A polynomial-time partitioning algorithm for weighted cactus graphs

Maike Buchin, Leonie Selbach
Ruhr University Bochum

EuroCG 2020
Motivation & Problem

\(p-(l, u)\)-partition problem

Given: Vertex-weighted graph \(G = (V, E, w)\),
non-negative integers \(l\) and \(u\) with \(l \leq u\),
positive integer \(p\).
Motivation & Problem

$p$-$(l, u)$-partition problem

Given: Vertex-weighted graph $G = (V, E, w)$, non-negative integers $l$ and $u$ with $l \leq u$, positive integer $p$

Find a vertex partition into $p$ clusters $V_1, V_2, \ldots, V_p$ such that $l \leq w(V_i) \leq u$ for all $1 \leq i \leq p$. 
Motivation & Problem

\( p-(l,u) \)-partition problem

Given: Vertex-weighted graph \( G = (V, E, w) \), non-negative integers \( l \) and \( u \) with \( l \leq u \), positive integer \( p \)

Find a vertex partition into \( p \) clusters \( V_1, V_2, \ldots, V_p \) connected and pairwise disjoint

such that \( l \leq w(V_i) \leq u \) for all \( 1 \leq i \leq p \).
**Motivation & Problem**

*p-(l, u)-partition problem*

Given: Vertex-weighted graph $G = (V, E, w)$, non-negative integers $l$ and $u$ with $l \leq u$, positive integer $p$

Find a vertex partition into $p$ clusters $V_1, V_2, \ldots, V_p$ such that $l \leq w(V_i) \leq u$ for all $1 \leq i \leq p$.

$$= \sum_{v \in V_i} w(v)$$
Motivation & Problem

$p$-(l, u)-partition problem
Given: Vertex-weighted graph $G = (V, E, w)$, non-negative integers $l$ and $u$ with $l \leq u$, positive integer $p$

Find a vertex partition into $p$ clusters $V_1, V_2, \ldots, V_p$ such that $l \leq w(V_i) \leq u$ for all $1 \leq i \leq p$.

Motivation
Fragmentation of biomedical structures
Motivation & Problem

$p$-$(l, u)$-partition problem
Find a $(l, u)$-partition with exactly $p$ clusters.
Motivation & Problem

\(p-(l, u)-\text{partition problem}\)
Find a \((l, u)\)-partition with exactly \(p\) clusters.
Motivation & Problem

$p$-$(l, u)$-partition problem
Find a $(l, u)$-partition with exactly $p$ clusters.

6-$(3, 12)$-partition
Motivation & Problem

$ p$-$(l, u)$-partition problem
Find a $(l, u)$-partition with exactly $p$ clusters.

6-$(3, 12)$-partition

Minimum/maximum-$(l, u)$-partition problem
Find a $(l, u)$-partition with the minimal resp. maximal number of clusters.
Motivation & Problem

\( p-(l, u) \)-partition problem
Find a \((l, u)\)-partition with exactly \(p\) clusters.

\[ \text{6-}(3, 12)\text{-partition} \]

Minimum/maximum-\((l, u)\)-partition problem
Find a \((l, u)\)-partition with the minimal resp. maximal number of clusters.

\[ \text{minimum (3,12)-partition} \]
(Related) Results

Related results
• series-parallel graphs NP-hard
(Related) Results

Related results

• series-parallel graphs \( \text{NP-hard} \)
  \( \mathcal{O}(u^4 p^2 n) \) \( \mathcal{O}(u^4 n) \)

• partial k-trees
  \( \mathcal{O}(u^{2(k+1)} p^2 n) \) \( \mathcal{O}(u^{2(k+1)} n) \)

\( p-(l, u) \)-partition problem
\( \text{min/max}-(l, u) \)-partition problem
(Related) Results

Related results

• series-parallel graphs  \textbf{NP-hard}
  \[ O(u^4 p^2 n) \quad O(u^4 n) \]

• partial k-trees
  \[ O(u^{2(k+1)} p^2 n) \quad O(u^{2(k+1)} n) \]

• trees  \textbf{polynomial}  (Decision)
  \[ O(p^4 n) \quad O(n^5) \]

\begin{itemize}
  \item \textit{p-} (l, u)-\textit{partition problem}
  \item \textit{min/max-} (l, u)-\textit{partition problem}
\end{itemize}
(Related) Results

Related results
- series-parallel graphs: NP-hard
  \[ \mathcal{O}(u^4 p^2 n) \quad \mathcal{O}(u^4 n) \]
- partial k-trees
  \[ \mathcal{O}(u^{2(k+1)} p^2 n) \quad \mathcal{O}(u^{2(k+1)} n) \]
- trees: polynomial (Decision)
  \[ \mathcal{O}(p^4 n) \quad \mathcal{O}(n^5) \]

Our results
- cactus graphs: polynomial
  \[ \mathcal{O}(p^4 n^2) \quad \mathcal{O}(n^6) \]

\[-\text{p-}(l, u)\text{-partition problem}\]
\[-\text{min/max-}(l, u)\text{-partition problem}\]
(Related) Results

Related results
• series-parallel graphs \( \text{NP-hard} \)
  \( O(u^4p^2n) \quad O(u^4n) \)
• partial k-trees
  \( O(u^{2(k+1)}p^2n) \quad O(u^{2(k+1)}n) \)
• trees \( \text{polynomial} \)  
  \( O(p^4n) \quad O(n^5) \)

Our results
• cactus graphs \( \text{polynomial} \)  
  \( O(p^4n^2) \quad O(n^6) \)
Preprocessing

DFS on some vertex $r \in G$ and store cycles
Preprocessing

DFS on some vertex \( r \in G \) and store cycles
Preprocessing

DFS on some vertex $r \in G$ and store cycles
Preprocessing

DFS on some vertex $r \in G$ and store cycles
Preprocessing

DFS on some vertex $r \in G$ and store cycles
Preprocessing

DFS on some vertex $r \in G$ and store cycles
Preprocessing

DFS on some vertex $r \in G$ and store cycles

$$C(r, z) = \langle r, x, y, z \rangle$$
Some Definitions

For a partition $P$ of $T_v$:

$|P| = \text{number of clusters in } P$

$C_v = \text{cluster containing the node } v$
Some Definitions

For a partition $P$ of $T_v$:

$|P|$ = number of clusters in $P$

$C_v$ = cluster containing the node $v$

subtree rooted in node $v$
Some Definitions

For a partition $P$ of $T_v$:

$|P| = \text{number of clusters in } P$

$C_v = \text{cluster containing the node } v$

Extendable $(l, u)$-partition of $T_v$:

$\bullet w(C_v) \leq u$

$\bullet l \leq w(C') \leq u$ for every cluster $C' \neq C_v$
Some Definitions

Partition Sets

\[ S(T_v) = \{(x, k) \mid \exists \text{ extendable } (l, u)\text{-partition } P \text{ of } T_v \]

\[ \text{such that } |P| = k \land w(C_v) = x \} \]
Some Definitions

Partition Sets

\[ S(T_v) = \{ (x, k) \mid \exists \text{ extendable } (l, u)-\text{partition } P \text{ of } T_v \text{ such that } |P| = k \land w(C_v) = x \} \]

\[ |S(T_v)| = \mathcal{O}(up) \]
Some Definitions

Partition Sets

\[ S(T_v) = \{ (x, k) \mid \exists \text{ extendable } (l, u)\text{-partition } P \text{ of } T_v \text{ such that } |P| = k \land w(C_v) = x \} \]

Lemma
\[ T_v \text{ has } p-(l, u)\text{-partition} \iff \exists (x, p) \in S(T_v) \text{ such that } l \leq x \leq u \]
Some Definitions

Partition Sets

\[ S(T_v) = \{ (x, k) \mid \exists \text{ extendable } (l, u) \text{-partition } P \text{ of } T_v \]
\[ \text{such that } |P| = k \land w(C_v) = x \} \]

Lemma

\( T_v \) has \( p \)-(\( l, u \))-partition \( \iff \exists (x, p) \in S(T_v) \text{ such that } l \leq x \leq u \)

Idea

Compute \( S(T_r) \) for \( r \) being the root of the DFS-Tree
Some Definitions

Partition Sets

\[ S(T_v) = \{(x, k) \mid \exists \text{ extendable } (l, u)\text{-partition } P \text{ of } T_v \]
\[ \text{such that } |P| = k \land w(C_v) = x \}\]

Lemma

\( T_v \) has \( p \)-(\( l, u \))-partition \( \iff \exists (x, p) \in S(T_v) \) such that \( l \leq x \leq u \)

Idea

Compute \( S(T_r) \) for \( r \) being the root of the DFS-Tree

+ include an efficient procedure for the cycles
Method

Compute partition $P$ of $T^i_v$ by combining partitions of $T^{i-1}_v$ and $T_{v_i}$
Compute partition $P$ of $T_v^i$ by combining partitions of $T_v^{i-1}$ and $T_{v_i}$.
Method

Compute partition $P$ of $T_v^i$ by combining partitions of $T_v^{i-1}$ and $T_{v_i}$.
Method

Compute partition $P$ of $T_v^i$ by combining partitions of $T_v^{i-1}$ and $T_{v_i}$.
Method

Compute partition $P$ of $T_v^i$ by combining partitions of $T_v^{i-1}$ and $T_{v_i}$

Given partitions $P'$ of $T_v^{i-1}$ and $P''$ of $T_{v_i}$.
Method

Compute partition $P$ of $T_{v_i}^i$ by combining partitions of $T_{v_i}^{i-1}$ and $T_{v_i}$

Given partitions $P'$ of $T_{v_i}^{i-1}$ and $P''$ of $T_{v_i}$.

as $(x_1, k_1) \in S(T_{v_i}^{i-1})$ and $(x_2, k_2) \in S(T_{v_i})$
Method

Compute partition $P$ of $T^i_v$ by combining partitions of $T^{i-1}_v$ and $T_{v_i}$

Option 1: merge

$w(C_v) = w(C''_v) + w(C''''_v)$

$|P| = |P'| + |P''| - 1$

$(x_1 + x_2, k_1 + k_2 - 1) \in S(T^i_v)$

if $x_1 + x_2 \leq u$ and $k_1 + k_2 - 1 \leq p$
Method

Compute partition $P$ of $T^i_v$ by combining partitions of $T^{i-1}_v$ and $T_{vi}$

Option 2: don’t merge

$w(C_v) = w(C'_v)$

$|P| = |P'| + |P''|$

$(x_1, k_1 + k_2) \in S(T^i_v)$

if $x_2 \geq l$ and $k_1 + k_2 \leq p$
Method

Compute partition $P$ of $T^i_v$ by combining partitions of $T^i_v^{-1}$ and $T_{v_i}$
Method

Compute partition $P$ of $T^i_v$ by combining partitions of $T^{i-1}_v$ and $T^i_{v_i}$

Compute set $S(T^i_v)$ by combining sets $S(T^{i-1}_v)$ and $S(T^i_{v_i})$
Method

Compute partition \( P \) of \( T_v^i \) by combining partitions of \( T_v^{i-1} \) and \( T_v^i \).

Compute set \( S(T_v^i) \) by combining sets \( S(T_v^{i-1}) \) and \( S(T_v^i) \):

\[
S(T_v^i) = S(T_v^{i-1}) \oplus S(T_v^i) \\
= \{(x_1 + x_2, k_1 + k_2 - 1) | x_1 + x_2 \leq u, k_1 + k_2 - 1 \leq p\} \\
\cup \{(x_1, k_1 + k_2) | l \leq x_2, k_1 + k_2 \leq p\}
\]

\( \oplus \)-operation \( O(u^2p^2) \)
Method

Compute partition $P$ of $T^i_v$ by combining partitions of $T^{i-1}_v$ and $T^i_{v_i}$

Compute set $S(T^i_v)$ by combining sets $S(T^{i-1}_v)$ and $S(T^i_{v_i})$

Dynamic approach

$S(T^0_v) = \{(w(v), 1)\}$

$S(T^i_v) = S(T^{i-1}_v) \oplus S(T^i_{v_i})$ for all edges $(v, v_i)$

bottom-up
Method

What about cycles?

consider different *configurations*
Method

What about cycles?

consider different \textit{configurations}
Method

What about cycles?

consider different configurations

\[
\begin{align*}
  w_0 & \quad \cdots \quad w_3 \\
  w_1 & \quad w_2 \\
  w_0 & \quad w_3 \\
  w_1 & \quad w_2 \\
  w_0 & \quad w_3 \\
  w_1 & \quad w_2 \\
  w_0 & \quad w_3 \\
  w_1 & \quad w_2 
\end{align*}
\]
Method

What about cycles?

consider different configurations
Method

What about cycles?

consider different configurations
Method

What about cycles in the graph?

consider different configurations
What about cycles in the graph?

consider different configurations
Method

What about cycles in the graph?

consider different configurations
What about cycles in the graph?

consider different configurations
Method

Computation of the partition sets
for nodes in some cycle \( C(w_0, w_{m-1}) = \langle w_0, w_1, \ldots, w_{m-1} \rangle \)
in different configurations \( j \)

\[
S_j(T_{w_{m-1}}) = \begin{cases} 
  S(T_{w_{m-1}}) & j = 1, 2 \\
  S(T_{w_{m-1}}) \oplus S_j(T_{w_{m-2}}) & \text{otherwise}
\end{cases}
\]

\[
S_j(T_{w_i}) = \begin{cases} 
  S(T_{w_i}^{c_i-1}) \oplus S_j(T_{w_{i+1}}) & j < m - i \\
  S(T_{w_i}^{c_i-1}) \oplus S_j(T_{w_{i-1}}) & j > m - i + 1 \\
  S(T_{w_i}^{c_i-1}) & \text{otherwise}
\end{cases}
\]

\[
S_j(T_{w_0}) = \begin{cases} 
  S(T_{w_0}^{k-1}) \oplus S_j(T_{w_1}) & j = 1 \\
  (S(T_{w_0}^{k-1}) \oplus S_j(T_{w_1})) \oplus S_j(T_{w_{m-1}}) & \text{otherwise}
\end{cases}
\]
Method

Computation of the partition sets
for nodes in some cycle $C(w_0, w_{m-1}) = \langle w_0, w_1, \ldots, w_{m-1} \rangle$
in different configurations $j$

\[
S_j(T_{w_{m-1}}) = \begin{cases} 
S(T_{w_{m-1}}) & j = 1, 2 \\
S(T_{w_{m-1}}) \oplus S_j(T_{w_{m-2}}) & \text{otherwise}
\end{cases}
\]

\[
S_j(T_{w_i}) = \begin{cases} 
S(T_{w_i}^{c_i-1}) \oplus S_j(T_{w_{i+1}}) & j < m - i \\
S(T_{w_i}^{c_i-1}) \oplus S_j(T_{w_{i-1}}) & j > m - i + 1 \\
S(T_{w_i}^{c_i-1}) & \text{otherwise}
\end{cases}
\]

\[
S_j(T_{w_0}) = \begin{cases} 
S(T_{w_0}^{k-1}) \oplus S_j(T_{w_1}) & j = 1 \\
(S(T_{w_0}^{k-1}) \oplus S_j(T_{w_1})) \oplus S_j(T_{w_{m-1}}) & \text{otherwise}
\end{cases}
\]

\[
S(T_{w_0}^k) = \bigcup_{j=1}^{m-1} S_j(T_{w_0}^k)
\]
Theorem
Given a weighted cactus graph $G$, a positive integer $p$ and two non-negative integers $l$ and $u$ (with $l \leq u$). The $p$-$(l, u)$-partition problem can be decided in time $O(u^2 p^2 n^2)$. 

Results
Theorem
Given a weighted cactus graph $G$, a positive integer $p$ and two non-negative integers $l$ and $u$ (with $l \leq u$). The $p$-$(l, u)$-partition problem can be decided in time $O(u^2 p^2 n^2)$. $O(p^4 n^2)$
reduce size of partition sets from $O(up)$ to $O(p^2)$
Results

**Theorem**

Given a weighted cactus graph $G$, a positive integer $p$ and two non-negative integers $l$ and $u$ (with $l \leq u$). The $p$-$(l, u)$-partition problem can be decided in time $O(u^2p^2n^2)$. $O(p^4n^2)$

reduce size of partition sets from $O(up)$ to $O(p^2)$

before:

```
\begin{array}{c}
\bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & x_m \\
x_1 & x_2 & x_3 & \quad & \quad & \quad & \quad & \quad & \quad & \quad & \quad & \quad \end{array}
```

for each $k \ (\leq p)$ \quad \Rightarrow \quad overall \ $O(up)$

\[ m \leq u \]
Results

Theorem
Given a weighted cactus graph $G$, a positive integer $p$ and two non-negative integers $l$ and $u$ (with $l \leq u$). The $p$-$(l, u)$-partition problem can be decided in time $O(u^2 p^2 n^2)$. $O(p^4 n^2)$

reduce size of partition sets from $O(up)$ to $O(p^2)$

before:

for each $k \ (\leq p)$  $\Rightarrow$ overall $O(up)$

$d = u - l \quad \leq d$  

$\Rightarrow \quad m \leq u$
Results

Theorem
Given a weighted cactus graph $G$, a positive integer $p$ and two non-negative integers $l$ and $u$ (with $l \leq u$). The $p$-$(l, u)$-partition problem can be decided in time $O(u^2 p^2 n^2)$. $O(p^4 n^2)$

reduce size of partition sets from $O(up)$ to $O(p^2)$

before: $d = u - l \leq d$ for each $k$ ($\leq p$) $\Rightarrow$ overall $O(up)$

now: $[a_1, a_2]$ $[a_2, a_3]$ $[a_l, a_{l+1}]$ for each $k$ ($\leq p$) $\Rightarrow$ overall $O(p^2)$
Results

**Theorem**
Given a weighted cactus graph $G$, a positive integer $p$ and two non-negative integers $l$ and $u$ (with $l \leq u$). The $p$-$(l, u)$-partition problem can be decided in time $O(p^4 n^2)$ and space $O(p^5 n^2)$. 

by storing additional information
and using backtracking
Theorem
Given a weighted cactus graph $G$ and two non-negative integers $l$ and $u$ (with $l \leq u$). The minimum and maximum $(l, u)$-partition problem can be solved in time $O(n^6)$ and space $O(n^7)$. 
Open problems

- NP-hard
- Polynomial-time algorithms for other graph classes?