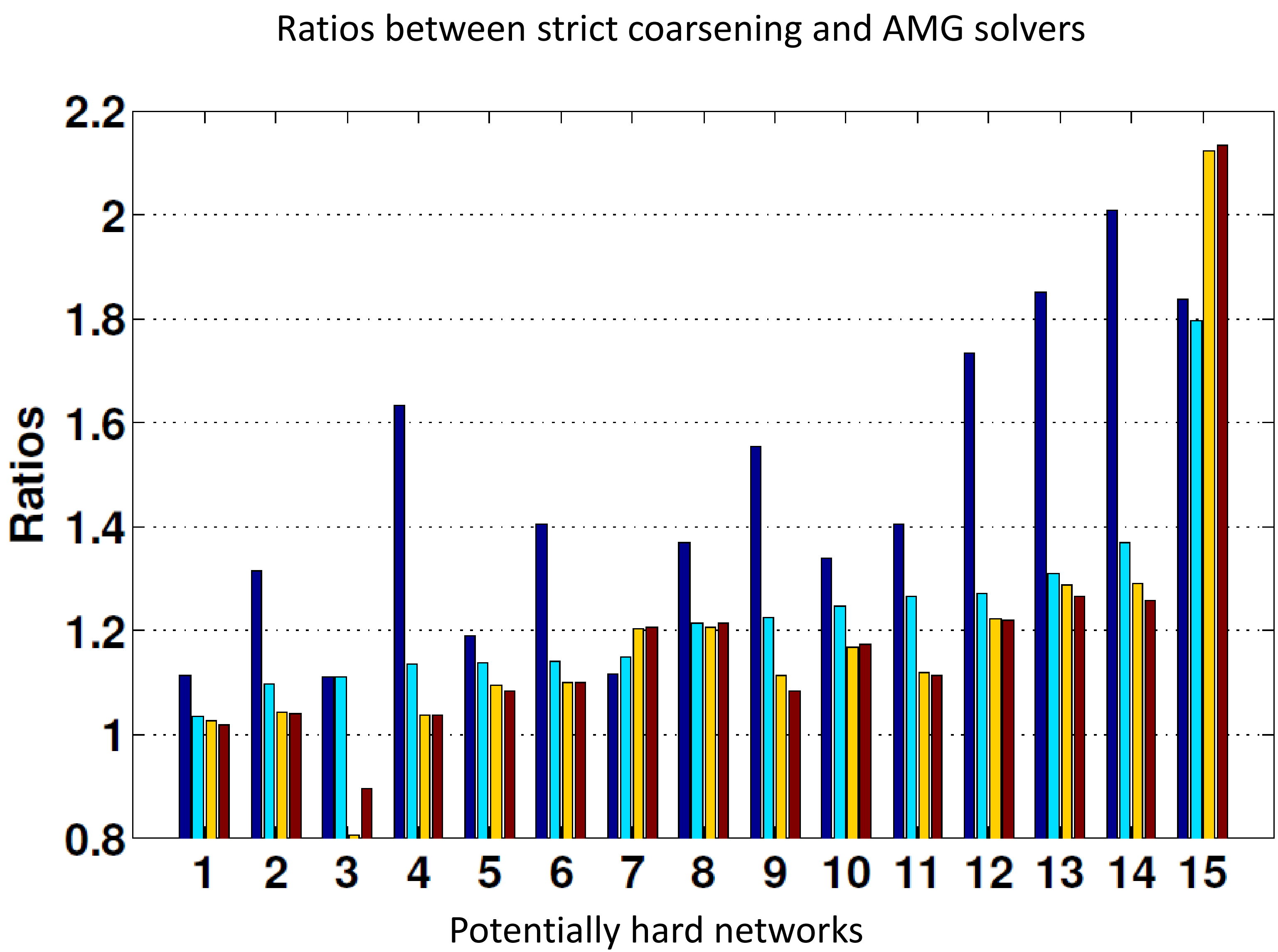


Potentially hard graphs for multilevel *k*-partitioning/clustering

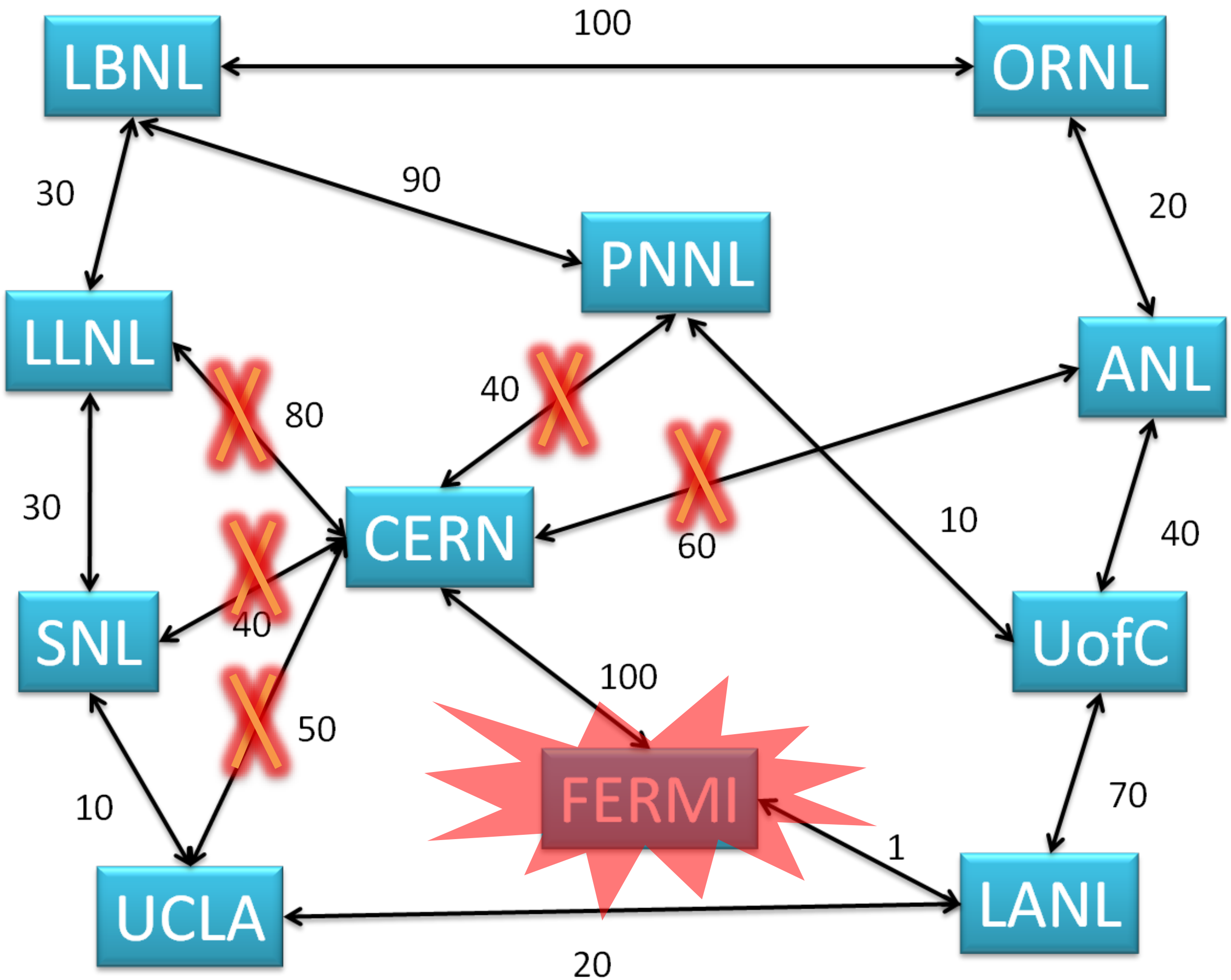


S, Sanders, Schulz "Advanced coarsening schemes for graph partitioning", 2012

Potentially hard graphs for multilevel algorithms, *k=4*

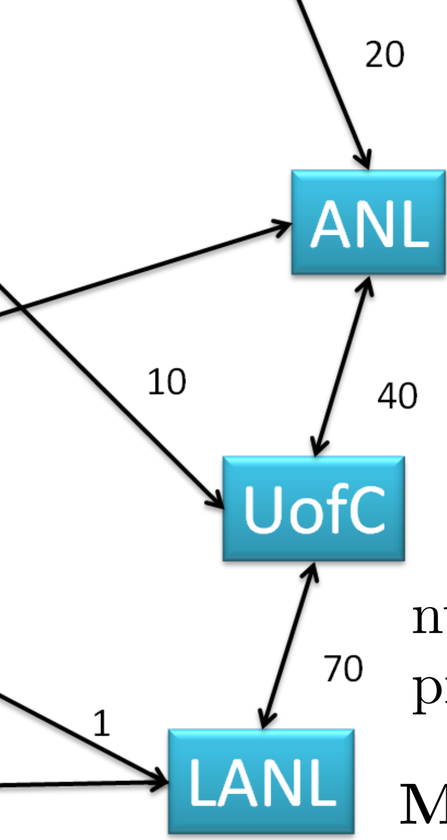


Response to Epidemics and Cyber Attacks



Open Science Grid: collaboration network example

Goldberg, Leyffer, S "Optimal Response to Epidemics and Cyber Attacks in Networks", 2015



site i closed/open	$x_i \in \{0,1\}$
infection probability at i	ϕ_i
number of shared users	w_{ij}
probability of $j \to i$ spread	p_{ij}

Model

connections between open sites, i.e., the utility of network

$$\underset{x}{\text{maximize}} \quad \sum_{ij \in E} w_{ij} x_i x_j$$

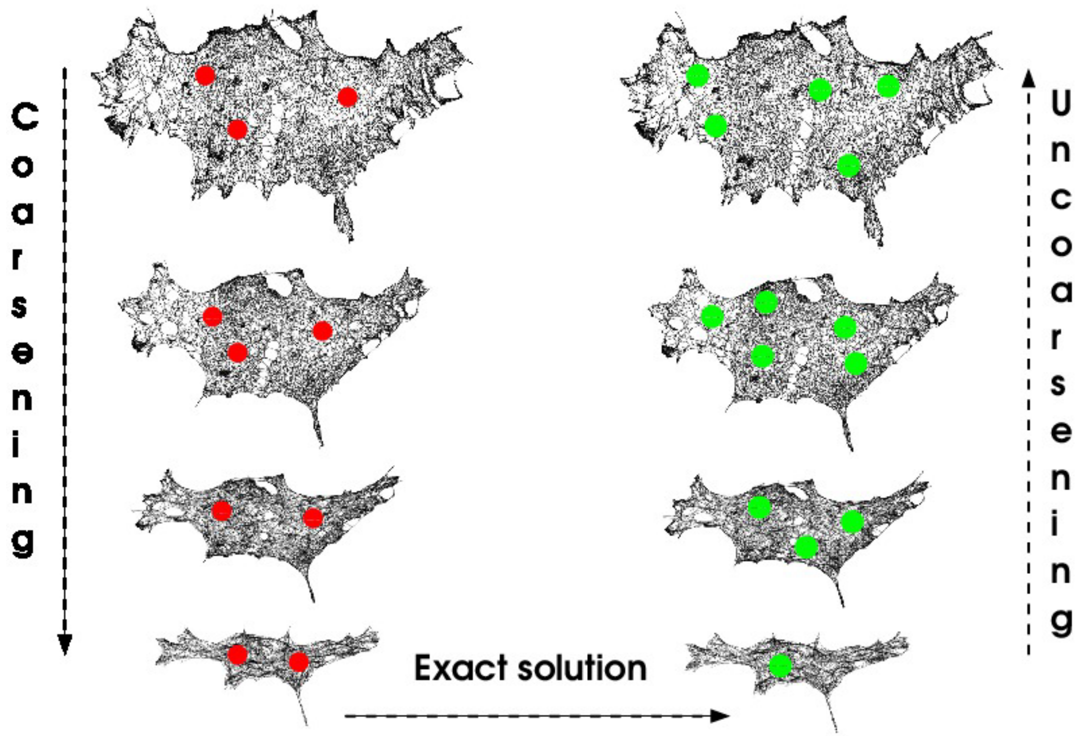
$$\text{subject to} \quad x_i - \prod_{j \in N(i)} (1 - p_{ij}\phi_j x_j) \le t_i \quad \forall i \in V$$

$$x \in \{0,1\}^n$$

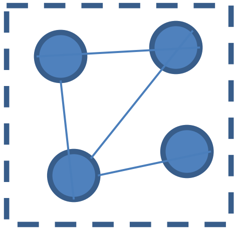
infection at node *i* is less than some constant

Multiscale Algorithm

function MSSolve(G)
if G is small **then**
    $S_f \leftarrow$ solve the problem exactly
else
    order infected nodes
    find coarse variables
    $G_c \leftarrow$ create coarse graph
    $S_c \leftarrow$ MSSolve(G_c)
    $S_f \leftarrow$ Interpolate(S_c)
    $S_f \leftarrow$ LocalRefinement(S_f)
end if
return S_f



Coarsening



New linear term

Links between accumulated nodes

Coarse model

$$\underset{X}{\text{maximize}} \quad \sum_{ij \in E_c} W_{ij} X_i X_j + \sum_{i \in V_c} A_i X_i$$

$$\text{subject to} \quad X_i - \prod_{j \in N(i)} (1 - P_{ij} \Phi_j X_j) \leq T_i \quad \forall i \in V_c$$

$$X \in \{0,1\}^n$$

$$P_{ij},\ W_{ij},\ \Phi_i,\ X_i,\ T_i \ \leftarrow \ \text{AMG coarsening, Galerkin reinforced by algebraic distance}$$

Uncoarsening

$$\underset{x}{\text{maximize}} \quad \sum_{i,j \in S} w_{ij} x_i x_j + \sum_{i \in S, j \notin S} w_{ij} x_i \tilde{x}_j + \sum_{i \in S} a_i x_i$$

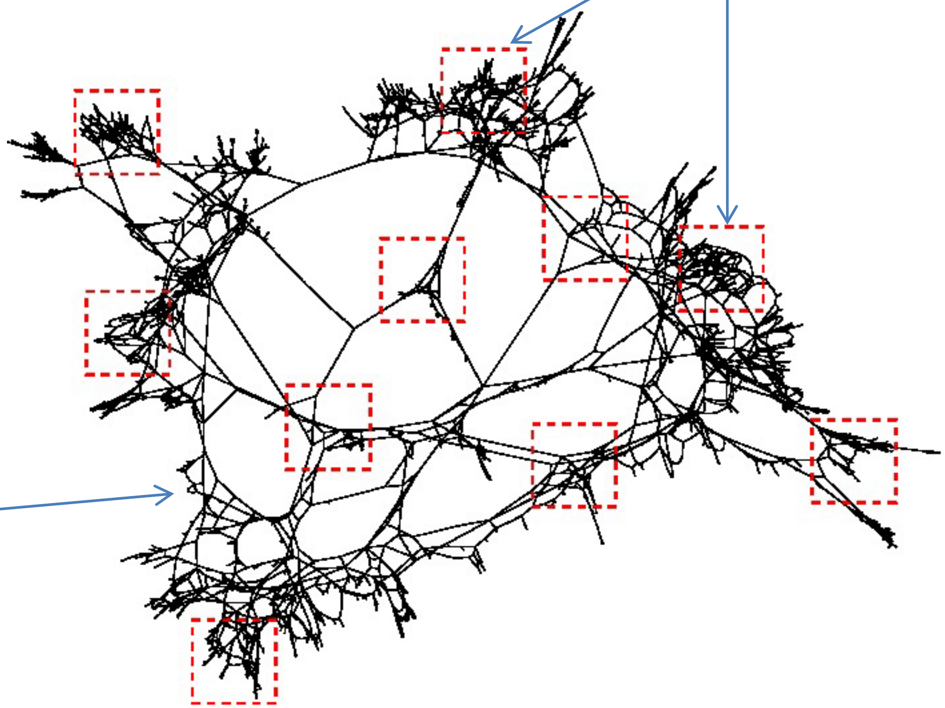
$$\text{subject to} \quad x_i - k_i \prod_{\substack{j \in N(i) \\ j \in S}} (1 - p_{ij}\phi_{j,t-1} x_j) \leq b_i \ \forall i \in V$$

$$x_i \in \{0, 1\} \ \forall i \in V$$

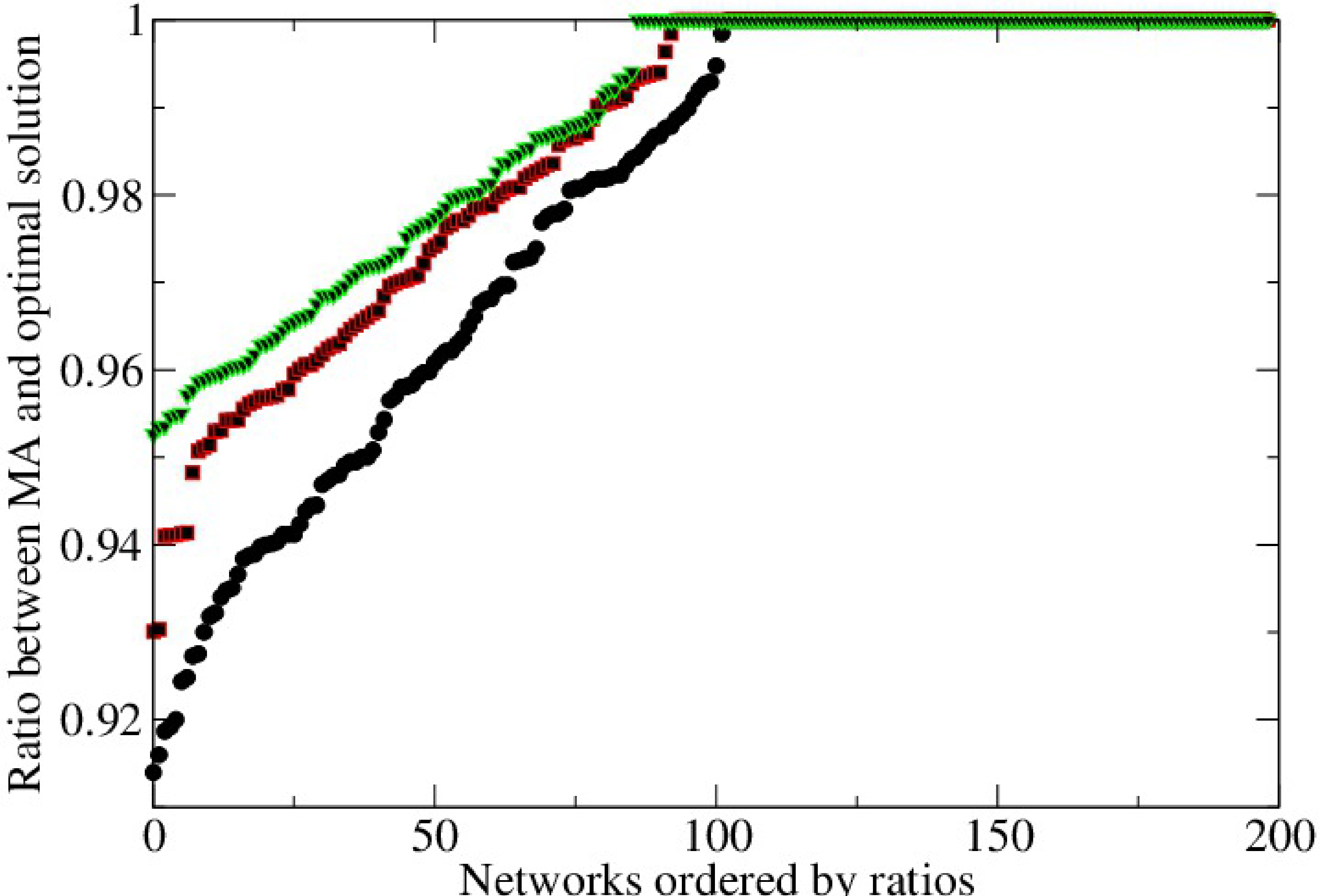
$$k_i = \prod_{j \in N(i), j \notin S} (1 - p_{ij}\phi_{j,t-1}\tilde{x}_j)$$

Local refinement

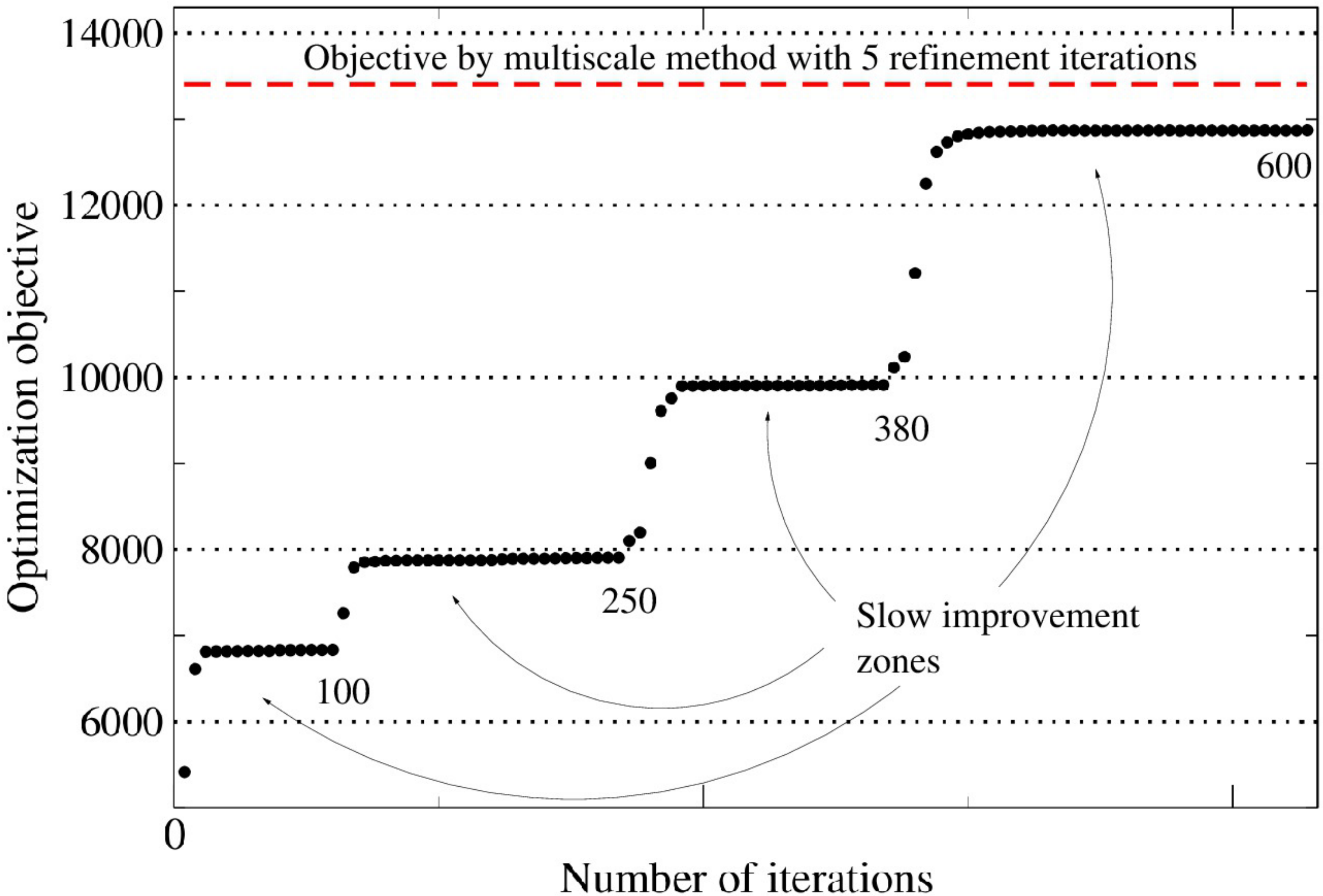
Boundary conditions



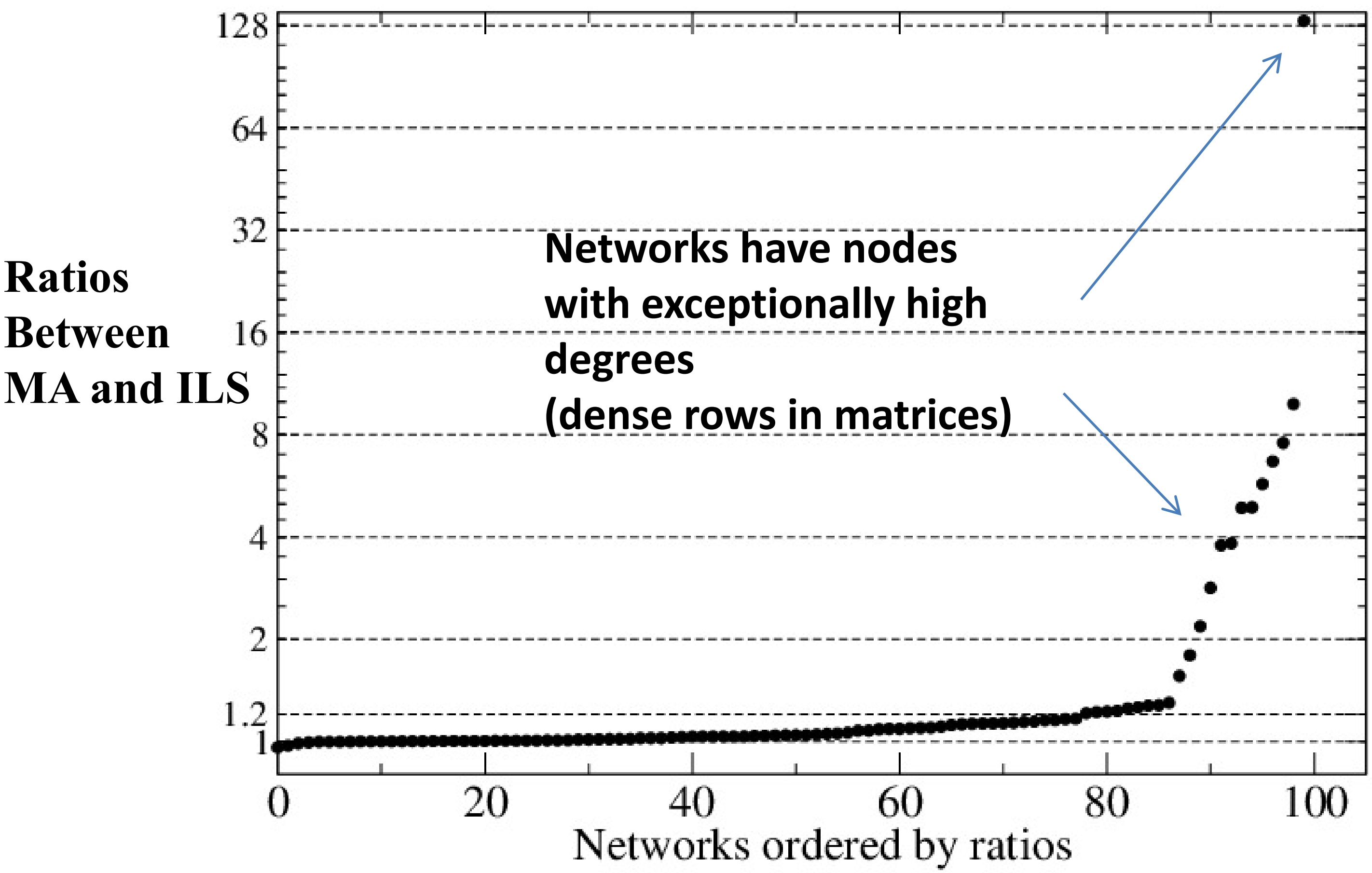
Small random graphs, < 80 nodes, < 400 edges
Erdos-Renyi, Barabasi-Albert, and R-MAT models



Iterated Local Search vs Multiscale
HIV spread model



Large-scale networks

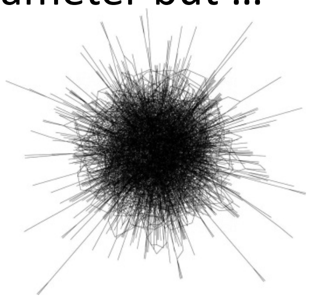


Network Generation, A Practical Approach

Theoretical questions
- What processes form a network?
- How to predict its future structure?
- Why should network have property X?

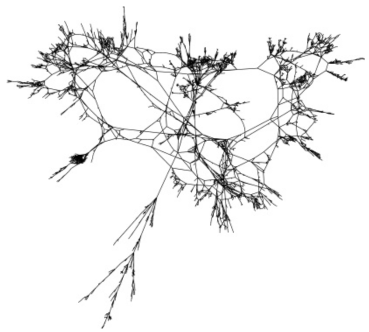
Practical question
- **Will my algorithm/heuristic work on networks created by similar processes?**

This artificial network has similar degrees, some eigs, diameter but …

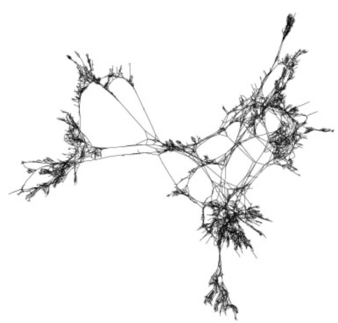


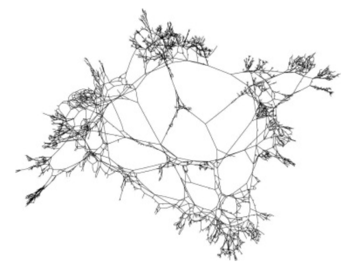
Is it really similar to the original network?

Artificial network



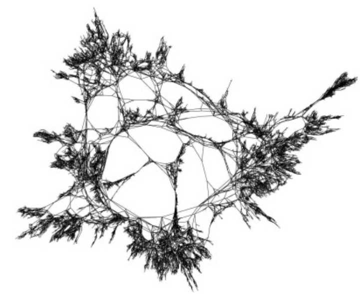
Artificial network





Original network

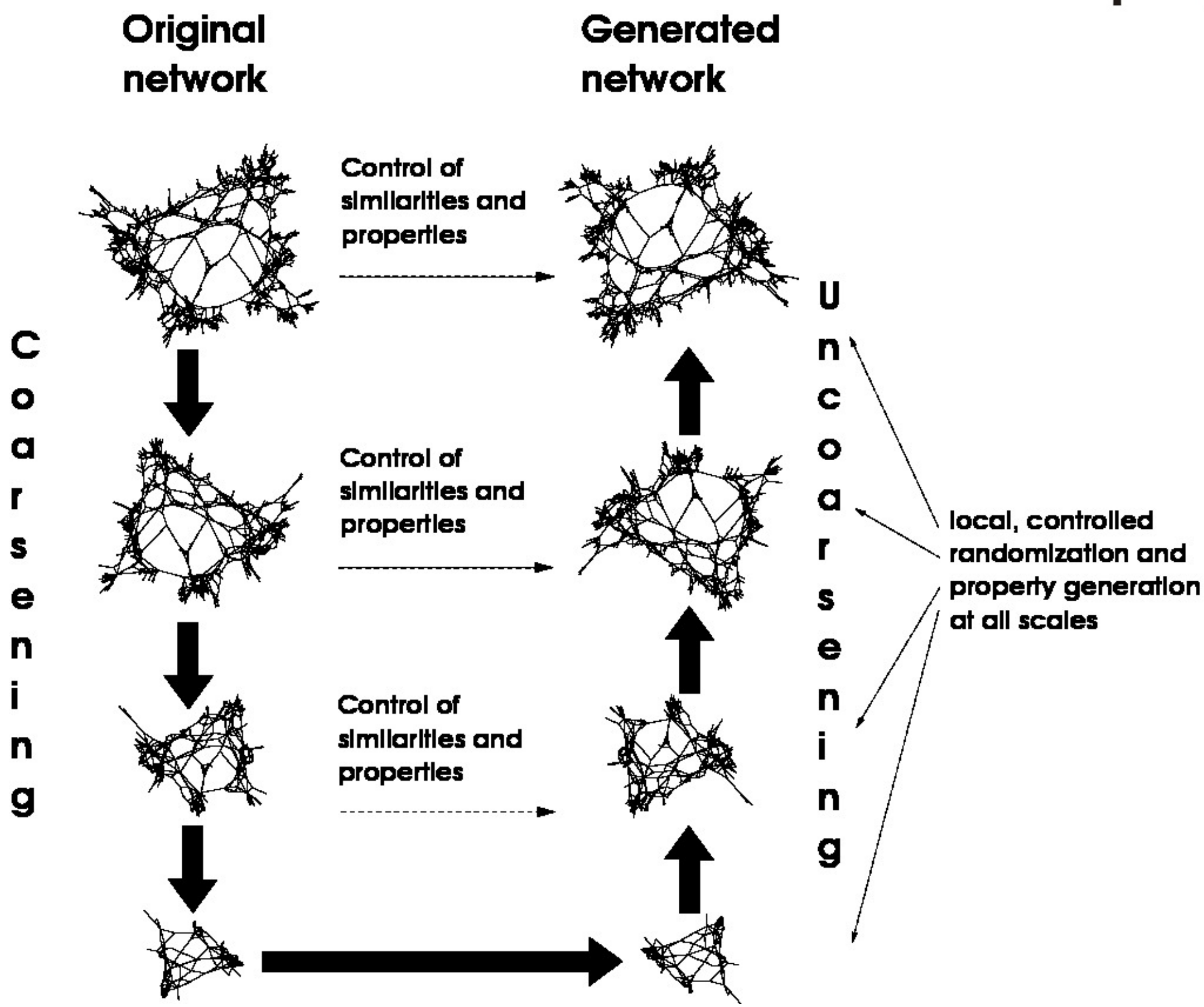
Artificial network



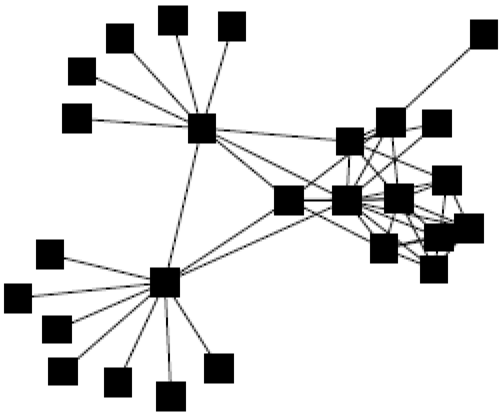
Artificial network

Properties taken into account by most of the existing network generators: degree distribution, clustering coefficient, some eigenvalues, diameter, etc.
They are different at different resolutions!

MUlti-
SCale
Entropic
Ne**T**work
G**E**n**E**rato**R**

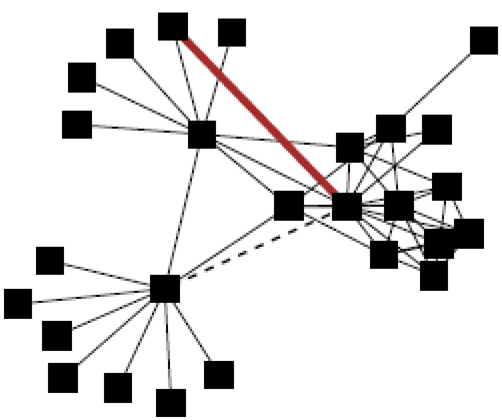


http://www.cs.clemson.edu/~isafro/musketeer



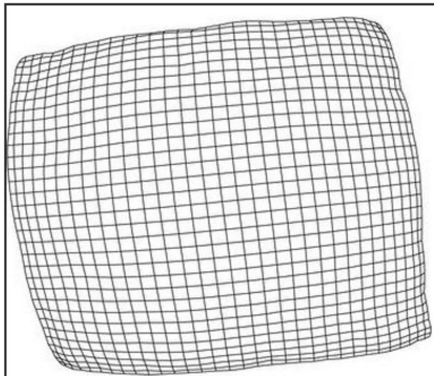
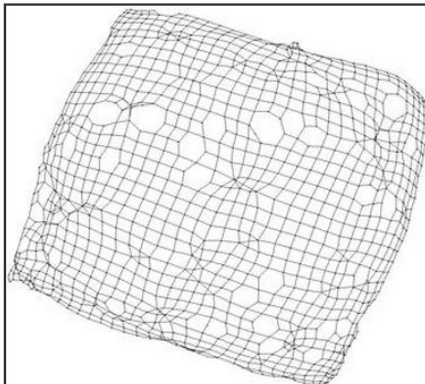
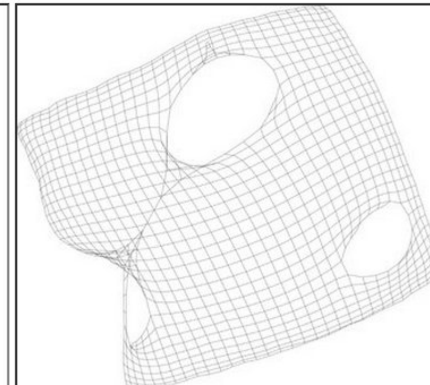
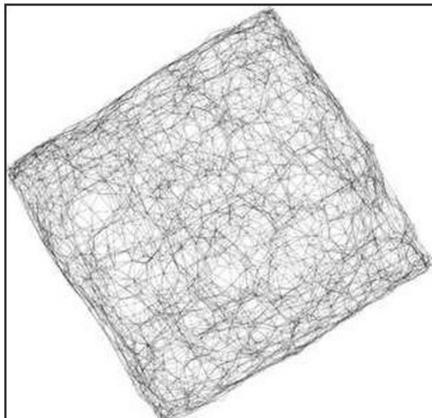
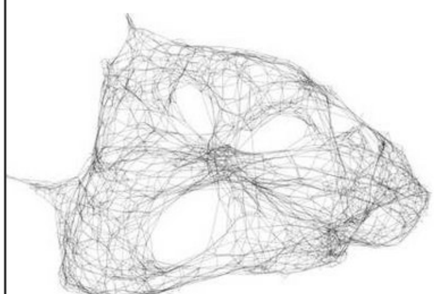
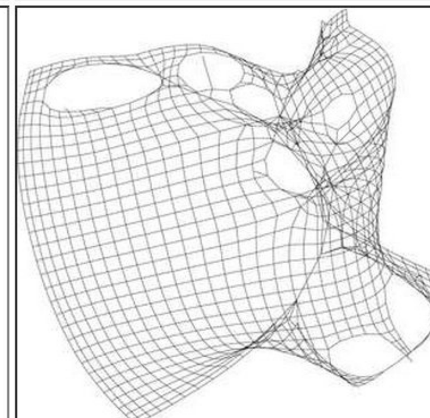
To create a new edge *uv*

- $d_2(i, j)$:= second shortest path between two neighbors
- Estimate $\mathbb{P}[d_2(i, j) = k]$

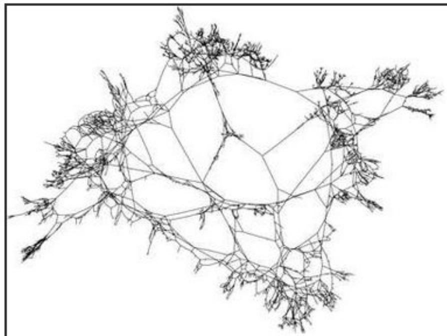
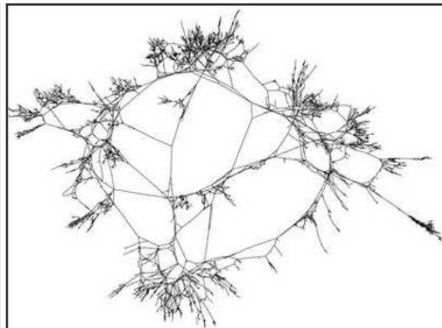
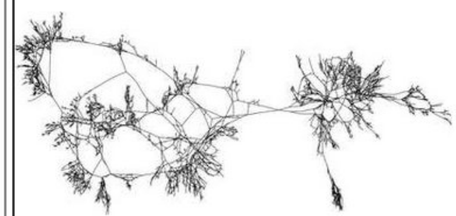
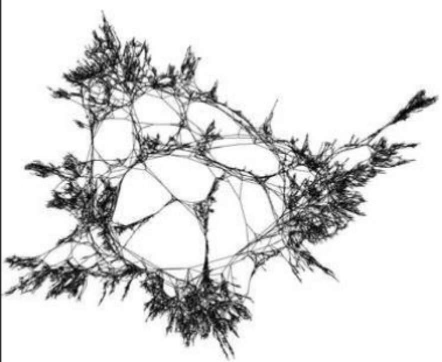
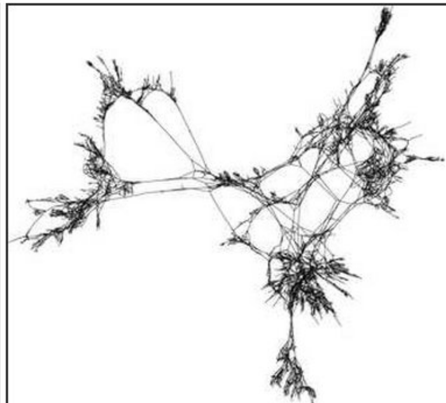
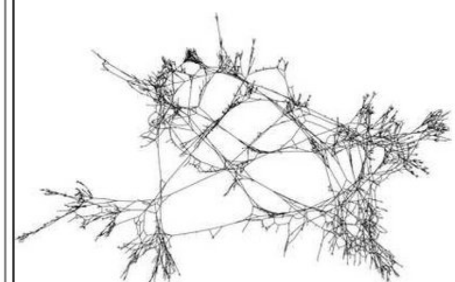


1. Sample *x* from the estimated distribution
2. Randomly select *u* and find *v* within distance *x*
3. Create edge *uv* with edge weight from a given distribution

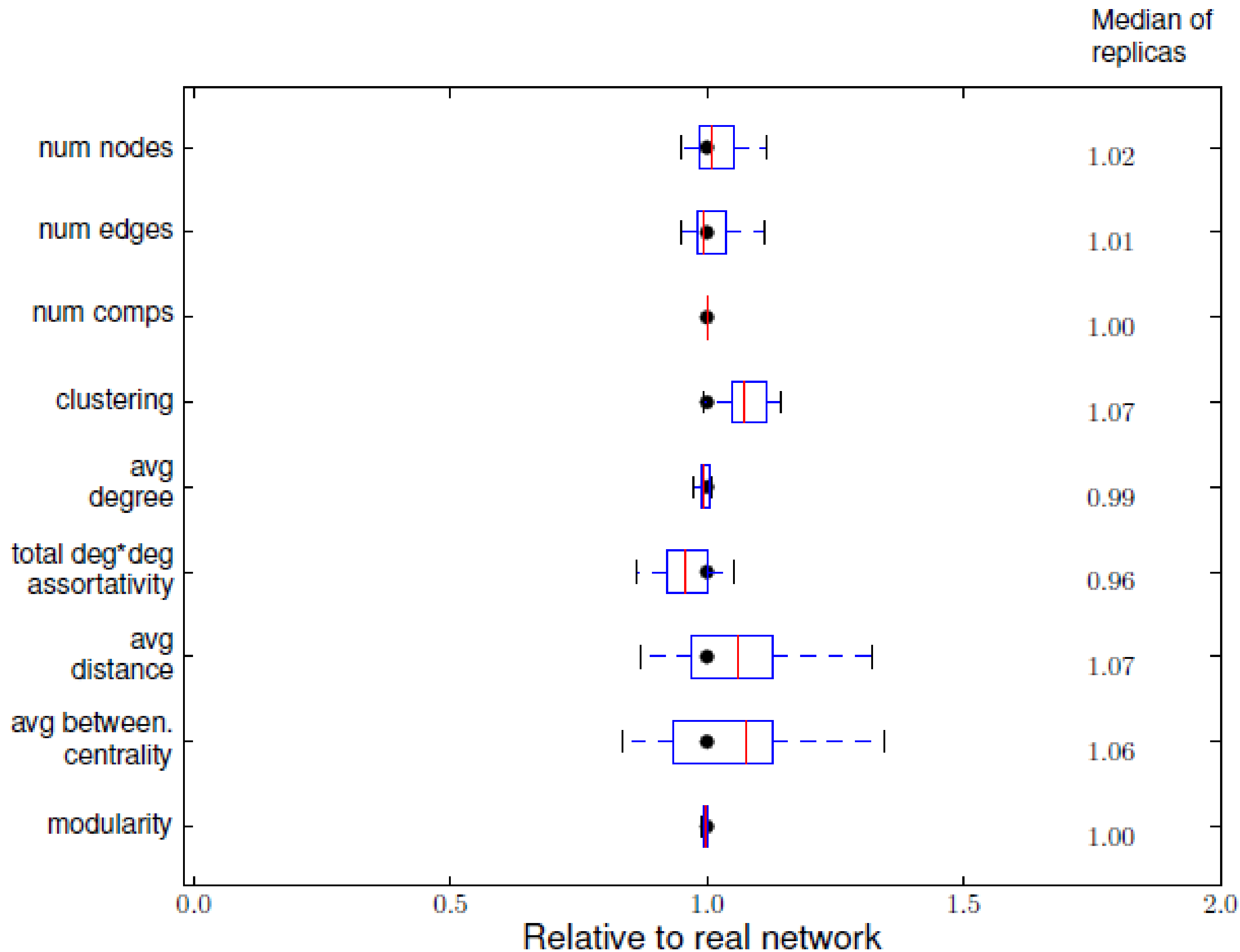
Toy Example: Mesh 33x33 by

Original graph: mesh 33x33	Generation with local changes	Generation with small number of global changes
		
Number of generated nodes is 3 times bigger	Global changes and number of generated nodes is 3 times bigger	Generation with small number of global changes
		

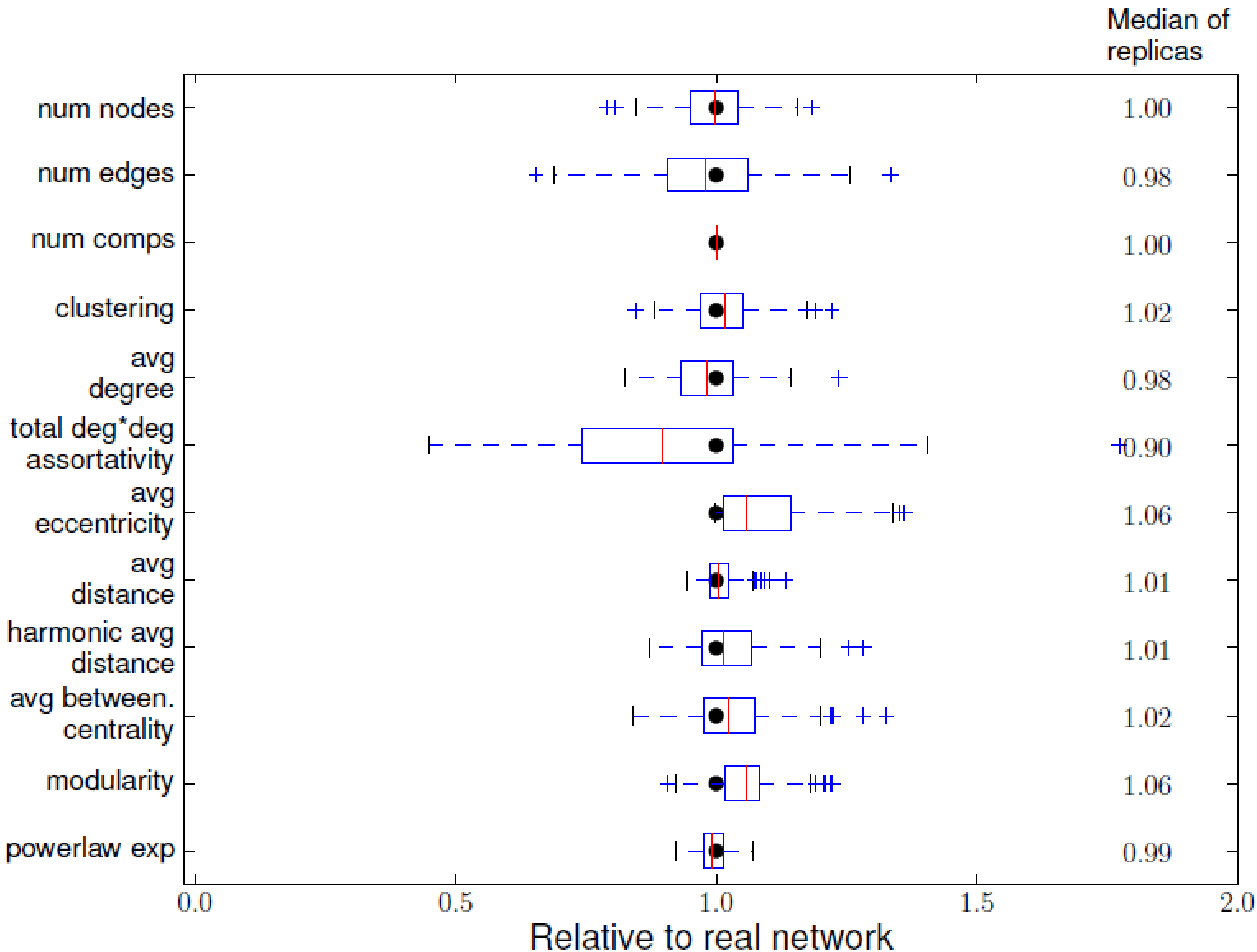
Example: Power Grid by

Original graph: US western states power grid, Watts, Strogatz, Nature, 1998	Generation with local changes	Generation with small number of global changes
		
Number of generated nodes is 3 times bigger	Global changes and number of generated nodes is twice bigger	Generation with small number of global changes
		

Example: Power Grid by

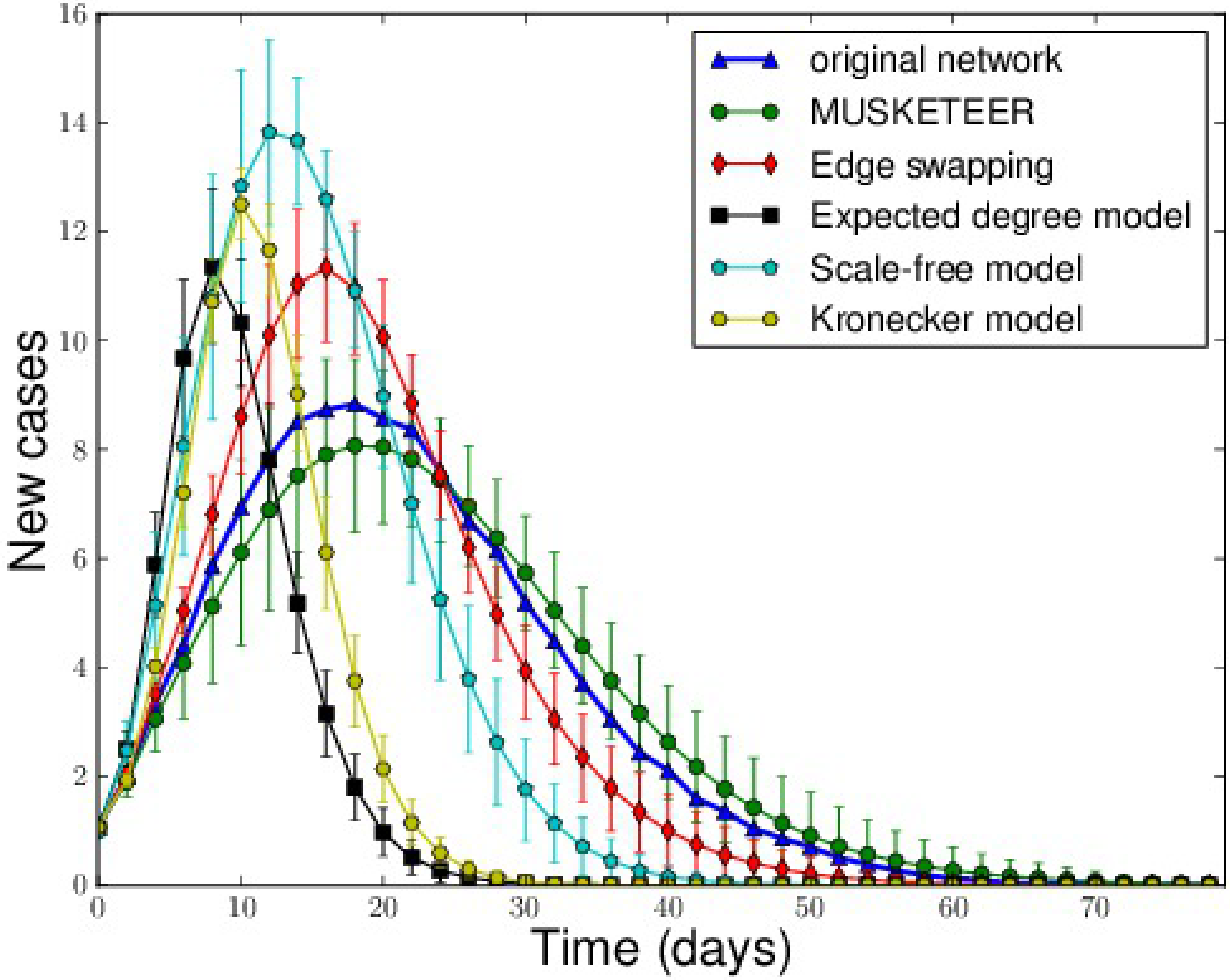


Example: Barabasi-Albert Model by

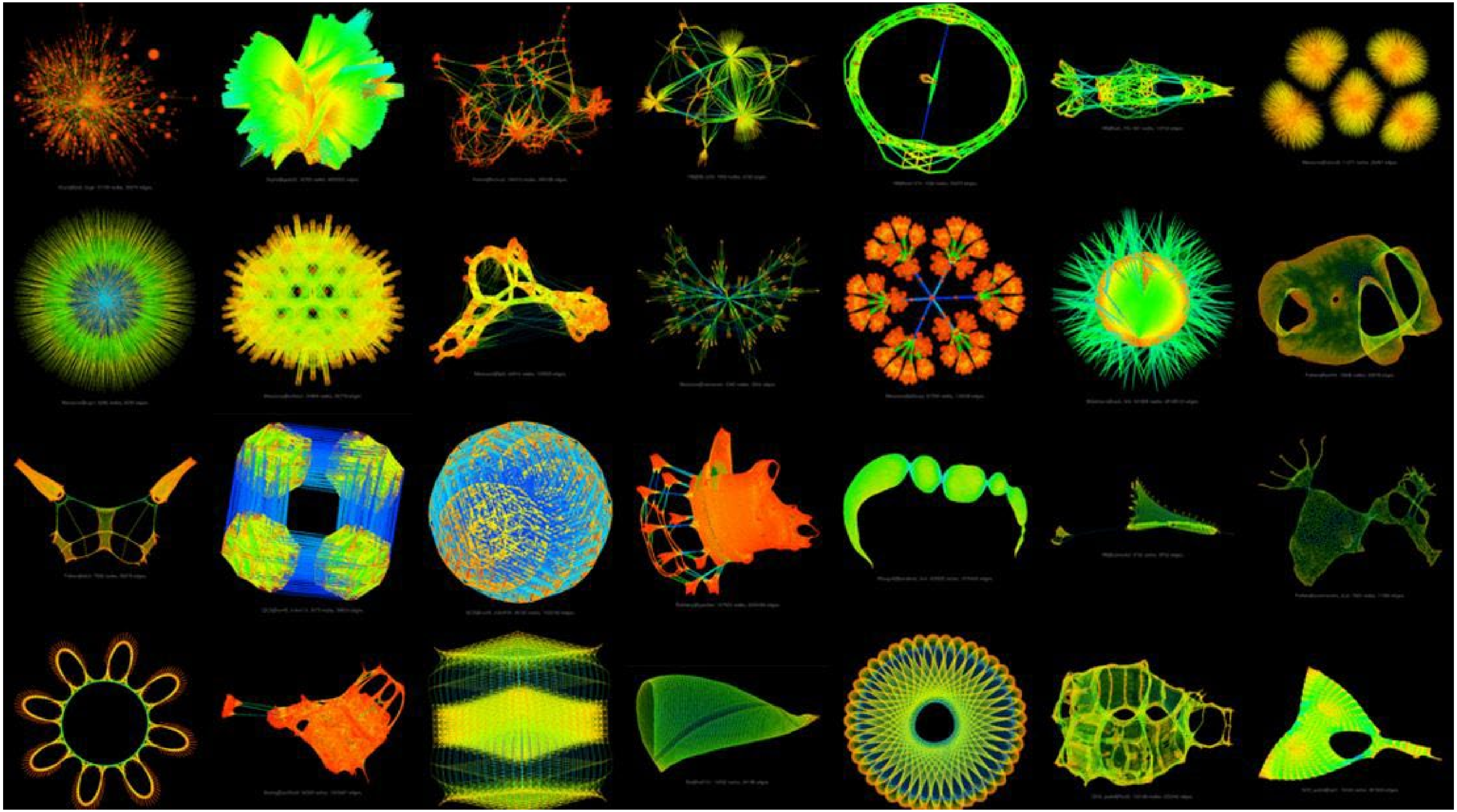


SEIR cascade on Colorado Springs Network

susceptible → exposed → recovered → susceptible

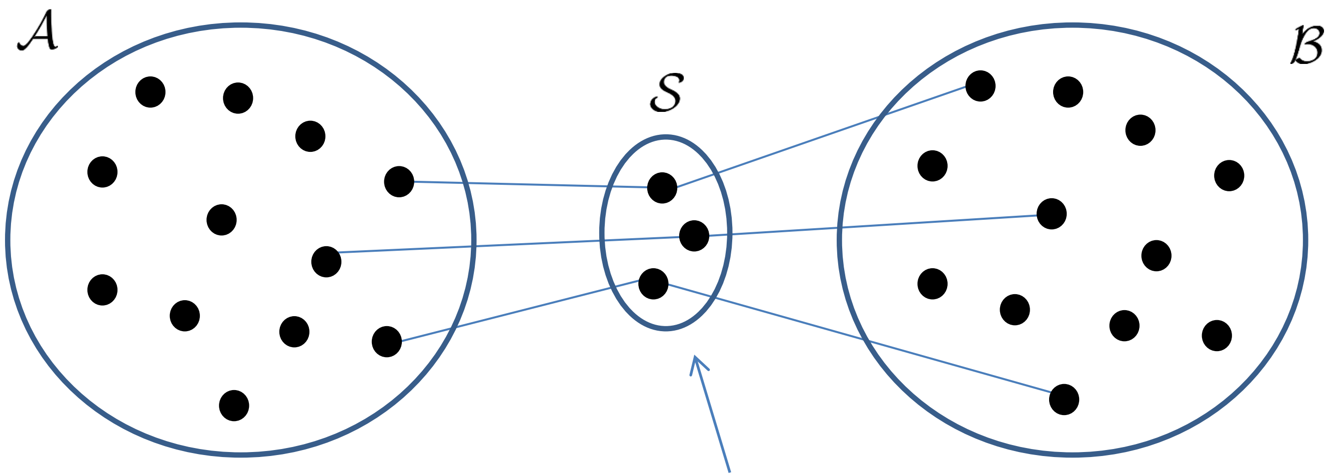


Multilevel Methods for Network Visualization



http://www.cise.ufl.edu/research/sparse/matrices/

Finding Minimum Vertex Separators



$$\min_{\mathcal{A},\mathcal{B}\subset\mathcal{V}} \quad |\mathcal{S}|$$

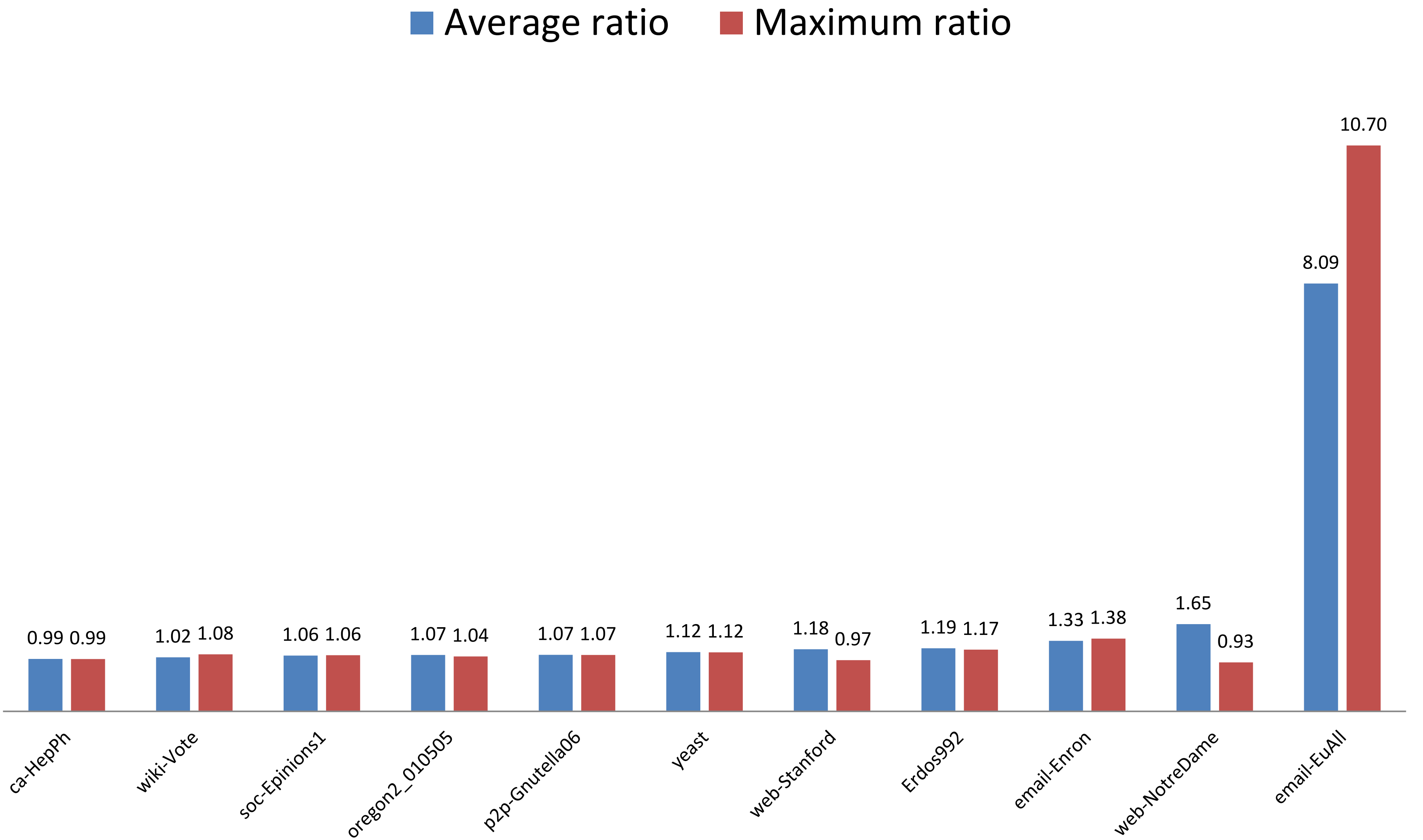
$$\text{subject to} \quad \mathcal{S} = \mathcal{V} \setminus (\mathcal{A} \cup \mathcal{B}), \quad \mathcal{A} \cap \mathcal{B} = \emptyset, \quad (\mathcal{A} \times \mathcal{B}) \cap \mathcal{E} = \emptyset,$$
$$\ell_a \leq |\mathcal{A}| \leq u_a, \quad \text{and} \quad \ell_b \leq |\mathcal{B}| \leq u_b.$$

Bilinear Quadratic Program $$\max_{\mathbf{x},\mathbf{y}\in\mathbb{R}^n} \quad \mathbf{c}^\mathsf{T}(\mathbf{x}+\mathbf{y}) - \gamma\mathbf{x}^\mathsf{T}(\mathbf{A}+\mathbf{I})\mathbf{y}$$

$$\text{subject to} \quad \mathbf{0} \leq \mathbf{x} \leq \mathbf{1}, \quad \mathbf{0} \leq \mathbf{y} \leq \mathbf{1}, \quad \ell_a \leq \mathbf{1}^\mathsf{T}\mathbf{x} \leq u_a, \quad \text{and} \quad \ell_b \leq \mathbf{1}^\mathsf{T}\mathbf{y} \leq u_b.$$

Hager, Hungerford “A Continuous Quadratic Programming Formulation of the Vertex Separator Problem”

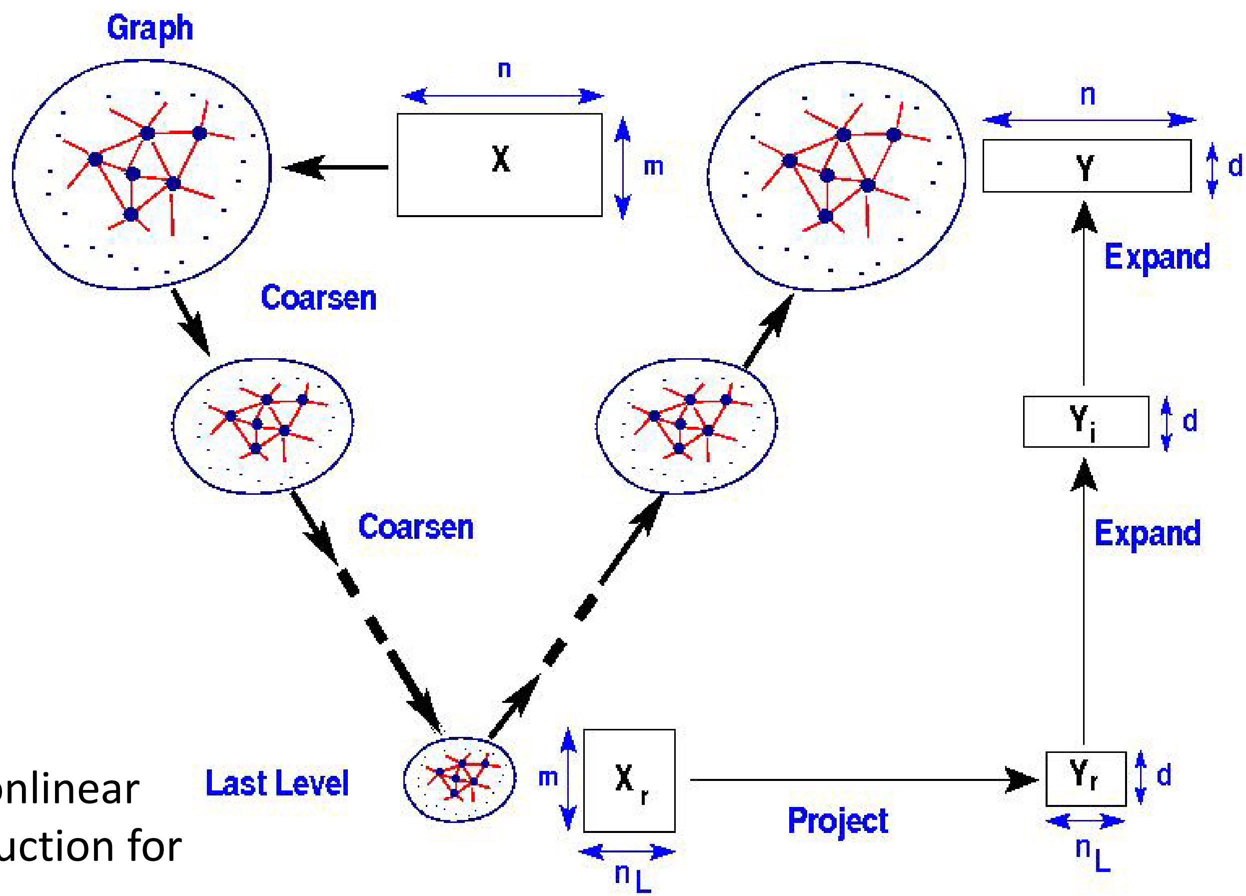
Finding Minimum Vertex Separators in Heavy Tailed Networks



METIS (KL/FM Refinement) vs AMG+Bilinear QP

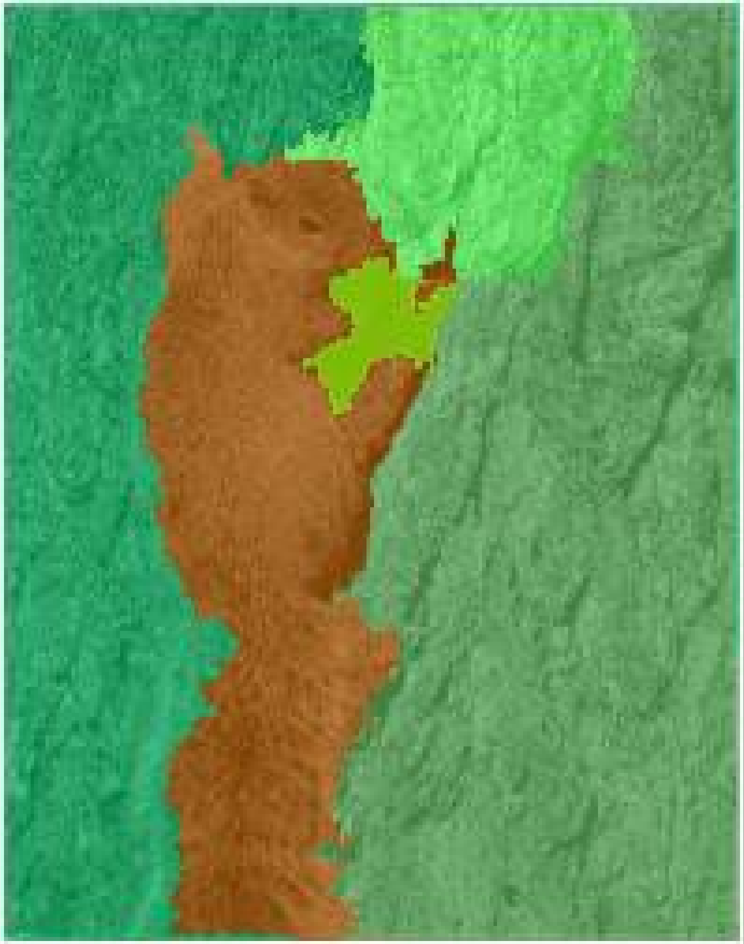
Dimensionality Reduction

Given a set of high dimensional data represented by vectors $x_1, \ldots, x_n$ in R^m, the task is to represent these with low dimensional vectors $y_1, \ldots, y_n \in R^d$ with $d \ll m$, such that nearby points remain nearby, and distant points remain distant.



[FSS] ”Multilevel Nonlinear Dimensionality Reduction for Manifold Learning”

Segmentation



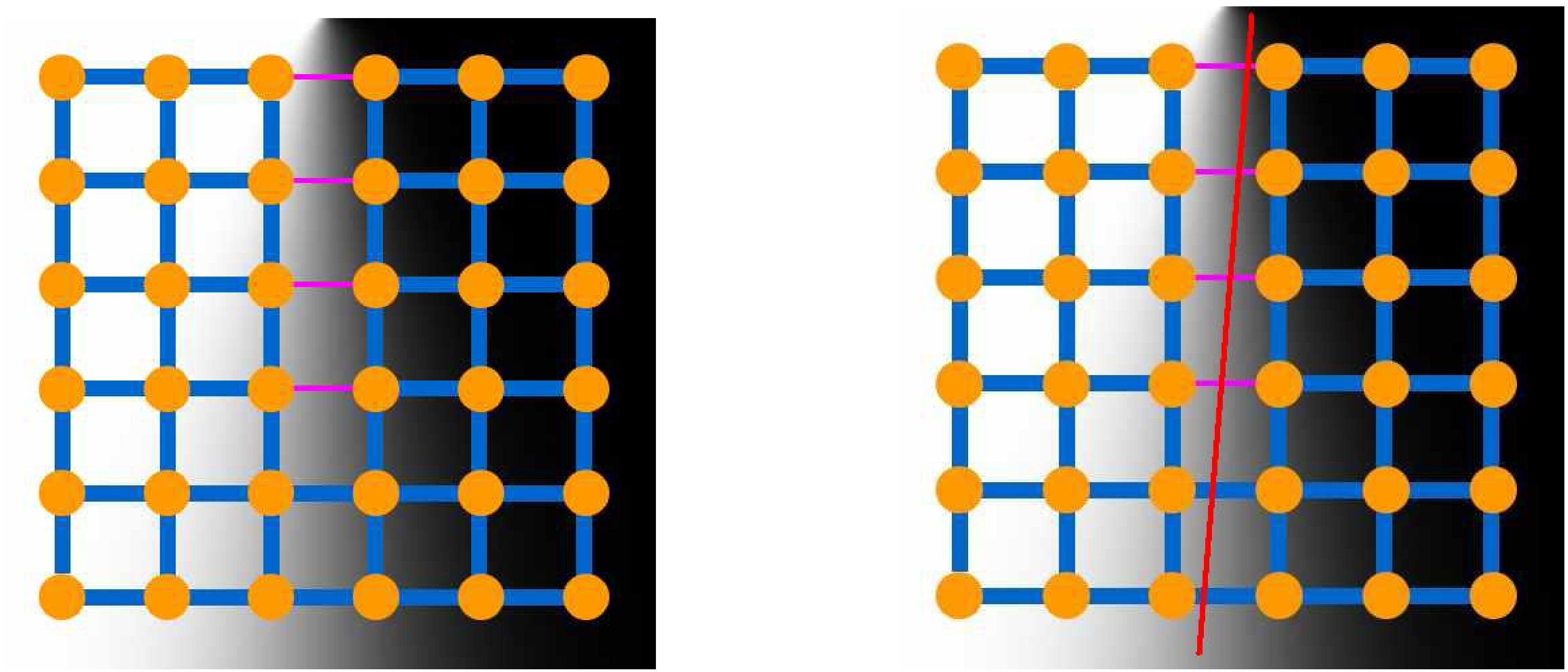
[SGSBB] ”Hierarchy and adaptivity in segmenting visual scenes”, Nature, 2006

Segmentation: The pixel graph

Low contrast - strong coupling, High contrast - weak coupling;
Segmentation $\equiv$ Low-energy cut

$$\text{minimize } \Gamma(u) = \frac{\sum_{i>j} w_{ij}(u_i - u_j)^2}{\sum_{i>j} w_{ij} u_i u_j}$$

Any boolean u that yields a low-energy $\Gamma(u)$ corresponds to a salient segment



Segmentation: Multiscale Approach

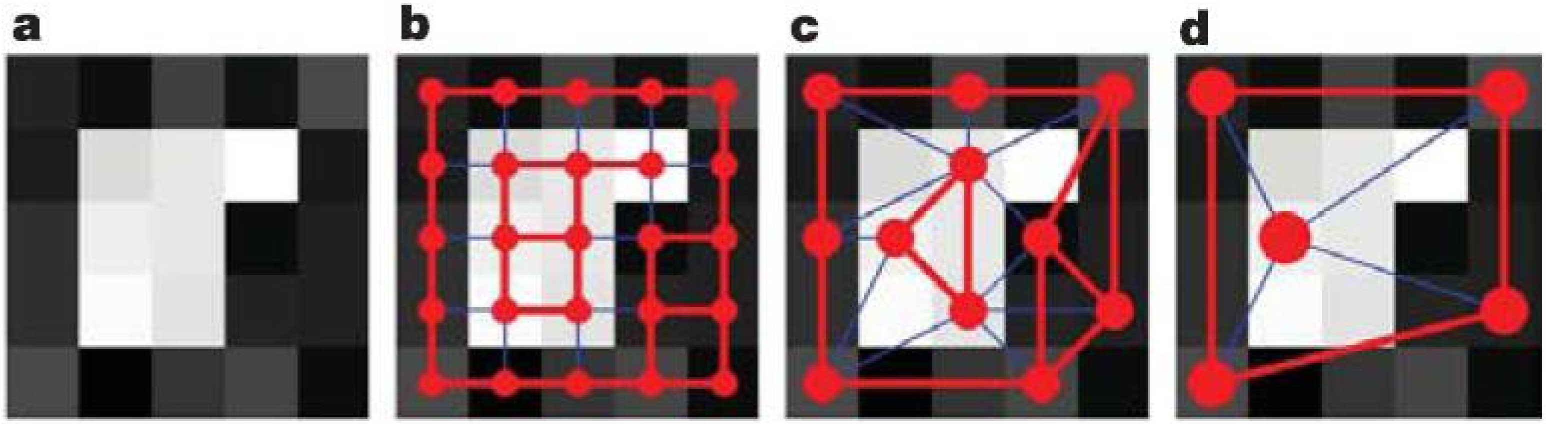
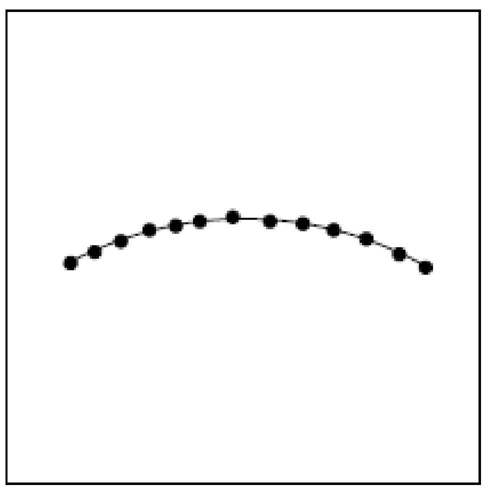


Figure 2 | The multiscale normalized cut graph approach. **a**, A simple image. **b**, Pixels of the image are nodes, represented by filled circles; strong coupling is represented by thick red lines, and weak coupling by thin blue lines. **c**, Adaptive coarsening. Each pixel in **b** is strongly coupled to one of the chosen seeds shown here (thus, pixels strongly coupled to a given seed form an aggregate). Couplings between the seeds are shown. **d**, An additional coarsening level. In this case, this is the level at which the salient segment is detected.

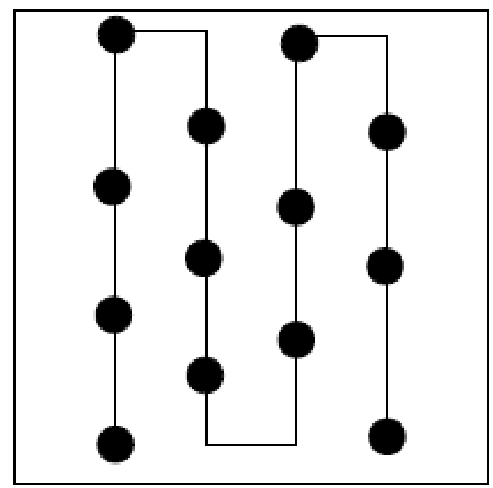
Two-dimensional layout problem

Find an optimal layout of 2D objects such that

1. the total length of the given connections between these objects will be minimal
2. the two-dimensional space will be well utilized and
3. the overlapping between objects will be as little as possible



(a)

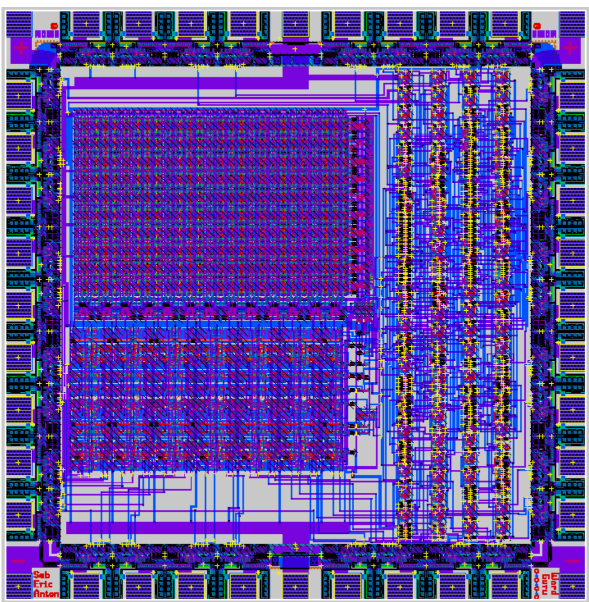


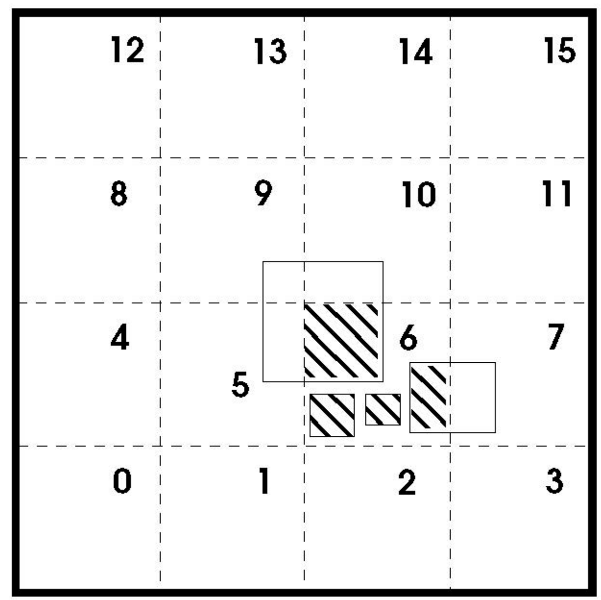
(b)

Two-dimensional layout problem

minimize Total edge length (*quadratic functional*)

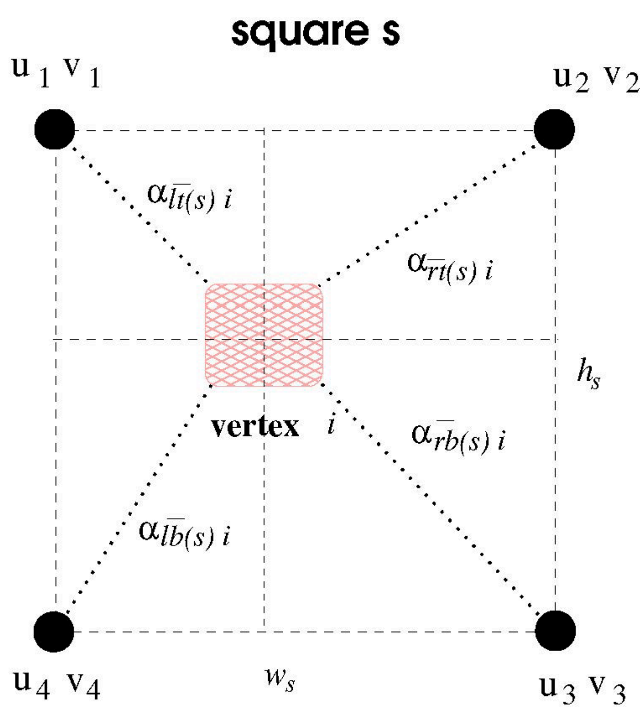
subject to ∀ small squares $\mathfrak{s}$ the amount of the material inside $\mathfrak{s}$ is less than its area (*linear inequality constraints*).

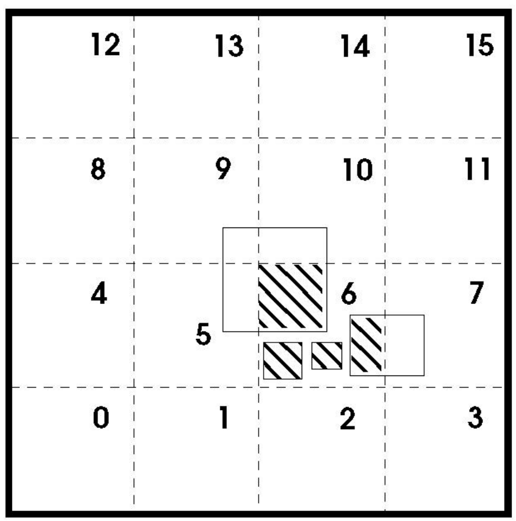




Material movement problem

$$\min_{u,v} \frac{1}{2} \sum_{ij \in E} w_{ij} \Bigg[\Bigg(\tilde{x}_i + \sum_{p \in c(i)} \alpha_{pi} u_p - \tilde{x}_j - \sum_{p \in c(j)} \alpha_{pj} u_p \Bigg)^2 + \Bigg(\tilde{y}_i + \sum_{p \in c(i)} \alpha_{pi} v_p - \tilde{y}_j - \sum_{p \in c(j)} \alpha_{pj} v_p \Bigg)^2 \Bigg]$$





$$\forall s,\ \mathfrak{eqd}(s) =$$

$$\frac{\Upsilon(s)+\Upsilon_r(s)}{2\mathcal{A}} h_y \frac{u_{\mathrm{rt}}(s)+u_{\mathrm{rb}}(s)}{2} - \frac{\Upsilon(s)+\Upsilon_l(s)}{2\mathcal{A}} h_y \frac{u_{\mathrm{lt}}(s)+u_{\mathrm{lb}}(s)}{2} +$$

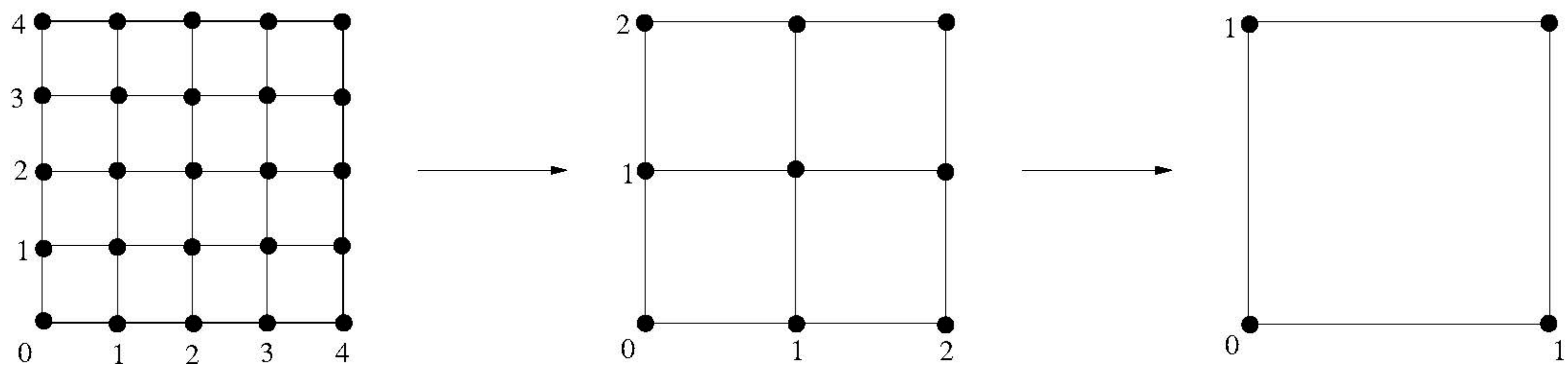
$$\frac{\Upsilon(s)+\Upsilon_t(s)}{2\mathcal{A}} h_x \frac{v_{\mathrm{rt}}(s)+v_{\mathrm{lt}}(s)}{2} - \frac{\Upsilon(s)+\Upsilon_b(s)}{2\mathcal{A}} h_x \frac{v_{\mathrm{rb}}(s)+v_{\mathrm{lb}}(s)}{2}$$

$$\leq M(s) - \Upsilon(s)$$

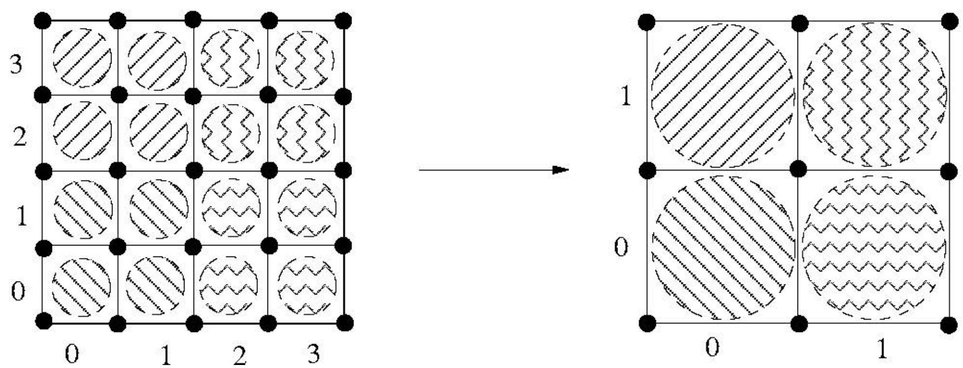
$\Upsilon(s)$	total area of nodes overlapping with s
h_x and h_y	width and height of s
$\mathcal{A}$	area of s
$\Upsilon_r(s)$	area of nodes overlapping with right neighbor square
$\Upsilon_l(s)$	...
$\Upsilon_t(s)$	...
$\Upsilon_b(s)$	...

Two-dimensional layout problem: coarsening

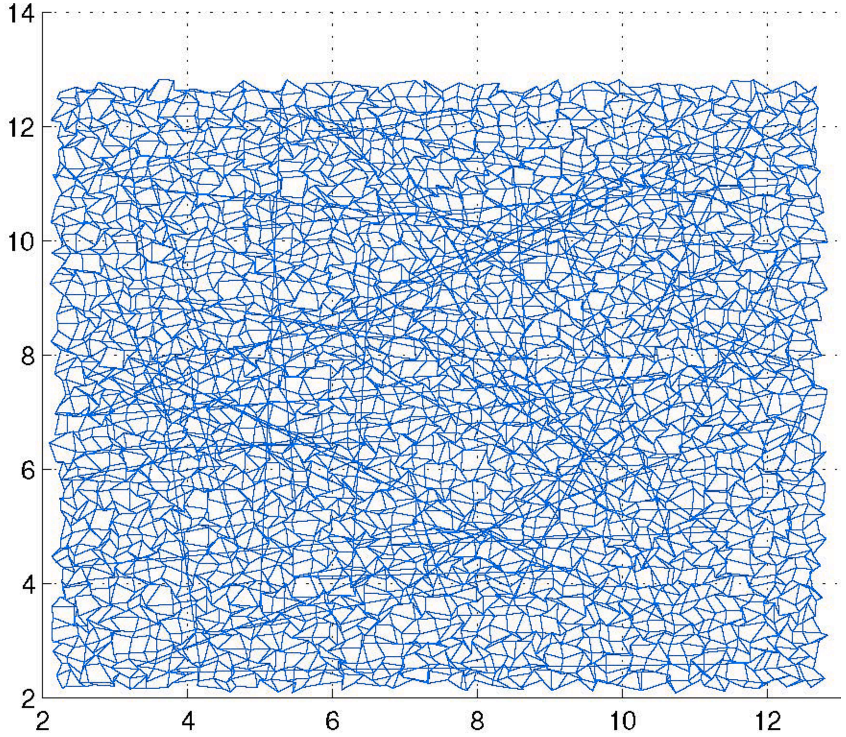
Variables

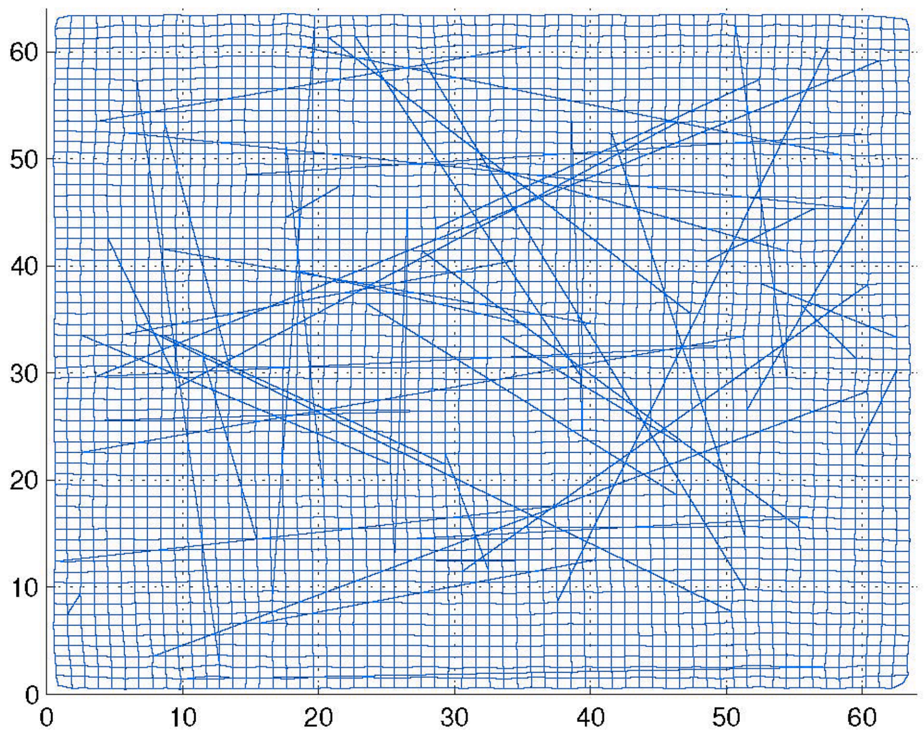


Constraints



Two-dimensional layout problem: example
Mesh 64x64 + Random Edges





Two-dimensional layout problem: VLSI Chip

Original

Multiscale Solver
By Brandt, Ron

- Ron, S, Brandt “Relaxation-based Coarsening and Multiscale Graph Organization”, 2011

- Chen, S “Algebraic Distance on Graphs”, 2011

- Leyffer, S “Fast Response to Infection Spread and Cyber Attacks on Large-scale Networks”, 2013

- S, Sanders, Schulz “Advanced Coarsening Schemes for Graph Partitioning”, 2013

- Gutfraind, Meyers, S “Multiscale Network Generator”, 2013
  http://www.cs.clemson.edu/~isafro/musketeer
  (**can be used to generate networks for your tests!**)

Surveys

- Brandt, Ron “Multigrid Solvers and Multilevel Optimization Strategies”, 2003
- Walshaw “Multilevel Refinement for Combinatorial Optimization”, 2008
- Buluc, Meyerhenke, S, Sanders, Schulz “Recent Advances in Graph Partitioning”, 2013
- Bartel et al. “An Experimental Evaluation of Multilevel Layout Methods”, 2011

The Minimum Workbound Problem

Goal: minimize over all π

$$wb(G,\pi) = \sum_i \max_{\substack{j \\ \pi(j)<\pi(i)}} w_{ij}(\pi(i)-\pi(j))^2$$

Generalization:

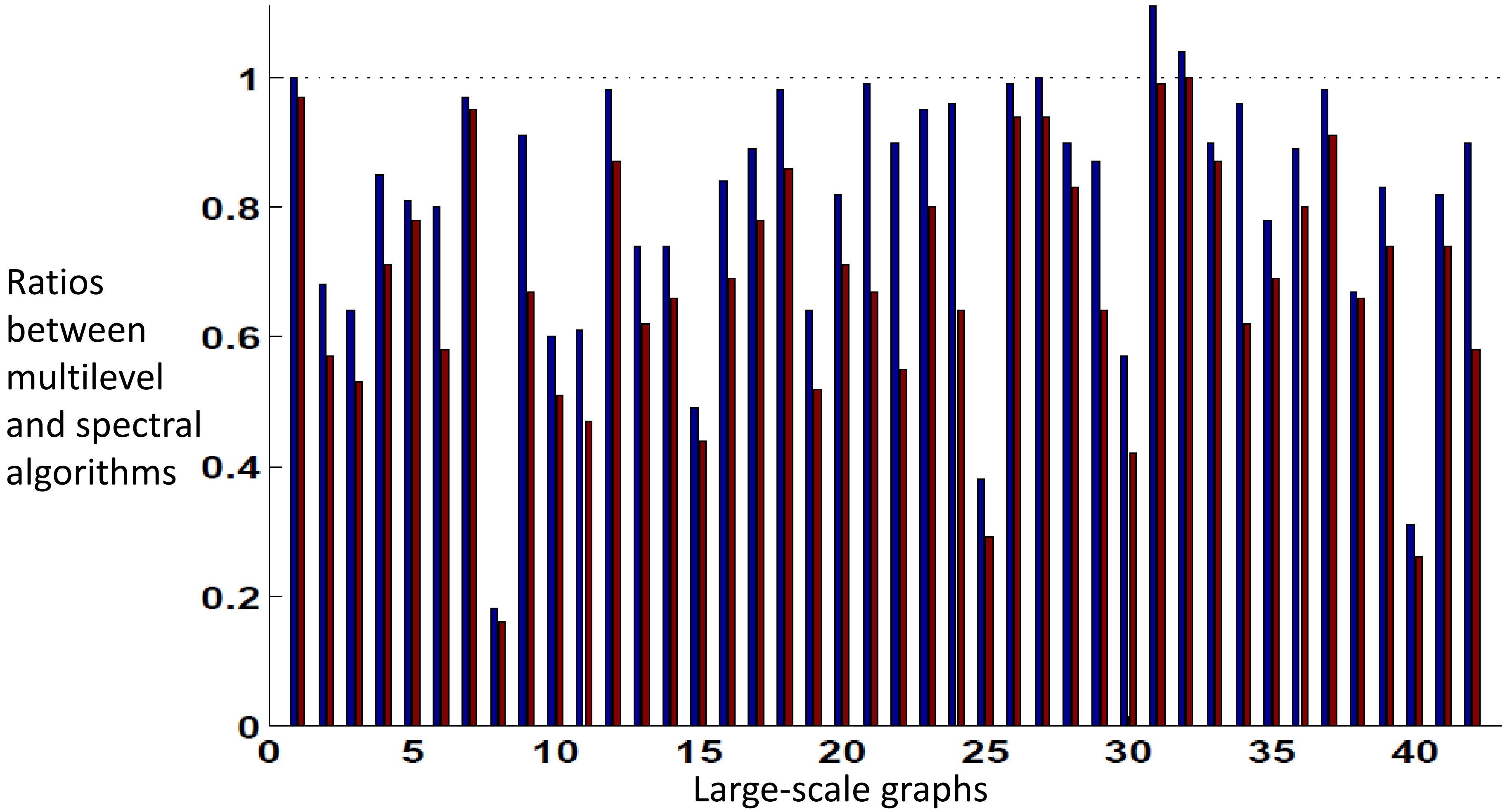
$$wb(G,x) = \sum_i \max_{j:x_j<x_i} w_{ij}(x_i-x_j)^2 \approx \sum_i \Big(\sum_{j:x_j<x_i} w_{ij}(x_i-x_j)^p \Big)^{2/p}$$

Window Minimization for the minimum workbound problem (Taylor exp.):

$$wb_p(W,\tilde{x},\delta) \approx wb_p(W,\tilde{x},\underline{0}) + \sum_{i\in W} \frac{\partial wb_p}{\partial \delta_i}(W,\tilde{x},\underline{0})\delta_i + \sum_{i,j\in W} \frac{\partial^2 wb_p}{\partial \delta_i \partial \delta_j}(W,\tilde{x},\underline{0})\delta_i\delta_j$$

Experimental Results: Minimum Workbound

[SRB] "Multilevel algorithms for linear ordering problems", 2008



Susceptible-Infected-Susceptible Model

The Kephart-White SIS model parameters:
S - number of susceptible nodes; I - number of infected nodes;
β - infection transmission rate; δ - rate of recovery from infection.

$$\begin{cases} \frac{dI}{dt} = \lambda S - \delta I \\ \frac{dS}{dt} = \delta I - \lambda S. \end{cases}$$

Chakrabarti et al. proposed a dynamical system of SIS

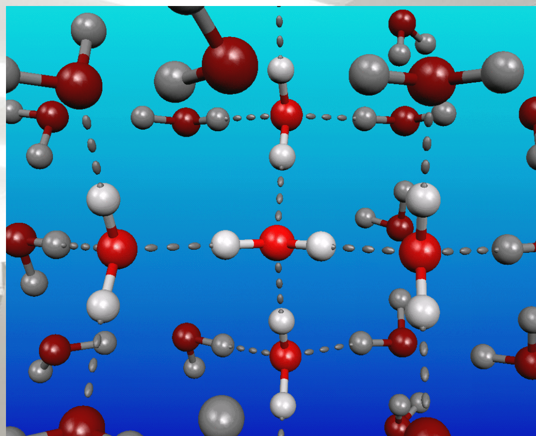
$$1 - \phi_{i,t} = (1 - \phi_{i,t-1}) h_{i,t} + \delta \phi_{i,t-1} h_{i,t}, \quad i = 1...|V|,$$

to describe the probability of keeping i in S, where

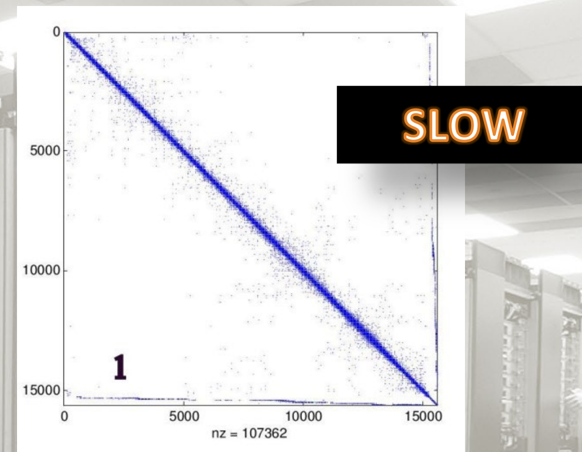
$$h_{i,t} = \prod_{j \in N(i)} (1 - p_{ij} \phi_{j,t-1}).$$

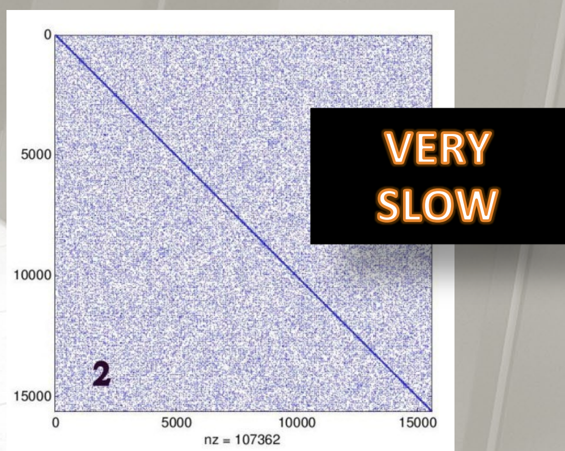
Epidemic threshold τ, a measure to predict when the infection outbreak disappears (comparable to β/δ).

Physical system

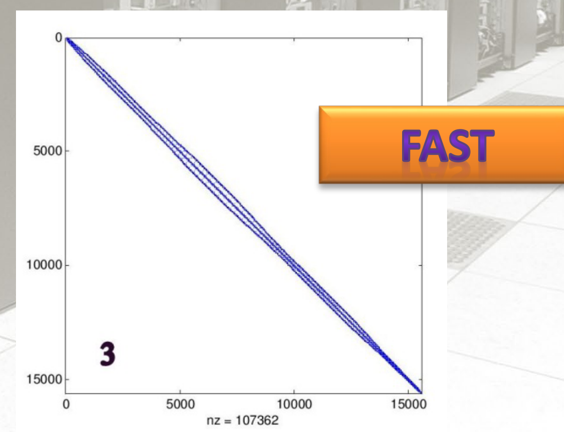


large system of equations
partially disordered

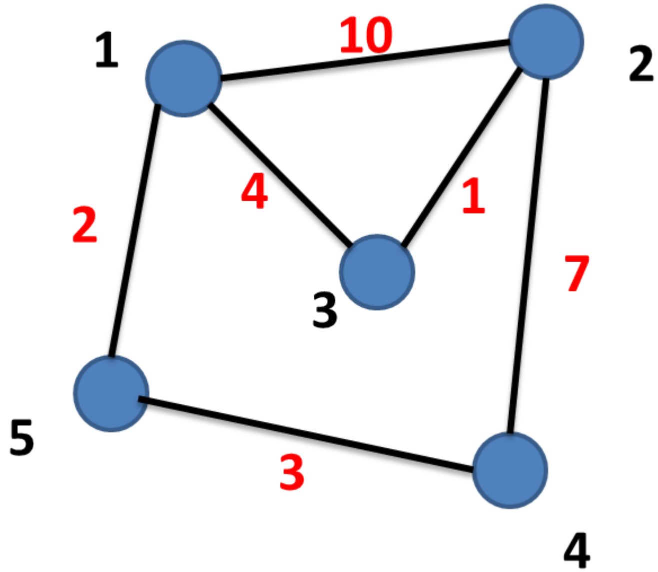




large system of equations
completely disordered



large system of equations
ordered to optimize calculations



Adjacency Matrix A

	1	2	3	4	5
1		10	4		2
2	10		1	7	
3	4	1			
4		7			3
5	2			3	

Laplacian L

	1	2	3	4	5
1	16	-10	-4		-2
2	-10	18	-1	-7	
3	-4	-1	5		
4		-7		10	-3
5	-2			-3	5

real, symmetric matrices

eigenvalues are real

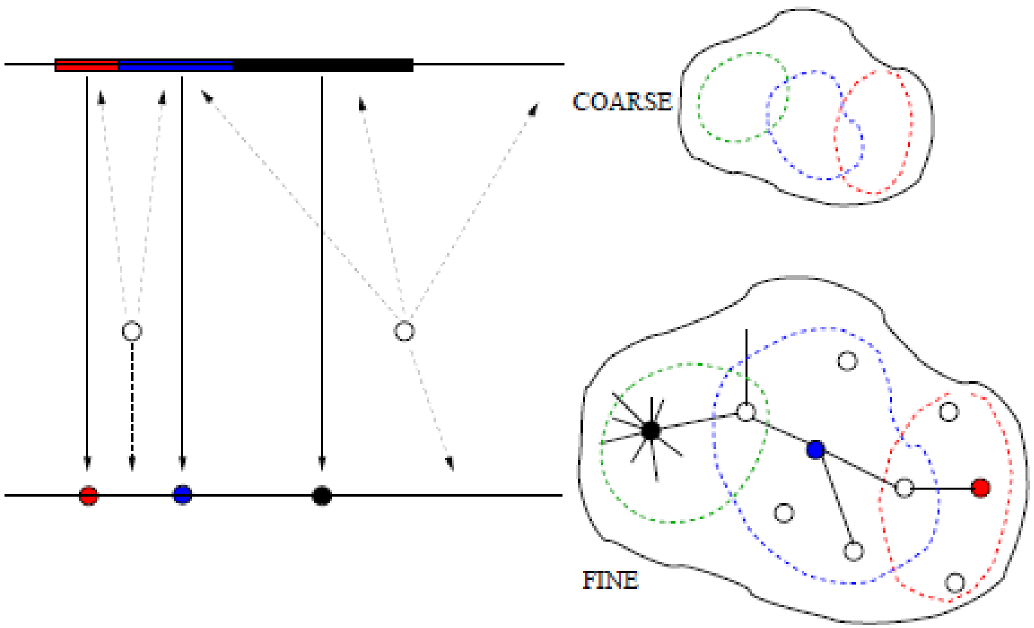
$L_{ij} = -\omega_{ij}$

$L_{ii} = \sum_{ij \in E} \omega_{ij}$

normalized Laplacian $\mathcal{L} = D^{-\frac{1}{2}} \cdot L \cdot D^{-\frac{1}{2}}$

Uncoarsening: Interpolation, Minimum p-sum Problem

1) Place the seeds according to their aggregates



2) Place other vertices by **minimizing** their local contribution to the total energy :

- $p = 1$: at their medians
- $p = 2$: at their weighted averages
- $p > 2$: solve minimization numericaly

Relaxation

Two types of pointwise relaxation that improve current solution:

- Compatible Relaxation: **keep coarse vertices (seeds) invariant** minimizing the energy of other vertices one-by-one wrt to the problem,
- Gauss-Seidel Relaxation: Improve **all** vertices.

Initial legal coordinates $x_i, \ \forall\, i \in V$

for *all* $i \in V$ $y_i \leftarrow x_i$

for *all* $i \in F$ *(Compatible)* / $i \in V$ *(Gauss-Seidel)* **do**

$$y_i = \arg\min_{y_i} \begin{cases} \left| \sum_{y_j < y_i,\ j \in V} w_{ij} - \sum_{y_j > y_i,\ j \in V} w_{ij} \right|, & \text{if } p = 1 \\ \sum_{j \in V} y_j w_{ij} / \sum_{j \in V} w_{ij}, & \text{if } p = 2 \\ \sum_{j \in V} w_{ij} (y_i - y_j)^p, & \text{if } p > 2 \end{cases}$$

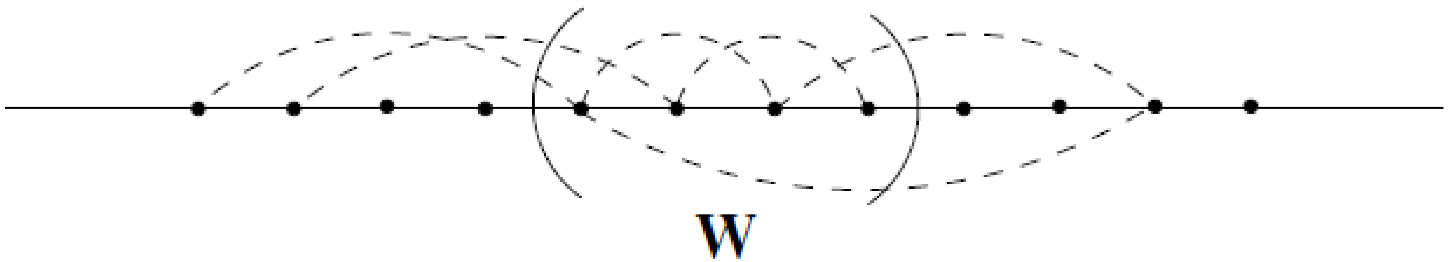
end

for all $i \in V$ $x_i = \frac{v_i}{2} + \sum_{y_k < y_i} v_k$

Uncoarsening: Local Refinement, *p=2*

Lemma: *Improving the ordering cost of W (a subset of consecutive vertices) cannot increase the cost of total ordering.*

Window minimization



$$\text{minimize } \sigma_2(W,\tilde{x},\delta) = \sum_{i,j\in W} w_{ij}(\tilde{x}_i+\delta_i-\tilde{x}_j-\delta_j)^2 + \sum_{\substack{i\in W \\ j\notin W}} w_{ij}(\tilde{x}_i+\delta_i-\tilde{x}_j)^2$$

- $\tilde{x}$ - current approximation
- δ - correction

Uncoarsening: Local Refinement, *p=2*

$$\text{minimize } \sigma_2(W, \tilde{x}, \delta) = \sum_{i,j \in W} w_{ij}(\tilde{x}_i + \delta_i - \tilde{x}_j - \delta_j)^2 + \sum_{\substack{i \in W \\ j \notin W}} w_{ij}(\tilde{x}_i + \delta_i - \tilde{x}_j)^2$$

To prevent the possible convergence of many coordinates to one point add

$$\sum_{i \in \mathfrak{W}} (\tilde{x}_i + \delta_i)^m v_i = \sum_{i \in \mathfrak{W}} \tilde{x}_i^m v_i \ , \ m = 1, 2$$

Final system of equations

$$\begin{cases} \sum_{j \in \mathfrak{W}} w_{ij}(\delta_i - \delta_j) + \delta_i \sum_{j \notin \mathfrak{W}} w_{ij} + \lambda_1 v_i + \lambda_2 v_i \tilde{x}_i = \sum_j w_{ij}(\delta_i - \delta_j) \\ \sum_i \delta_i v_i = 0 \\ \sum_i \delta_i v_i \tilde{x}_i = 0 \end{cases}$$

Linear Arrangement: Spectral Approach

$$\text{minimize over real } x \quad E(x) = \sum_{i,j} w_{ij}(x_i - x_j)^2$$
$$\text{subject to} \quad \sum_i x_i^2 = 1, \quad \sum_i x_i = 0.$$

$$\Longleftrightarrow$$

$$\text{minimize over real } x \quad E(x) = x^T A x$$
$$\text{subject to} \quad x^T B x = 1, \ \sum_i x_i = 0$$
$$\text{where} \quad a_{ij} = -w_{ij}, \ a_{ii} = \sum_j w_{ij}, \ b_{ij} = \delta_{ij}$$

$$\Longleftrightarrow$$

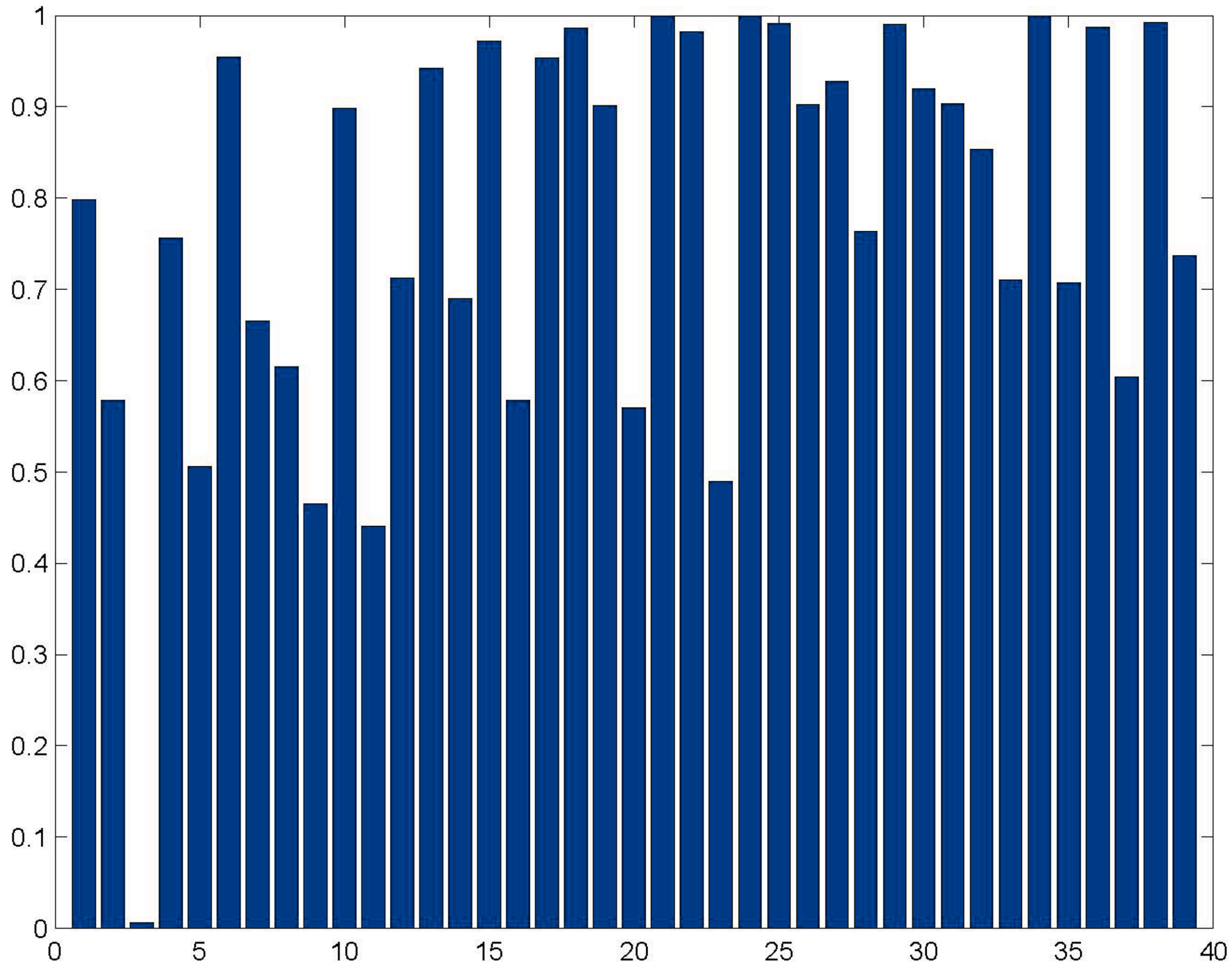
x is the second eigenvector of $\quad Ax = \lambda Bx.$

Heuristics: order the vertices according to the eigenvector of the second smallest eigenvalue.

Experimental Results: Linear Arrangement, *p=2*

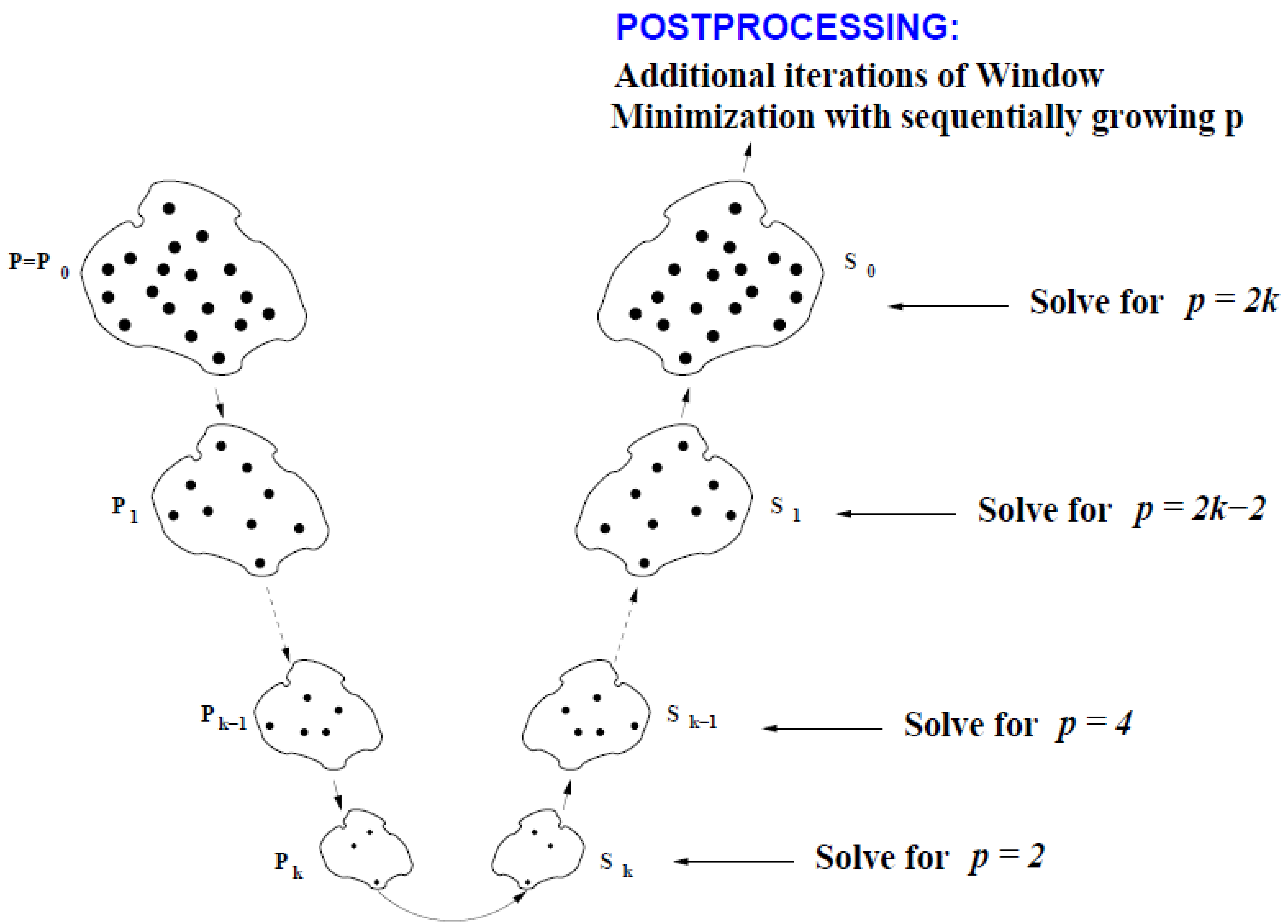
[SRB] "Multilevel algorithm for the minimum 2-sum problem", 2006

Ratios between multilevel and spectral algorithms



Large-scale graphs

Linear Arrangement, Larger powers



Linear Arrangement, Larger powers

- Define $\widehat{w}_{ij} = w_{ij}(\tilde{x}_i - \tilde{x}_j)^{p-2}$
- Substitute w_{ij} with $\widehat{w}_{ij}$ in

minimize $\sigma_p(W, \tilde{x}, \delta) =$

$$= \sum_{i,j \in W} w_{ij}(\tilde{x}_i + \delta_i - \tilde{x}_j - \delta_j)^p + \sum_{\substack{i \in W \\ j \notin W}} w_{ij}(\tilde{x}_i + \delta_i - \tilde{x}_j)^p =$$

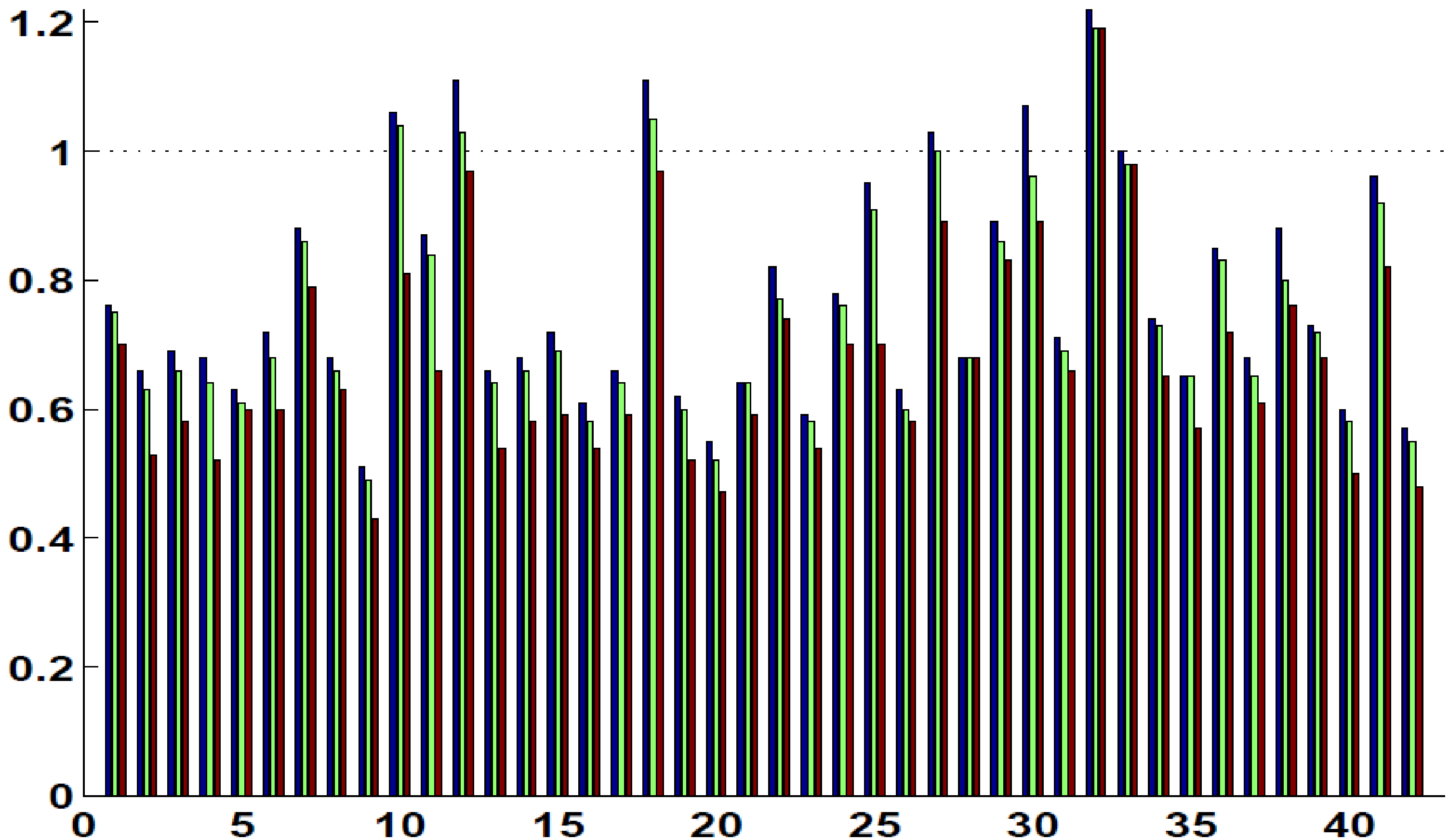
$$= \sum_{i,j \in W} \widehat{w}_{ij}(\tilde{x}_i + \delta_i - \tilde{x}_j - \delta_j)^2 + \sum_{\substack{i \in W \\ j \notin W}} \widehat{w}_{ij}(\tilde{x}_i + \delta_i - \tilde{x}_j)^2 \approx$$

$$\approx \widehat{\sigma}_2(W, \tilde{x}, \delta)$$

Experimental Results: Linear Arrangement, $p=\infty$

[SRB] "Multilevel algorithms for linear ordering problems", 2008

Ratios between multilevel and spectral algorithms



Large-scale graphs