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

\[ \mathcal{V} \implies S \quad \mathcal{B} \quad \mathcal{W} \]

- \( S \) separates \( \mathcal{B} \) from \( \mathcal{W} \)
- \( \mathcal{E} \cap (\mathcal{B} \times \mathcal{W}) = \emptyset \)
- \(|S| \) 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 $\rightarrow (\mathcal{B}, \mathcal{S}, \mathcal{W})$ partition

- $\text{level}(u) = \text{dist}(u, s)$
- 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

CIRC351: level sets and their weights

weight of level sets

21 levels

0 2 4 6 8 10 12 14 16 18 20 22

0 5 10 15 20 25
Level sets tridiagonal matrix

Cost Function \((\alpha, \beta)\)

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

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

if imbalance \(\leq\) \(\alpha\)

\[
\text{partition is acceptable}
\]

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

else

\[
\text{cost} = \infty
\]
endif
GENAND (George and Liu, 1983)

\[ \text{GENAND}(V, E) \]

- Find pseudoperipheral node \( s \)
- create compressed tridiagonal matrix from the level sets of \( s \)
- find best \((B, S, W)\) partition
- order \( S \) last
- \[ \text{GENAND}(B, E \cap (B \times B)) \]
- \[ \text{GENAND}(W, E \cap (W \times W)) \]
• $h \times k$ grid
• perfect balance,
  $\alpha = 1, |S| = \sqrt{2hk}$
• in general,
  $|S| = \frac{2}{\sqrt{1+\alpha}} \sqrt{hk}$
• relative size
  $1 \leq \frac{|S|}{\min(h,k)} \leq 2$
Dual source level half-sets $\rightarrow (\mathcal{B}, \mathcal{S}, \mathcal{W})$ partition

- $\text{level}(u) = \text{dist}(u, s) - \text{dist}(u, t)$
- 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 $\rightarrow (\mathcal{B}, \mathcal{S}, \mathcal{W})$ partition
Level weights histogram

CIRC351: half level sets and their weights

weight of half level sets vs. 41 levels
Level half-sets pentadiagonal matrix

Cost Function \((\alpha, \beta)\)

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

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

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

- Partition close to sink, \(\langle B, S, W \rangle = \langle 212, 21, 118 \rangle\), cost = 351.00
- Partition close to source, \(\langle B, S, W \rangle = \langle 177, 21, 153 \rangle\), cost = 21.43
Improve partition \((\mathcal{B}, S, \mathcal{W})\), via trimming

- basic idea, choose one of two moves
- while still improving
  - Partition separator \(S\) into four disjoint sets.
    \[
    S = S^{00} \sqcup S^{01} \sqcup S^{10} \sqcup S^{11}
    \]
    \[
    S^{11} = \partial \mathcal{B} \cap \partial \mathcal{W} \quad \text{adjacent to both} \quad (1)
    \]
    \[
    S^{10} = \partial \mathcal{B} \setminus \partial \mathcal{W} \quad \text{adjacent to } \mathcal{B}, \text{ not } \mathcal{W} \quad (2)
    \]
    \[
    S^{01} = \partial \mathcal{W} \setminus \partial \mathcal{B} \quad \text{adjacent to } \mathcal{W}, \text{ not } \mathcal{B} \quad (3)
    \]
    \[
    S^{00} = S \setminus \partial \mathcal{W} \setminus \partial \mathcal{B} \quad \text{not adjacent to } \mathcal{W} \text{ or } \mathcal{B} \quad (4)
    \]
  - Return best partition from \((\mathcal{B}, S, \mathcal{W})\),
    \((\mathcal{B} \cup S^{10}, S \setminus S^{10}, \mathcal{W})\) and \((\mathcal{B}, S \setminus S^{01}, \mathcal{W} \cup S^{01})\)
Partition before and after trimming

CIRC351 : width 2, (B,S,W) = (175,58,118), cost 60.826

CIRC351 : trimmed width 2, (B,S,W) = (175,22,154), cost 22.395
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

initial partition, \(<B,S,W> = <424, 50, 288 >\)

trimmed partition, \(<B,S,W> = <424, 31, 307 >\)
“Cutting Corners”, selective widening (continued)

**wide separator**

![Wide separator diagram](image)

- wide partition, 4 steps, \(<B,S,W> = <398,80,284>\)

**max flow separator**

![Max flow separator diagram](image)

- best partition, \(<B,S,W> = <398,31,333>, \text{cost} = 31.79\)
“Cutting Corners”, selective widening (continued)

Wide separator

Max flow separator

Wide partition, 4 steps, $<B,S,W> = <366, 101, 295>$

Best partition, $<B,S,W> = <366, 31, 365>$, cost = 31.01
Algorithm : \((B, S, W) = \text{findpartition}(V, E)\)

(1) \((B^0, S^0, W^0) = \text{coldstart}(V, E)\)
(2) \((B^1, S^1, W^1) = \text{trim}(B, S, W)\)
(3) \((B, S, W) = \text{better of } (B^0, S^0, W^0) \text{ and } (B^1, S^1, W^1)\)
(4) while 1
(5) \((\hat{B}, \hat{S}, \hat{W}) = \text{expand}(B, S, W)\)
(6) \((B^*, S^*, W^*) = \text{improve}(\hat{B}, \hat{S}, \hat{W})\)
(7) if \((B^*, S^*, W^*)\) is no better than \((B, S, W)\) then
(8) return \((B, S, W)\)
(9) end if
(10) \((B, S, W) = (B^*, S^*, W^*)\)
(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 \( d(s, t) = \text{maxdist} \) break
    *s* = *t* ; maxdist = \( d(s, t) \)
  end

• fast convergence, 3 or 4 iterations required
CIRC_2055, triangulated disk, cost distribution

- dual source \((B, S, W)\) 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
$37 \times 44$ 9-pt grid, cost distribution

- dual source $(B, S, W)$ 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

initial partition, \( <B,S,W> = <101, 25, 112> \)

initial partition, \( <B,S,W> = <112, 14, 112> \)
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 $I_k \otimes A$
  - Dual source partitions can be done together
  - Trimming can also be done together
$37 \times 44$ 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 $k = 4$ almost the same as $k = 1$.
• 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”

\[ \text{map}(u) = \lceil \text{dist}(u, s) + \text{dist}(u, t) \rceil \]

\[ \text{map}(u) = \lfloor \text{dist}(u, s) + \text{dist}(u, t) \rfloor - 1 \]
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. $\min_i \text{dist}(c_k, c_i)$ 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 \( \tilde{c}_i \) with respect to others
        evaluate scattering of the centers
    end for
    replace one or more centers, perform MIBFS
end while
Beyond bisection

- Instead of $(B, S, W)$ bisection, consider trisection, quadrisection, octasection
- Find two or more levels of separators at once
- Replace $\text{dist}(u, s) - \text{dist}(u, t)$ with function of $\text{dist}(u, s_1), \text{dist}(u, s_2), \text{dist}(u, s_3)$, 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