Using multiple breadth-first search to find separators of a graph

Cleve Ashcraft, Roger Grimes

LSTC

`cleve@lstc.com, grimes@lstc.com`

Sparse Days Meeting 2013 at CERFACS

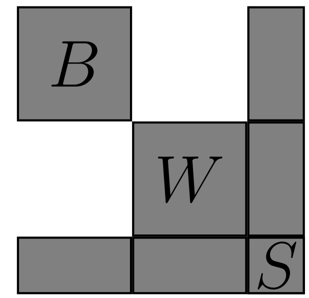
June 17-18, 2013

Toulouse, FRANCE

Goal of the work

$$\mathcal{V} \Longrightarrow \begin{tikzcd} & \mathcal{S} & \\ \mathcal{B} \arrow[ur] & & \mathcal{W} \arrow[ul] \end{tikzcd}$$

$\underline{\mathsf{split}(\mathcal{V})}$
find vertex separator $\mathcal{S}$
$\mathsf{split}(\mathcal{B})$
$\mathsf{split}(\mathcal{W})$



- $\mathcal{S}$ **separates** $\mathcal{B}$ **from** $\mathcal{W}$
- $\mathcal{E} \cap (\mathcal{B} \times \mathcal{W}) = \emptyset$
- $|\mathcal{S}|$ **small,** $|\mathcal{B}| \approx |\mathcal{W}|$ **good balance**
- **cost function, quantify goodness of partition**
- **recursive algorithm, finds low-fill matrix orderings**

Outline of talk

- "Cold Start" to find separator
- Improve the partition
- Expand to "wide" separator
- Multiple breadth first search
- MPP Experiments
- Extensions and related work

Outline of talk

- **"Cold Start" to find separator**
  - **Single source level sets, George & Liu, 1981**
  - **Dual source level sets**
- **Improve the partition**
- **Expand to "wide" separator**
- **Multiple breadth first search**
- **MPP Experiments**
- **Extensions and related work**

Outline of talk

- “Cold Start” to find separator
- Improve the partition
  - Max Flow solver, A. & Liu, 1998
  - “Trimming”, one sided improvement
- Expand to “wide” separator
- Multiple breadth first search
- MPP Experiments
- Extensions and related work

Outline of talk

- **"Cold Start" to find separator**
- **Improve the partition**
- **Expand to "wide" separator**
  - **Expand by levels, A. & Liu, 1998**
  - **"Cutting Corners", selective expansion**
- **Multiple breadth first search**
- **MPP Experiments**
- **Extensions and related work**

Outline of talk

- "Cold Start" to find separator
- Improve the partition
- Expand to "wide" separator
- Multiple breadth first search
  - Find pseudodiameter pair via sequential MBFS
  - Find many pseudodiameter pairs via independent MBFS
- MPP Experiments
- Extensions and related work

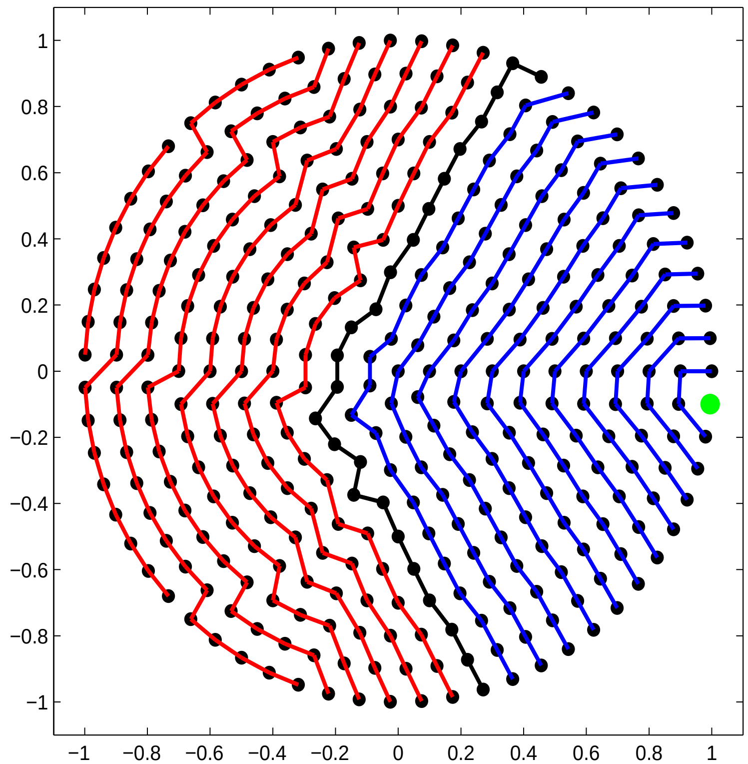
Outline of talk

- **"Cold Start" to find separator**
- **Improve the partition**
- **Expand to "wide" separator**
- **Multiple breadth first search**
- **MPP Experiments**
- **Extensions and related work**
  - **Linelets – preconditioners from CFD**
  - k**-way partitions using MIBFS**
  - **Beyond bisection**

Outline of talk

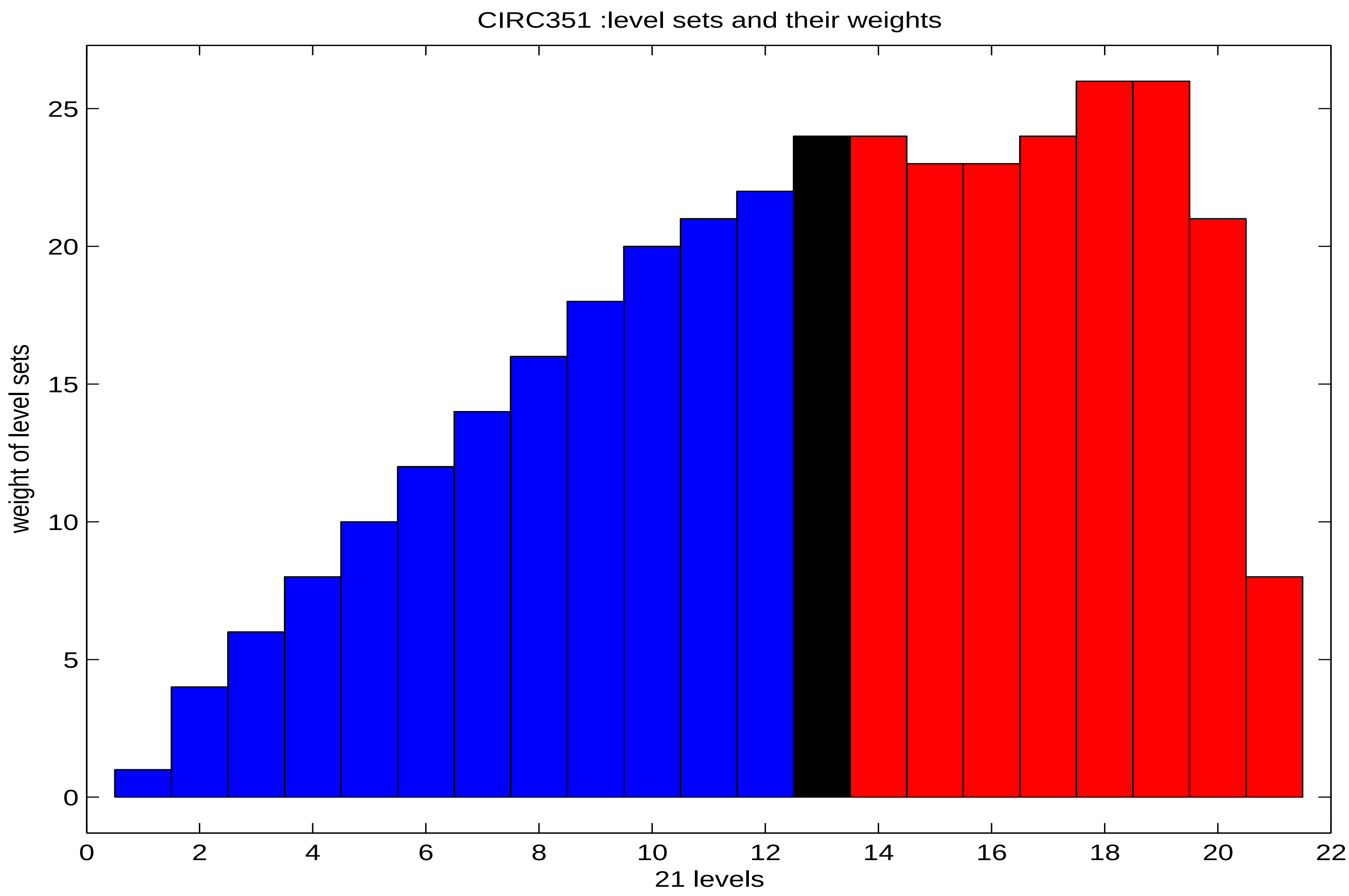
- **“Cold Start” to find separator**
  - **Single source level sets, George & Liu, 1981**
  - **Dual source level sets**
- **Improve the partition**
- **Expand to “wide” separator**
- **Multiple breadth first search**
- **MPP Experiments**
- **Extensions and related work**

Single source level sets $\longrightarrow (\mathcal{B}, \mathcal{S}, \mathcal{W})$ partition

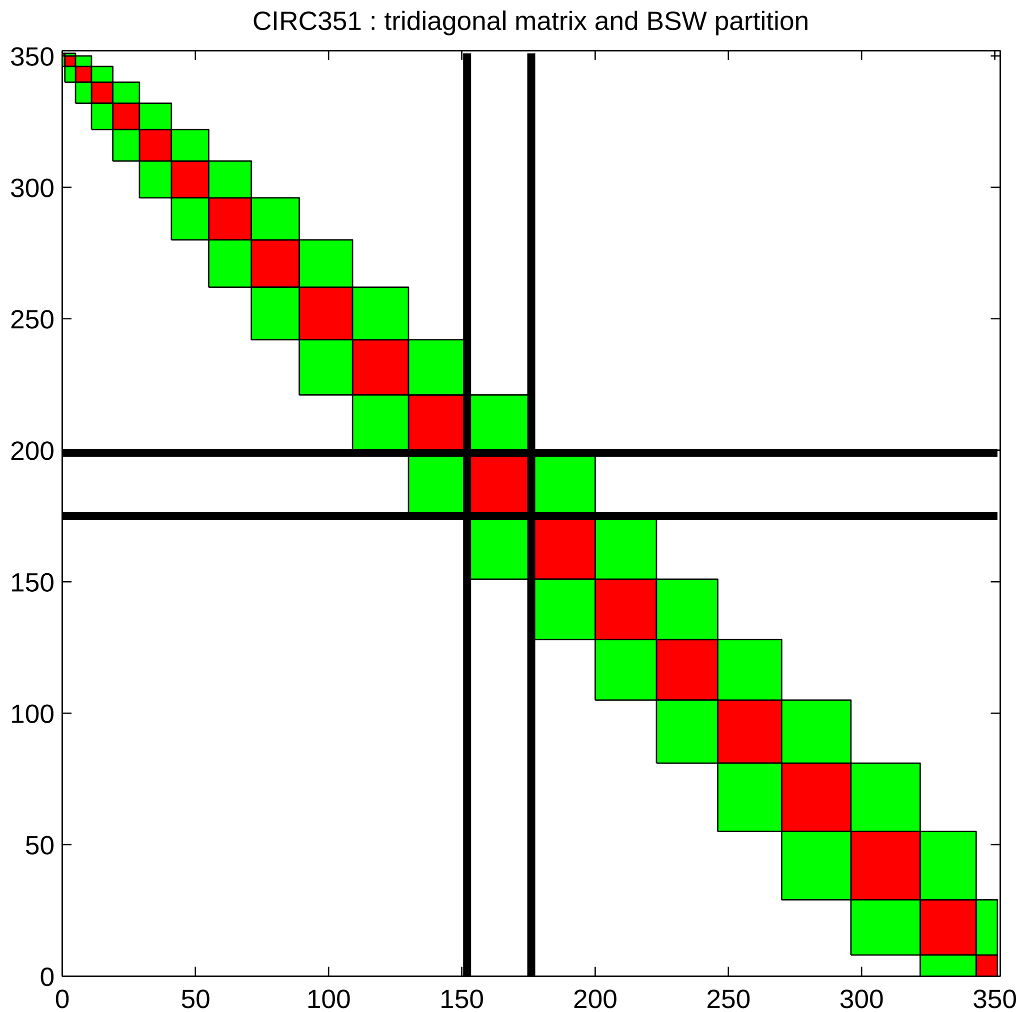


- $\mathsf{level}(u) = \mathsf{dist}(u, s)$
- **source node in green**
- $\mathcal{B}$ **nodes in blue**
- $\mathcal{W}$ **nodes in red**
- $\mathcal{S}$ **nodes in black**
- **nodes connected by level set**
- **separator minimal**

Level weights histogram



Level sets tridiagonal matrix



Cost Function(α, β)

$$\text{diff} = \frac{\text{abs}(|B| - |W|)}{|B| + |S| + |W|}$$

$$\text{imbalance} = \frac{\max(|B|, |W|)}{\min(|B|, |W|)}$$

`if` imbalance $\leq \alpha$

  partition is acceptable

  $\text{cost} = |S|(1 + \beta \cdot \text{diff})$

`else`

  $\text{cost} = \infty$

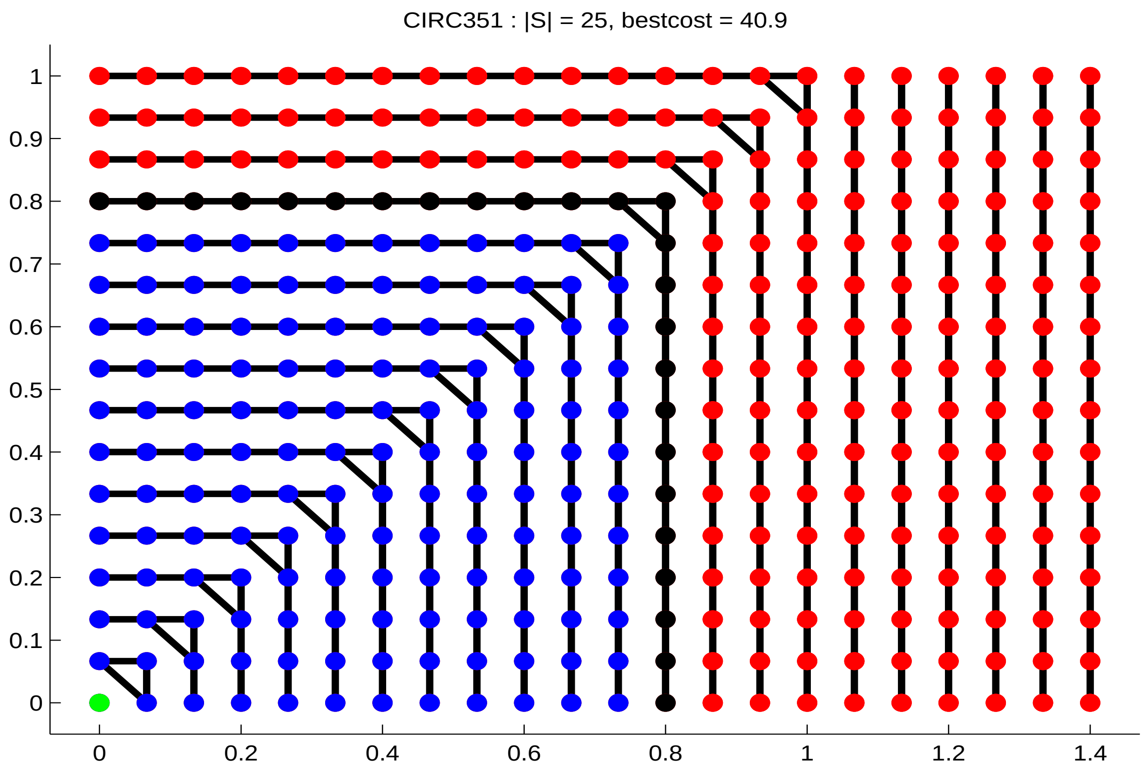
`endif`

GENAND (George and Liu, 1983)

GENAND(V, E)

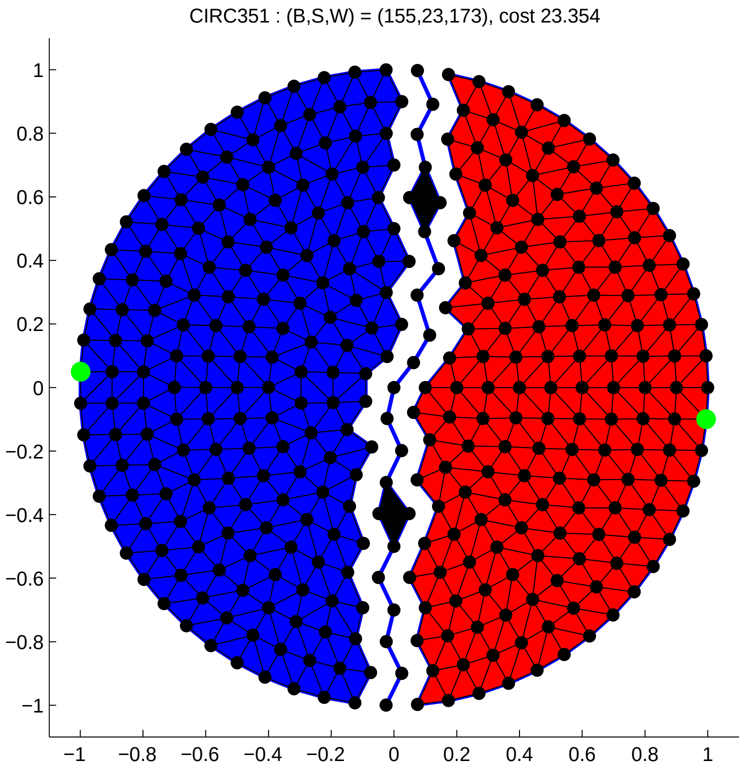
- **Find pseudoperipheral node** s
- **create compressed tridiagonal matrix from the level sets of** s
- **find best** $(\mathcal{B}, \mathcal{S}, \mathcal{W})$ **partition**
- **order** S **last**
- GENAND$(B, E \cap (B \times B))$
- GENAND$(W, E \cap (W \times W))$

Regular grid, 9-point operator, corner starting node



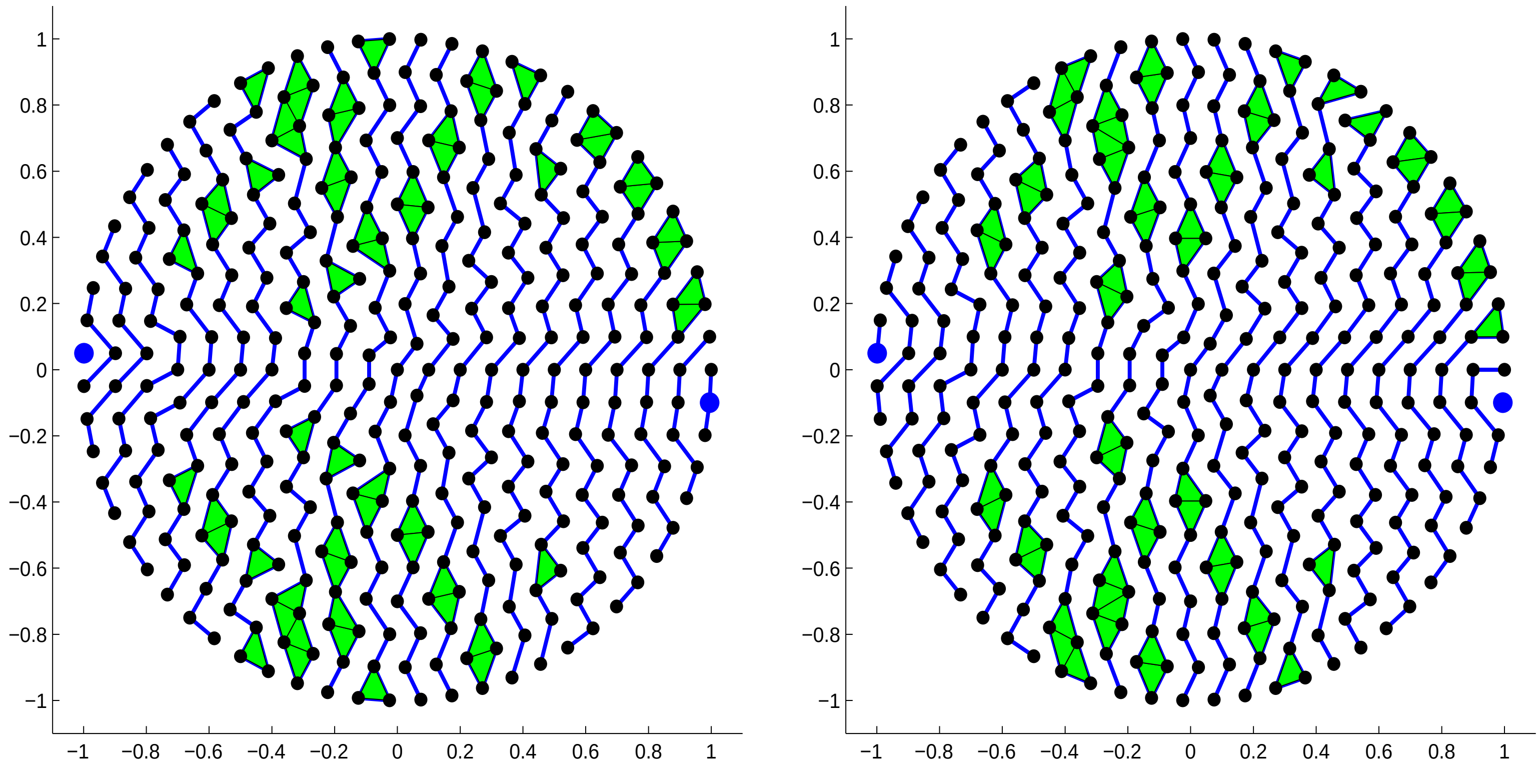
- **$h \times k$ grid**
- **perfect balance,** $\alpha = 1, |S| = \sqrt{2\,h\,k}$
- **in general,** $|S| = \frac{2}{\sqrt{1+\alpha}}\sqrt{h\,k}$
- **relative size** $1 \leq \frac{|S|}{\min(h,k)} \leq 2$

Dual source level half-sets $\longrightarrow (\mathcal{B}, \mathcal{S}, \mathcal{W})$ partition

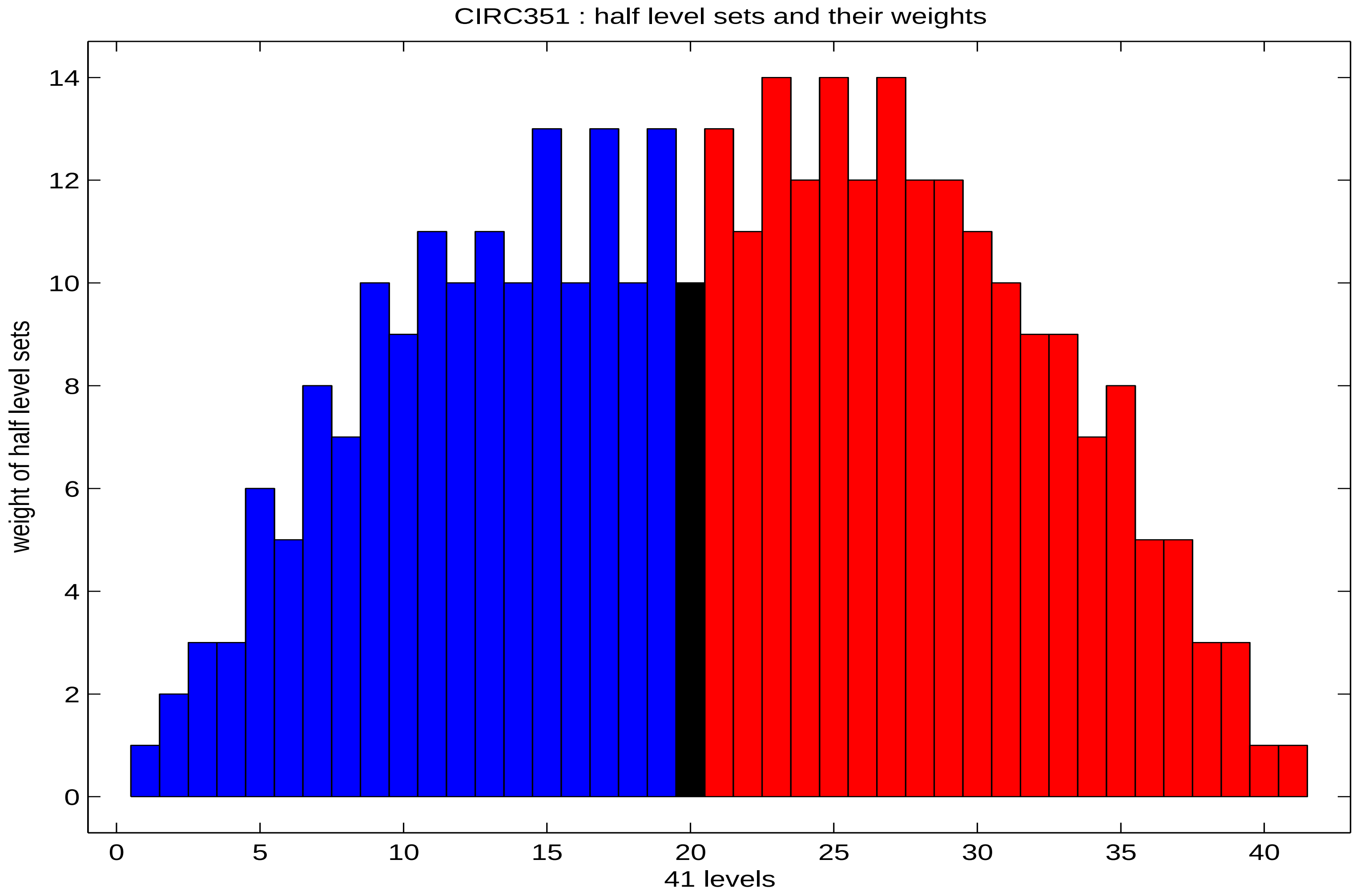


- $\mathsf{level}(u) = \mathsf{dist}(u, s) - \mathsf{dist}(u, t)$
- **two source nodes in green**
- $\mathcal{B}$ **nodes in blue**
- $\mathcal{W}$ **nodes in red**
- $\mathcal{S}$ **nodes in black**
- **adjacent half-sets form a separator**
- **separator NOT minimal**

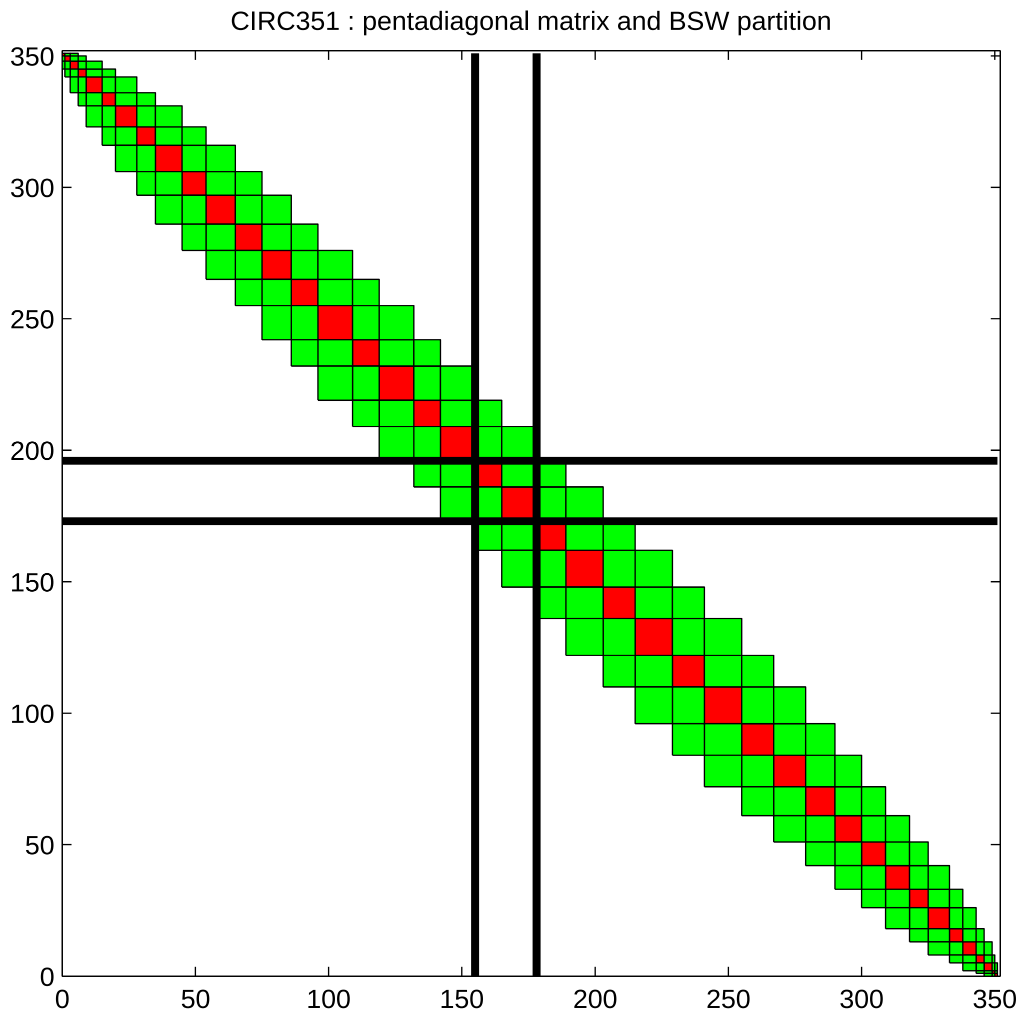
Dual source level half-sets $\longrightarrow (\mathcal{B}, \mathcal{S}, \mathcal{W})$ partition



Level weights histogram



Level half-sets pentadiagonal matrix



Cost Function(α, β)

$$\text{diff} = \frac{\text{abs}(|B| - |W|)}{|B| + |S| + |W|}$$

$$\text{imbalance} = \frac{\max(|B|, |W|)}{\min(|B|, |W|)}$$

`if` imbalance $\leq \alpha$

  partition is acceptable

  $\text{cost} = |S|(1 + \beta \cdot \text{diff})$

`else`

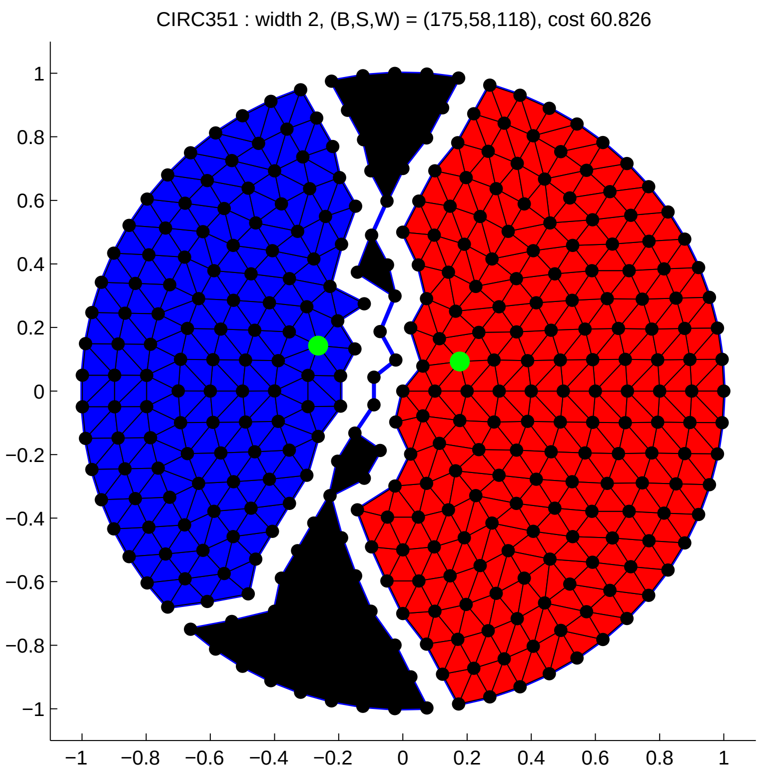
  $\text{cost} = \infty$

`endif`

Outline of talk

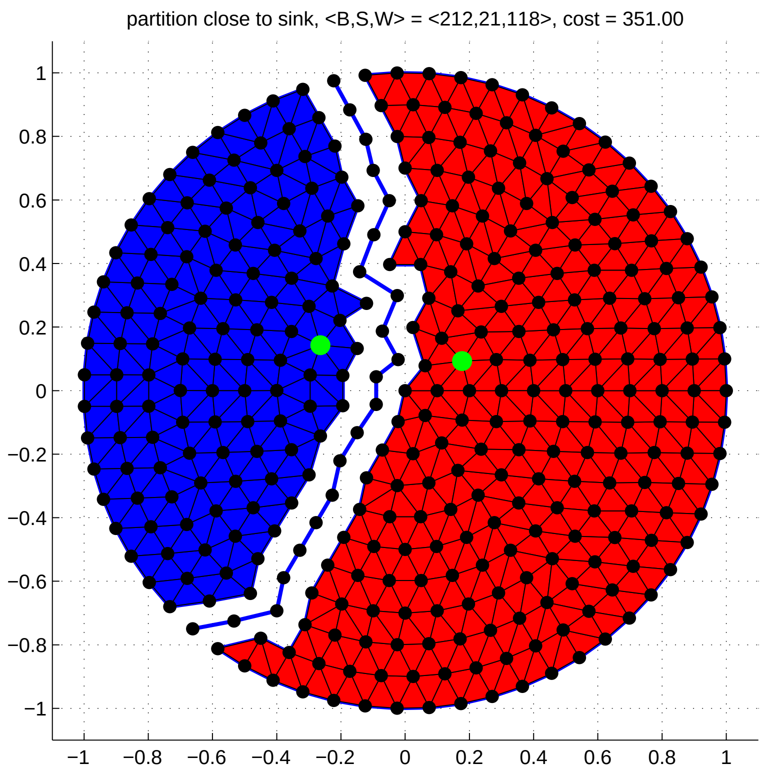
- “Cold Start” to find separator
- **Improve the partition**
  - **Max Flow solver, A. & Liu, 1998**
  - **“Trimming”, one sided improvement**
- Expand to “wide” separator
- Multiple breadth first search
- MPP Experiments
- Extensions and related work

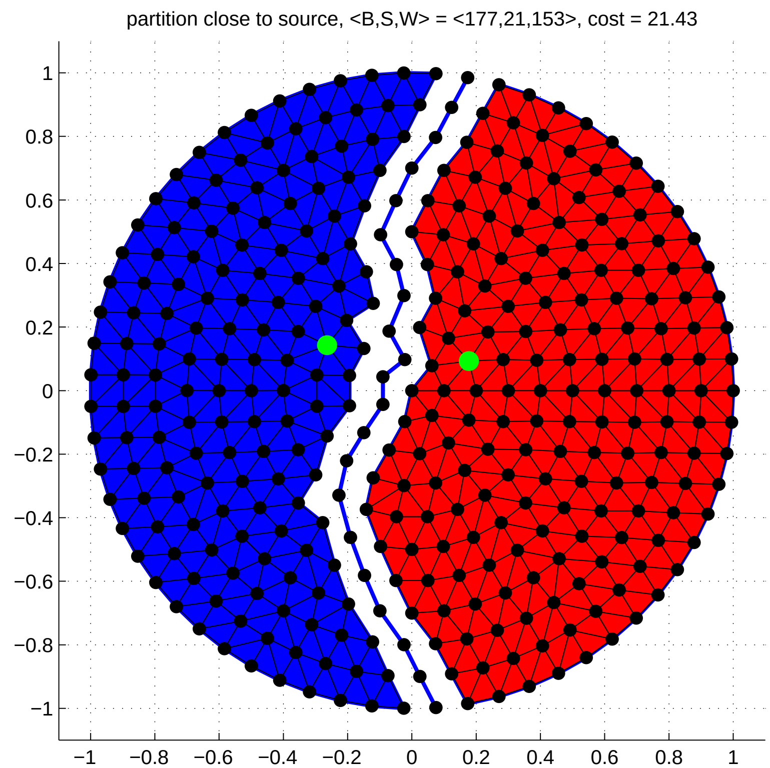
Improve partition $(\mathcal{B}, \mathcal{S}, \mathcal{W})$, using max flow



- **compress** $\mathcal{B}$ **to the source** s
- **compress** $\mathcal{W}$ **to the sink** t
- **expand** $\mathcal{S}$ **into a network of nodes and arcs**
- **solve the max flow problem**
- **A. & Liu, SIMAX 1998**

Network max flow partition improvement





Improve partition $(\mathcal{B}, \mathcal{S}, \mathcal{W})$, via trimming

- **basic idea, choose one of two moves**
- **while still improving**
  - **Partition separator $\mathcal{S}$ into four disjoint sets.** $\mathcal{S} = \mathcal{S}^{00} \sqcup \mathcal{S}^{01} \sqcup \mathcal{S}^{10} \sqcup \mathcal{S}^{11}$

$$\mathcal{S}^{11} = \partial\mathcal{B} \cap \partial\mathcal{W} \quad \textbf{adjacent to both} \tag{1}$$

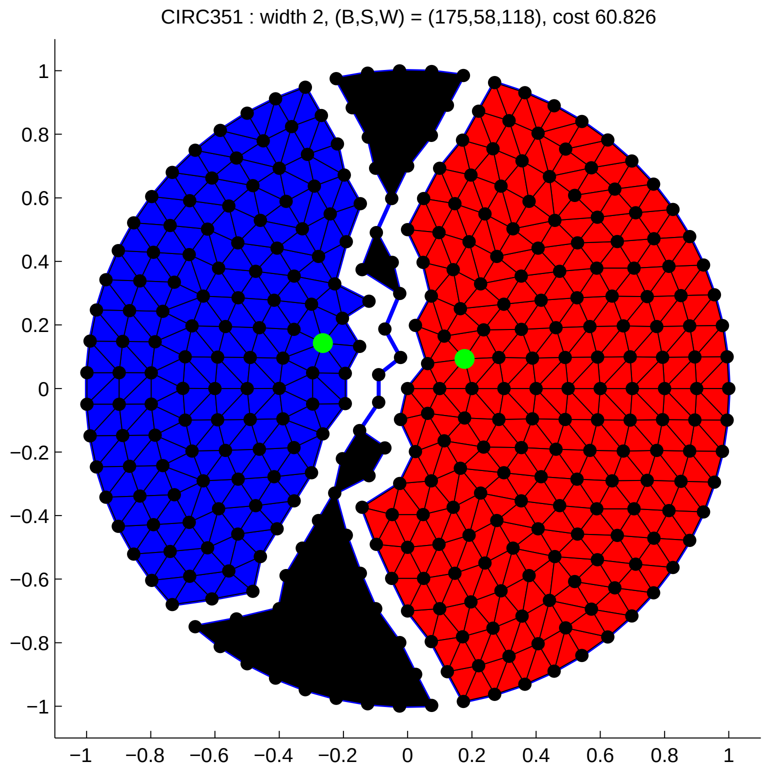
$$\mathcal{S}^{10} = \partial\mathcal{B} \setminus \partial\mathcal{W} \quad \textbf{adjacent to } \mathcal{B}\textbf{, not } \mathcal{W} \tag{2}$$

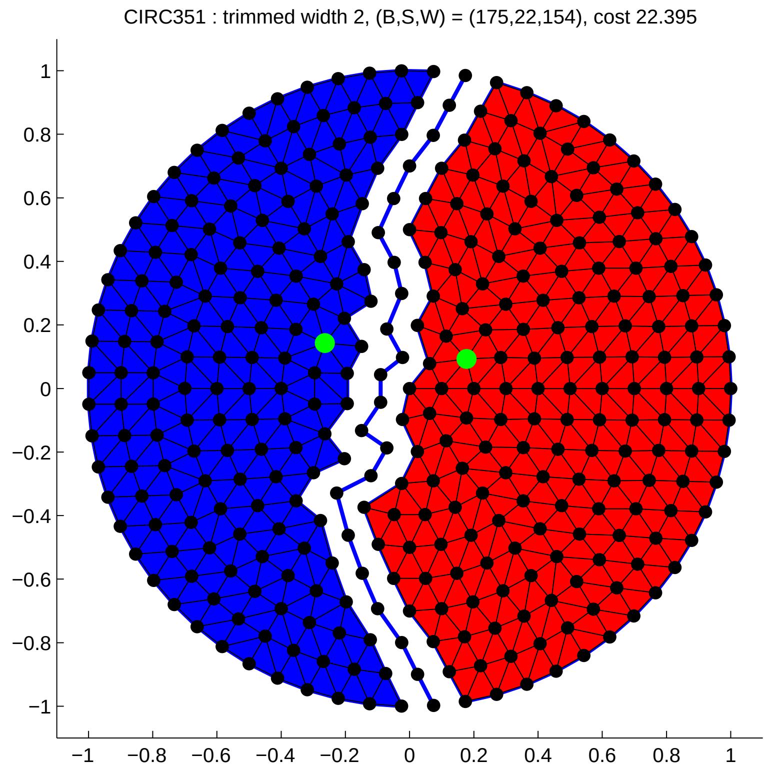
$$\mathcal{S}^{01} = \partial\mathcal{W} \setminus \partial\mathcal{B} \quad \textbf{adjacent to } \mathcal{W}\textbf{, not } \mathcal{B} \tag{3}$$

$$\mathcal{S}^{00} = \mathcal{S} \setminus \partial\mathcal{W} \setminus \partial\mathcal{B} \quad \textbf{not adjacent to } \mathcal{W} \textbf{ or } \mathcal{B} \tag{4}$$

  - **Return best partition from $(\mathcal{B}, \mathcal{S}, \mathcal{W})$,** $(\mathcal{B} \cup \mathcal{S}^{10}, \mathcal{S} \setminus \mathcal{S}^{10}, \mathcal{W})$ **and** $(\mathcal{B}, \mathcal{S} \setminus \mathcal{S}^{01}, \mathcal{W} \cup \mathcal{S}^{01})$

Partition before and after trimming

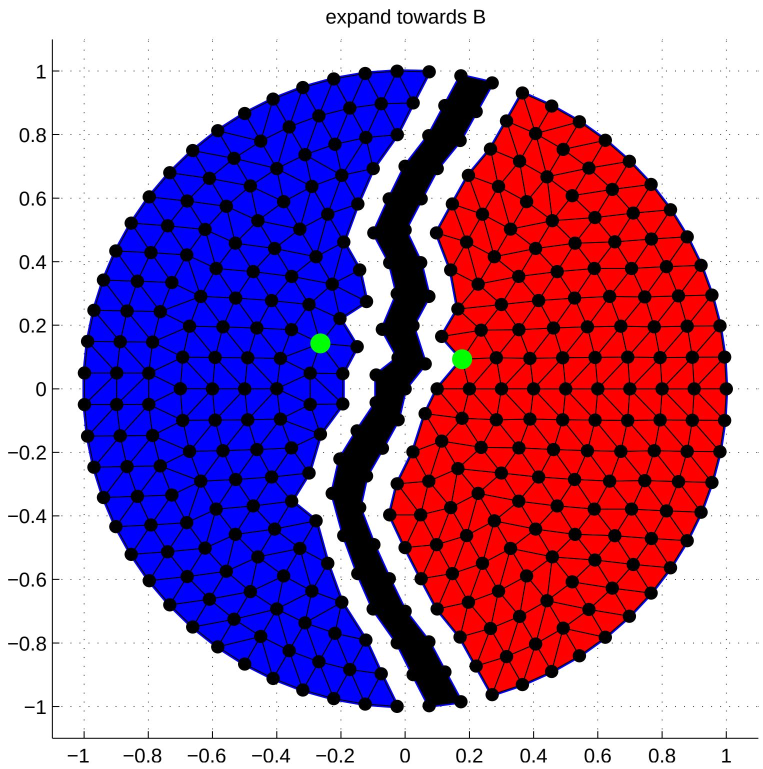


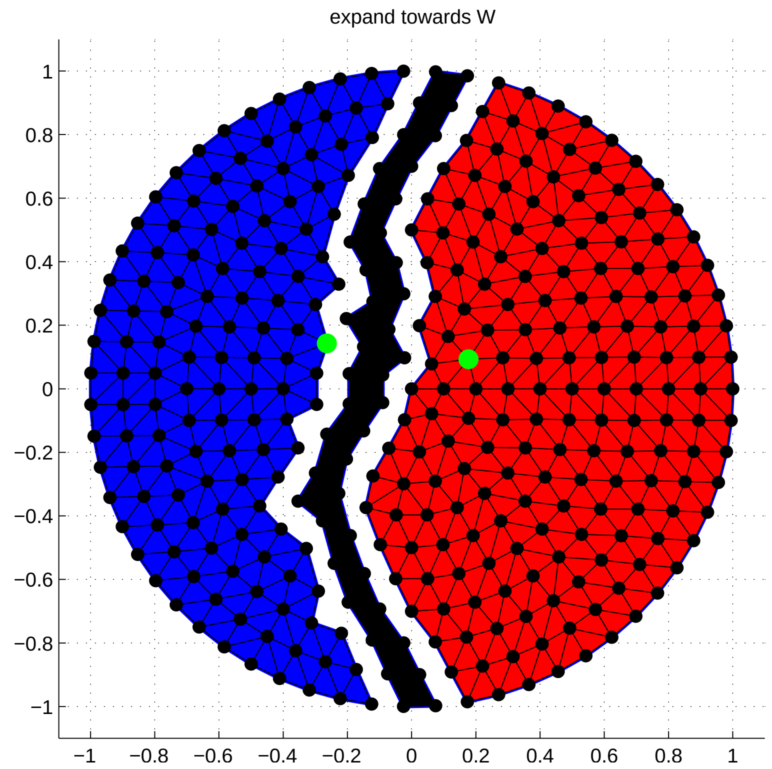


Outline of talk

- “Cold Start” to find separator
- Improve the partition
- **Expand to “wide” separator**
  - **Expand by levels, A. & Liu, 1998**
  - **“Cutting Corners”, selective expansion**
- Multiple breadth first search
- MPP Experiments
- Extensions and related work

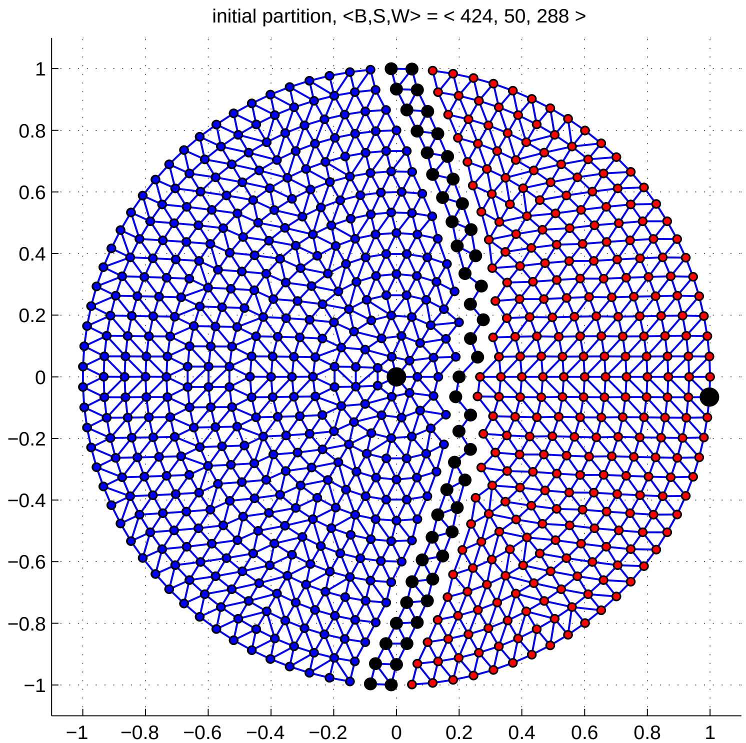
Expand minimal separator to wide separator add layers on one or both sides, A. & Liu, 1998



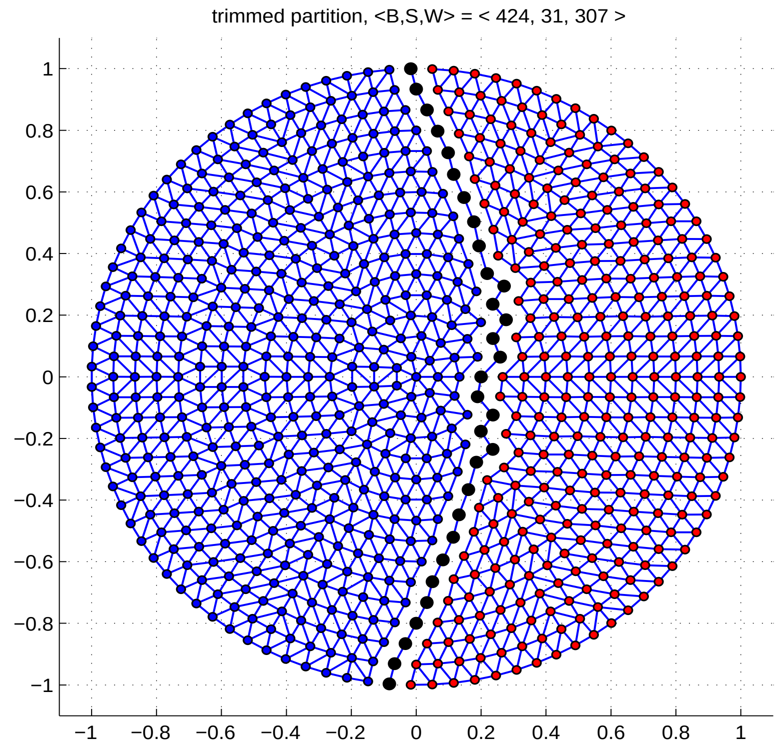


"Cutting Corners", selective widening

original partition

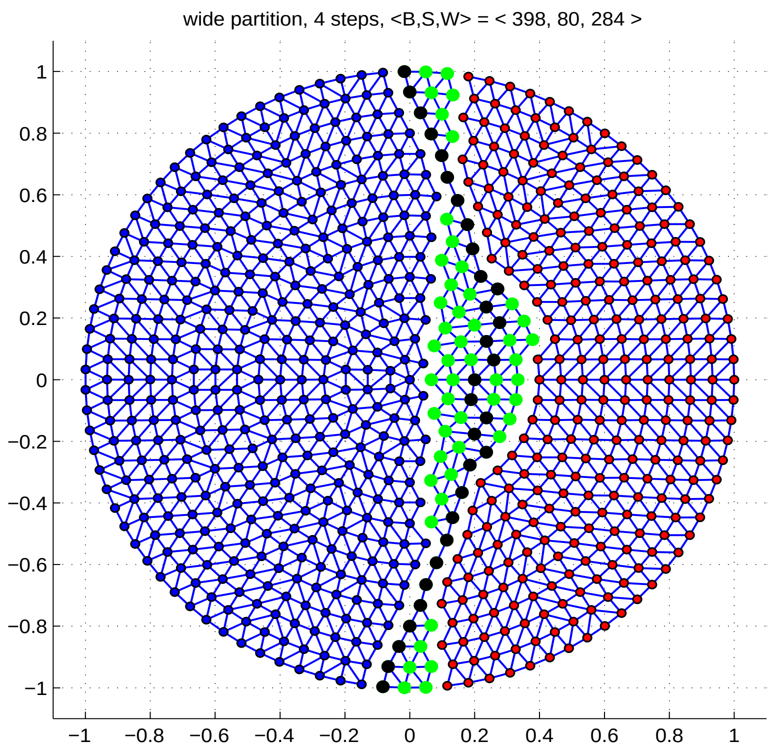


trimmed partition

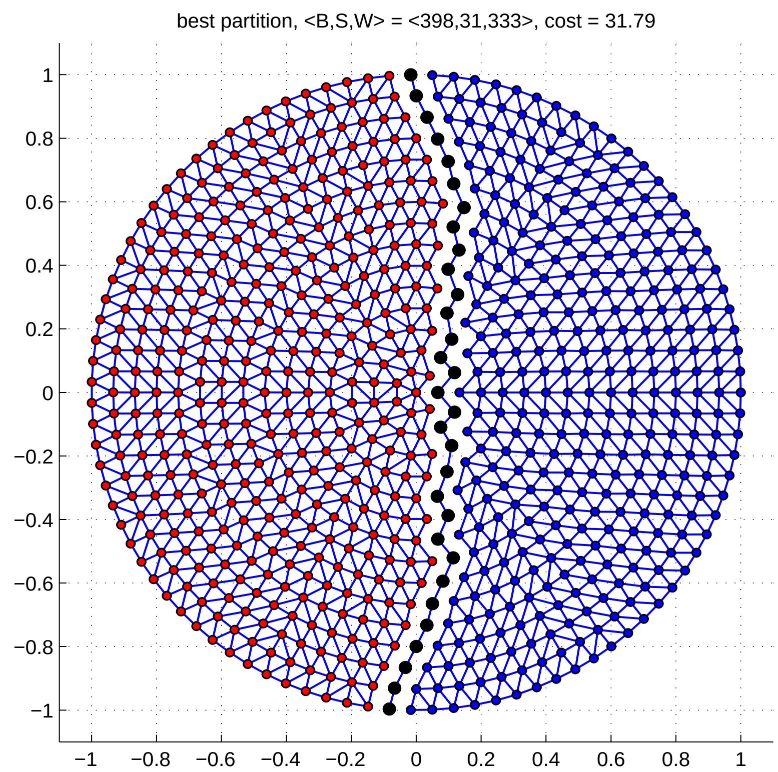


"Cutting Corners", selective widening (continued)

wide separator

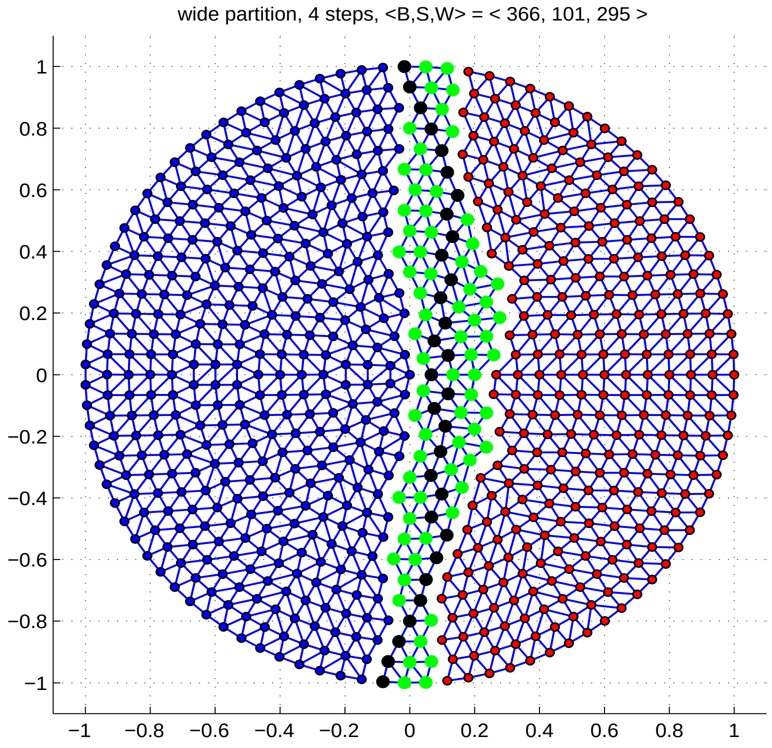


max flow separator

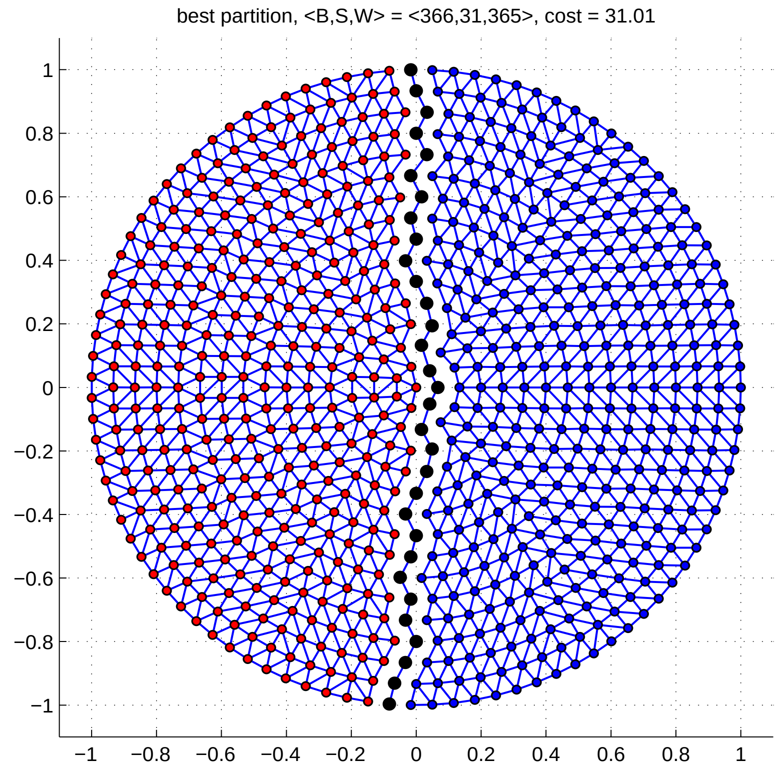


"Cutting Corners", selective widening (continued)

wide separator



max flow separator



Algorithm : $(\mathcal{B}, \mathcal{S}, \mathcal{W}) = \mathsf{findpartition}(\mathcal{V}, \mathcal{E})$

---

(1) $(\mathcal{B}^0, \mathcal{S}^0, \mathcal{W}^0) = \mathsf{coldstart}(\mathcal{V}, \mathcal{E})$
(2) $(\mathcal{B}^1, \mathcal{S}^1, \mathcal{W}^1) = \mathsf{trim}(\mathcal{B}, \mathcal{S}, \mathcal{W})$
(3) $(\mathcal{B}, \mathcal{S}, \mathcal{W}) =$ **better of** $(\mathcal{B}^0, \mathcal{S}^0, \mathcal{W}^0)$ **and** $(\mathcal{B}^1, \mathcal{S}^1, \mathcal{W}^1)$
(4) **while 1**
(5) $\quad (\widehat{\mathcal{B}}, \widehat{\mathcal{S}}, \widehat{\mathcal{W}}) = \mathsf{expand}(\mathcal{B}, \mathcal{S}, \mathcal{W})$
(6) $\quad (\mathcal{B}^*, \mathcal{S}^*, \mathcal{W}^*) = \mathsf{improve}(\widehat{\mathcal{B}}, \widehat{\mathcal{S}}, \widehat{\mathcal{W}})$
(7) $\quad$ **if** $(\mathcal{B}^*, \mathcal{S}^*, \mathcal{W}^*)$ **is no better than** $(\mathcal{B}, \mathcal{S}, \mathcal{W})$ **then**
(8) $\quad\quad$ **return**$(\mathcal{B}, \mathcal{S}, \mathcal{W})$
(9) $\quad$ **end if**
(10) $\quad (\mathcal{B}, \mathcal{S}, \mathcal{W}) = (\mathcal{B}^*, \mathcal{S}^*, \mathcal{W}^*)$
(11) **end while**

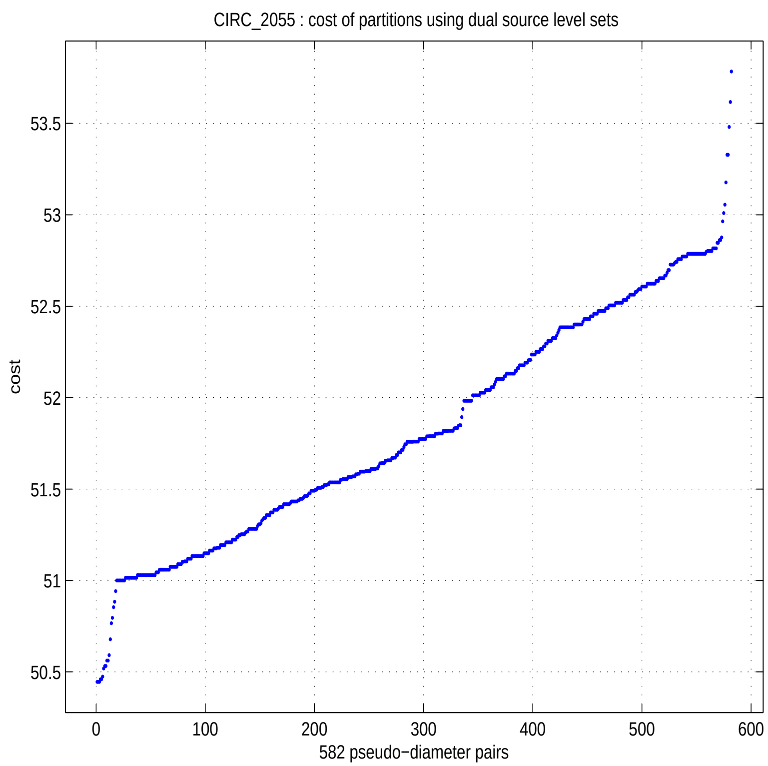
Outline of talk

- "Cold Start" to find separator
- Improve the partition
- Expand to "wide" separator
- Multiple breadth first search
  - Find pseudodiameter pair via sequential MBFS
  - Find many pseudodiameter pairs via independent MBFS
- MPP Experiments
- Extensions and related work

Find pseudo-diameter pair

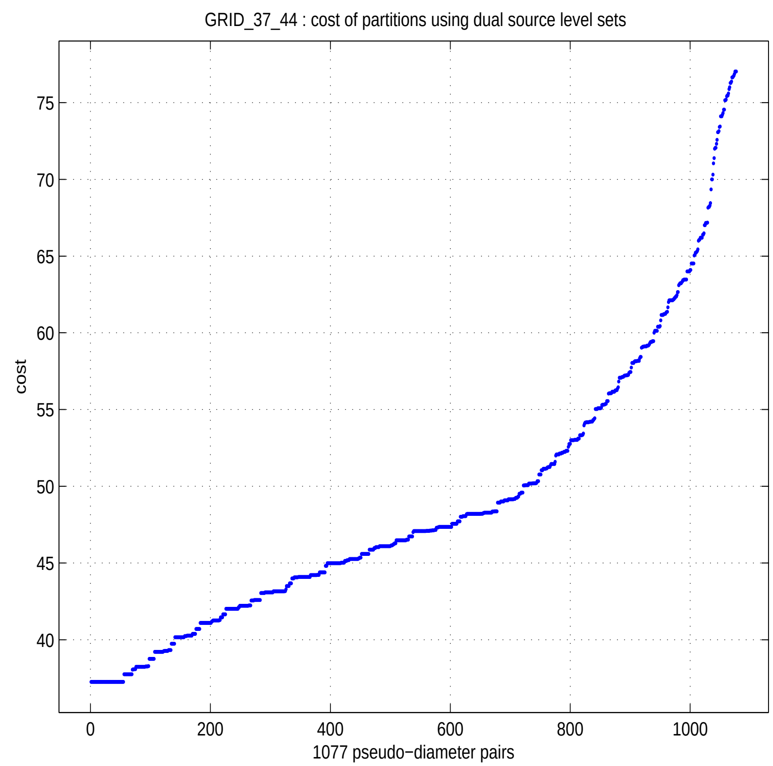
- s **and** t **are a diameter pair if** $d(s,t) = \max_{u,v} d(u,v)$
- s **and** t **are a pseudo-diameter pair**
  $d(s,t) = \max_u d(s,u) = \max_u d(t,u)$
- s **= random vertex ;** maxdist **= 0**
  while 1
      drop BFS from s
      find t **s.t.** $d(s,t) \geq d(s,u)$ **for all** u
      if $d(s,t) =$ maxdist **break**
      $s = t$ **;** maxdist $= d(s,t)$
  end
- **fast convergence, 3 or 4 iterations required**

CIRC_2055, triangulated disk, cost distribution



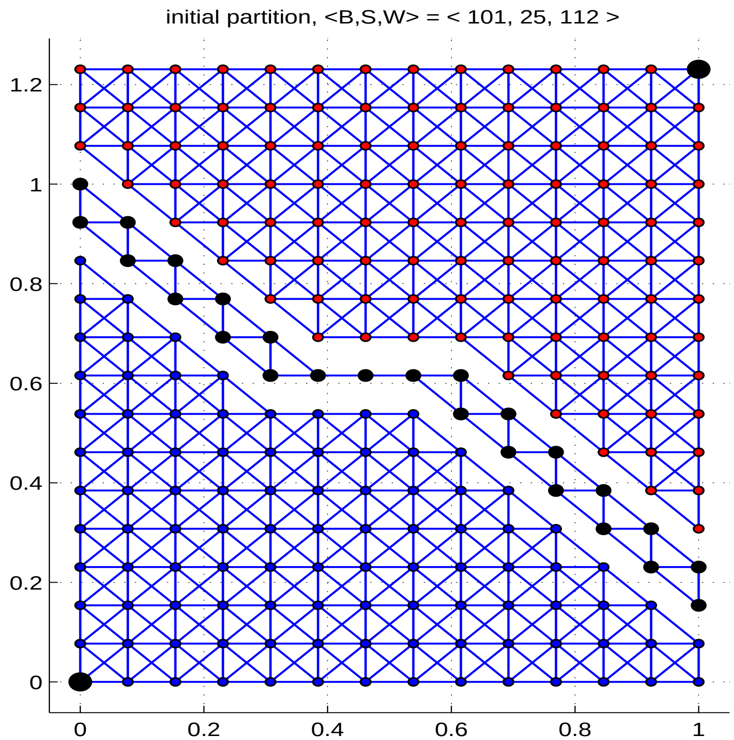
- dual source $(\mathcal{B}, \mathcal{S}, \mathcal{W})$ followed by trimming
- used each of the 2055 nodes as the seed
- 582 unique pseudo-diameter pairs
- average of 2.99 BFS used for each run, min 2, max 5
- cost varies by a factor of 1.07

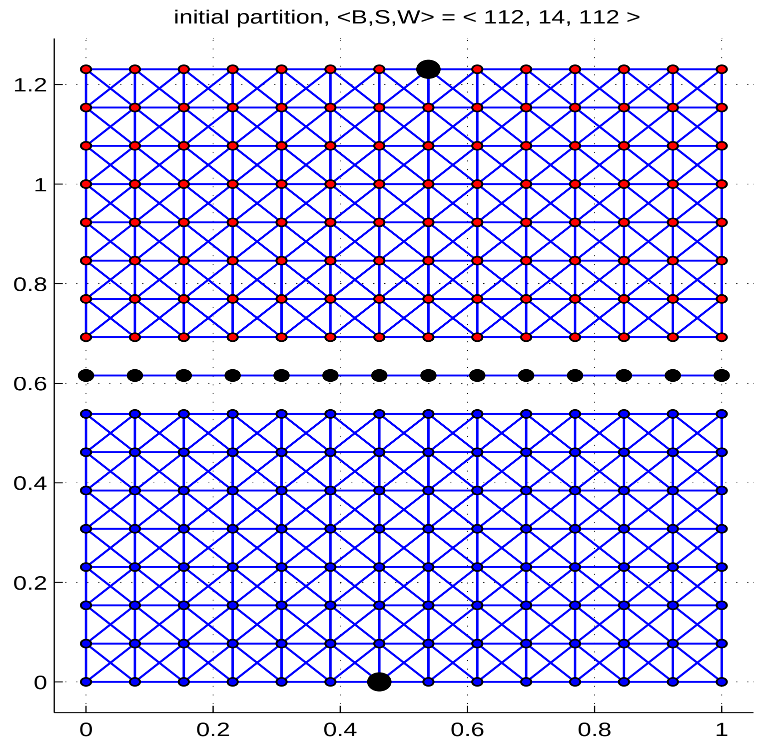
37×44 9-pt grid, cost distribution



- dual source $(\mathcal{B}, \mathcal{S}, \mathcal{W})$ followed by trimming
- used each of the 1628 nodes as the seed
- 1094 unique pseudo-diameter pairs
- average of 3.11 BFS used for each run, min 2, max 5
- cost varies by a factor of 2

Some pseudo-diameter pairs are good, some bad

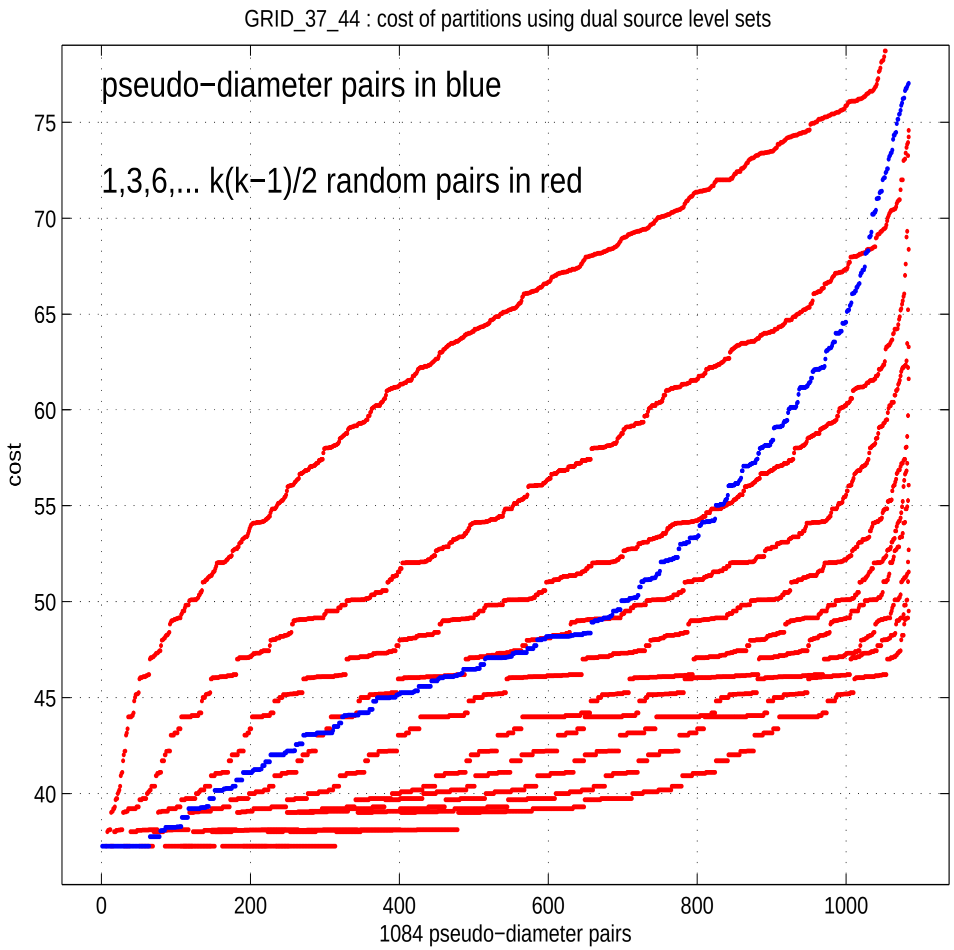




MIBFS – Multiple Independent Breadth First Searches

- k different root vertices
- compute k different distance vectors
- k(k-1)/2 different dual source pairs
- k(k-1)/2 different trimmed partitions
- In MPP distributed memory
  - All k BFS can be done together, graph is that of $I_{\mathsf{k}} \otimes A$
  - Dual source partitions can be done together
  - Trimming can also be done together

37×44 9-pt grid, MIBFS cost distribution



- **k $\geq$ 4 better than pseudo-diameter pairs**

Outline of talk

- “Cold Start” to find separator
- Improve the partition
- Expand to “wide” separator
- Multiple breadth first search
- MPP Experiments
- Extensions and related work

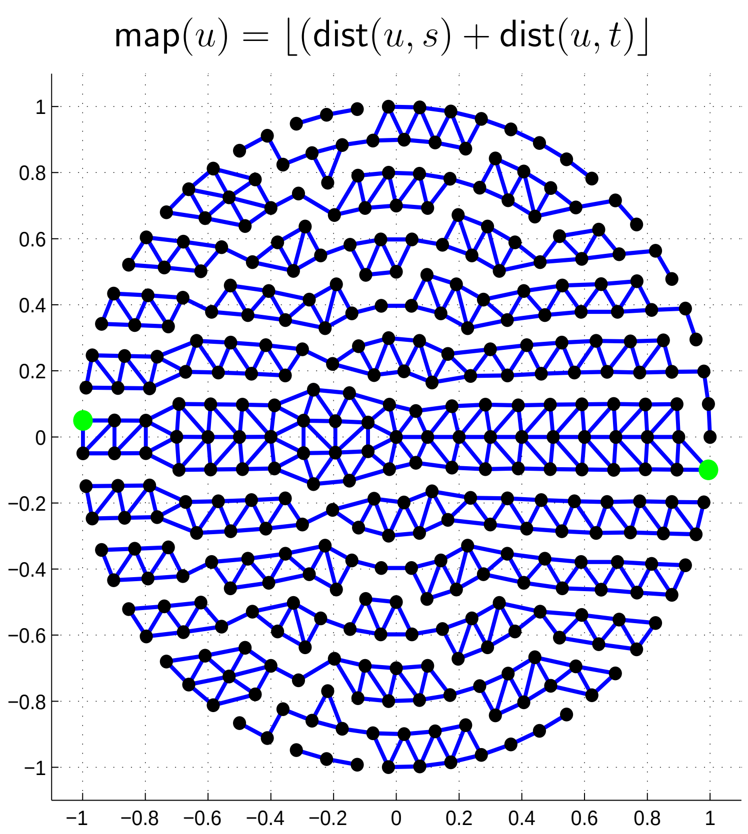
MPP experiments – work in progress

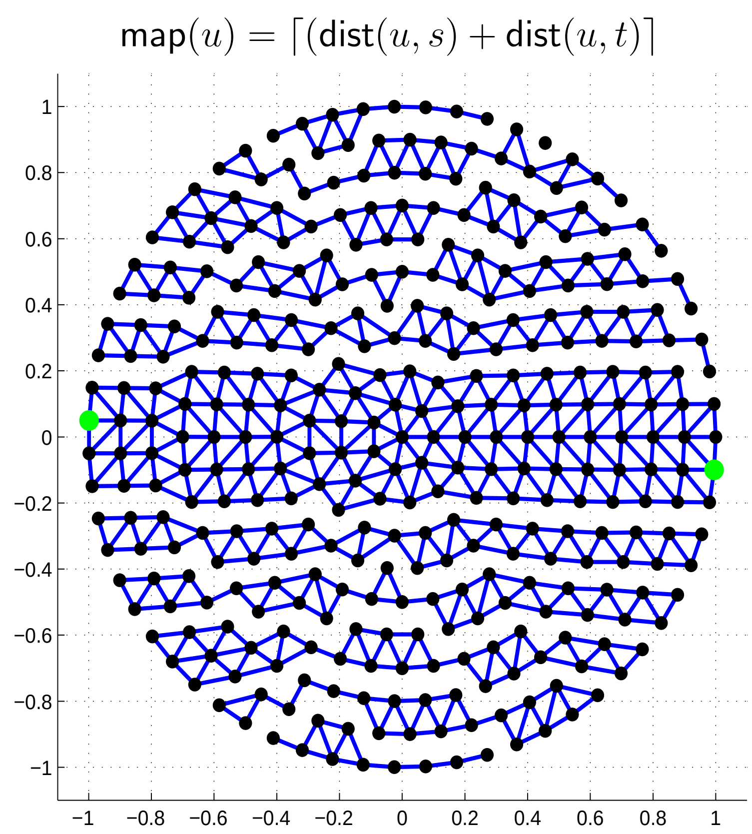
- **MIBFS "vectorized" across processors**
  - **cost for $\mathsf{k} = 4$ almost the same as $\mathsf{k} = 1$.**
- **trimming "vectorized" across processors**
- **To Do :**
  - **"vectorize" expansion to wide separators**
  - **explore several expansions simultaneously**
  - **gather one network onto one processor, improve via max flow**

Outline of talk

- “Cold Start” to find separator
- Improve the partition
- Expand to “wide” separator
- Multiple breadth first search
- MPP Experiments
- **Extensions and related work**
  - Linelets
  - Extension of farthest point clustering
  - Beyond bisection

Idea for dual source level sets comes from “linelets”





Farthest point clustering (Gonzales, 1985)

- **Goal : edge-based domain decomposition**
- **Strategy : find set of k maximally dispersed vertices to form centers of the domains**
- **Key point : given $\{c_1, c_2, \cdots, c_{k-1}\}$, find new center c_k s.t. $\min_i \mathsf{dist}(c_k, c_i)$ is maximized**
- **Sequential process :**
  - **start with random point c_1, perform BFS from c_1**
  - **find c_2 farthest from c_1, perform BFS from c_2**
  - **find c_3 farthest from c_1 and c_2, perform BFS from c_3, etc.**

Extension of farthest point clustering

Choose random $\{c_1, c_2, \cdots, c_{k-1}\}$
Perform MIBFS from $\{c_1, c_2, \cdots, c_{k-1}\}$
while not satisfied
  for each i **independently**
    remove c_i **from set**
    find best $\widehat{c_i}$ **with respect to others**
    evaluate scattering of the centers
  end for
  replace one or more centers, perform MIBFS
end while

Beyond bisection

- **Instead of $(\mathcal{B}, \mathcal{S}, \mathcal{W})$ bisection, consider trisection, quadrisection, octasection**
- **Find two or more levels of separators at once**
- **Replace $\mathsf{dist}(u, s) - \mathsf{dist}(u, t)$ with function of $\mathsf{dist}(u, s_1)$, $\mathsf{dist}(u, s_2)$, $\mathsf{dist}(u, s_3)$, etc**
- **Multiple component trimming**
- **Multiple component expansion**
- **Multiple component max flow solvers**

Summary

- Three improvements
  - single source vs dual source level sets
  - max flow (expensive) vs trimming (cheap)
  - expansion by levels vs selective expansion
- MIBFS (Multiple independent breadth first search)
  - cheap in MPP, # of communication steps is bounded above by the diameter of the graph
- Trimming the wide separator
  - cheap in MPP, # of communication steps is bounded above by the maximum width of a wide separator