Sharing the Cost More Efficiently: Improved Approximation for the Multicommodity Rent-or-Buy Problem

Jochen Könemann

Department of Combinatorics & Optimization
University of Waterloo
Work done while visiting Università di Roma “La Sapienza”.

Joint with L. Becchetti, S. Leonardi, and M. Pál
A roadmap for this talk

- MRoB: Problem definition and previous work
  - Prelude: Steiner forests
    - MRoB definition
    - Previous work and our contribution
  - Approximating MRoB: Recap of [Gupta et al.’03]
- Our algorithm and its analysis
Steiner forests: **Intro**

**Input:**
- Undirected graph $G = (V, E)$
- Edge costs $c_e \geq 0$ for all $e \in E$.
- Terminal-pairs $R = \{(s_1, t_1), \ldots, (s_k, t_k)\} \subseteq V$

**Goal:** Compute min-cost forest $F$ in $G$ such that $s$ and $t$ are in same tree for all $(s, t) \in R$
**Steiner forests: Example**

- Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)
- All edges have unit cost.

![Diagram](diagram.png)
Steiner forests: Example

- Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)
- All edges have unit cost.

Total cost is 4!
A roadmap for this talk

- MRoB: Problem definition and previous work
  - Prelude: Steiner forests
  - MRoB definition
  - Previous work and our contribution
- Approximating MRoB: Recap of [Gupta et al.'03]
- Our algorithm and its analysis
MRoB: Problem Definition

Input:
- Undirected graph $G = (V, E)$, costs $c_e \geq 0$ for all $e \in E$.
- Terminal-pairs $R = \{(s_1, t_1), \ldots, (s_k, t_k)\} \subseteq V$
- Parameter $M \geq 1$
MRoB: Problem Definition

- **Input:**
  - Undirected graph $G = (V, E)$, costs $c_e \geq 0$ for all $e \in E$.
  - Terminal-pairs $R = \{(s_1, t_1), \ldots, (s_k, t_k)\} \subseteq V$
  - Parameter $M \geq 1$

- Solution has **bought** edges $E_b$ and **rented** edges $E_r$. $E_r \cup E_b$ contains an $s, t$-path for all $(s, t) \in R$
MRoB: Problem Definition

- **Input:**
  - Undirected graph $G = (V, E)$, costs $c_e \geq 0$ for all $e \in E$.
  - Terminal-pairs $R = \{(s_1, t_1), \ldots, (s_k, t_k)\} \subseteq V$
  - Parameter $M \geq 1$

- Solution has **bought** edges $E_b$ and **rented** edges $E_r$
  - $E_r \cup E_b$ contains an $s, t$-path for all $(s, t) \in R$
  - Bought edge $e \in E_b$ costs $M \cdot c_e$
**MRoB: Problem Definition**

- **Input:**
  - Undirected graph $G = (V, E)$, costs $c_e \geq 0$ for all $e \in E$.
  - Terminal-pairs $R = \{(s_1, t_1), \ldots, (s_k, t_k)\} \subseteq V$
  - Parameter $M \geq 1$

- Solution has **bought** edges $E_b$ and **rented** edges $E_r$
  $E_r \cup E_b$ contains an $s, t$-path for all $(s, t) \in R$

- **Bought** edge $e \in E_b$ costs $M \cdot c_e$

- **Rented** edge $e \in E_r$ costs $f_e \cdot c_e$
  $f_e$ is the flow on $e$: \# pairs in $R$ separated by $e$
MRoB: Problem Definition

- **Input:**
  - Undirected graph \( G = (V, E) \), costs \( c_e \geq 0 \) for all \( e \in E \).
  - Terminal-pairs \( R = \{(s_1, t_1), \ldots, (s_k, t_k)\} \subseteq V \)
  - Parameter \( M \geq 1 \)

- Solution has **bought** edges \( E_b \) and **rented** edges \( E_r \)
  - \( E_r \cup E_b \) contains an \( s, t \)-path for all \( (s, t) \in R \)

- **Bought** edge \( e \in E_b \) costs \( M \cdot c_e \)

- **Rented** edge \( e \in E_r \) costs \( f_e \cdot c_e \)
  - \( f_e \) is the flow on \( e \): \# pairs in \( R \) separated by \( e \)

- **Goal:** Compute a solution of minimum cost
Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)

All edges have unit cost and \( M = 2 \)
MRoB: Example

- Example with four terminal pairs: $R = \{(s_i, t_i)\}_{1 \leq i \leq 4}$
- All edges have unit cost and $M = 2$

Total cost: $2 \cdot 1 + 4 = 6$
A roadmap for this talk

- MRoB: Problem definition and previous work
  - Prelude: Steiner forests
  - MRoB definition
- Previous work and our contribution
- Approximating MRoB: Recap of [Gupta et al.’03]
- Our algorithm and its analysis
## Previous Work

<table>
<thead>
<tr>
<th>Reference</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agrawal, Klein, Ravi ’95</td>
<td>Primal-dual 2-apx for Steiner forests</td>
</tr>
<tr>
<td>Goemans, Williamson ’95</td>
<td></td>
</tr>
<tr>
<td>Meyerson, Munagala ’02</td>
<td>O(1)-apx for Single sink Buy-at-Bulk</td>
</tr>
<tr>
<td>Kumar, Gupta, Roughgarden ’02</td>
<td>Primal-dual O(1)-apx for MRoB</td>
</tr>
<tr>
<td>Gupta, Kumar, Roughgarden ’03</td>
<td>Randomized 3.55-apx for SRoB</td>
</tr>
<tr>
<td>Jain, Vazirani ’02</td>
<td>Cost-sharing Mechanism for Steiner trees</td>
</tr>
<tr>
<td>Pal, Tardos ’03</td>
<td>and Single-Sink Rent-or-Buy</td>
</tr>
<tr>
<td>Gupta, Kumar, Pal, Roughgarden ’03</td>
<td>Randomized 12-apx for MRoB</td>
</tr>
<tr>
<td></td>
<td>Analysis can be improved to 8-apx.</td>
</tr>
</tbody>
</table>
Our Contribution

**Theorem:**
There is a $4 + 2\sqrt{2} \approx 6.82$ approximation algorithm for the MRoB problem.

**Main ingredients:**

- Uses existing algorithms for Steiner forests and prize-collecting Steiner tree.
  Implies that dual solution has laminar structure.
- New cost-sharing mechanism
- Simpler analysis
A roadmap for this talk

- MRoB: Problem definition and previous work
- Approaching MRoB: Recap of [Gupta et al.’03]
- Our algorithm and its analysis
Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)

All edges have unit cost and \( M = 2 \)

Mark each demand pair with prob \( 1/M \).
MRoB Algorithm from [Gupta et al. ’03]

- Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)
- All edges have unit cost and \( M = 2 \)

Mark each demand pair with prob 1/M.
Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)

All edges have unit cost and \( M = 2 \)

Mark each demand pair with prob \( 1/M \).

Construct Steiner forest \( F \) on marked pairs.
MRoB Algorithm from [Gupta et al. ’03]

- Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)
- All edges have unit cost and \( M = 2 \)

Mark each demand pair with prob \( 1/M \).
Construct Steiner forest \( F \) on marked pairs.
Example with four terminal pairs: $R = \{(s_i, t_i)\}_{1 \leq i \leq 4}$

All edges have unit cost and $M = 2$

Mark each demand pair with prob $1/M$.

Construct Steiner forest $F$ on marked pairs.

Rent edges on min-cost path in $G|F$ for all unmarked pairs.
Example with four terminal pairs: $R = \{(s_i, t_i)\}_{1 \leq i \leq 4}$

All edges have unit cost and $M = 2$

Mark each demand pair with prob $1/M$.

Construct Steiner forest $F$ on marked pairs.

Rent edges on min-cost path in $G|F$ for all unmarked pairs.
MRoB Algorithm from [Gupta et al. ’03]

- Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)
- All edges have unit cost and \( M = 2 \)

Mark each demand pair with prob \( 1/M \).
Construct Steiner forest \( F \) on marked pairs.
Rent edges on min-cost path in \( G|F \) for all unmarked pairs.

Performance guarantee of algorithm depends on Steiner forest algorithm!
Strict Steiner Forest Algorithms

- Given a Steiner forest instance $I = (G, c, R)$.
- Have an $\alpha$-approximate algorithm $\mathcal{A}$. $\mathcal{A}(I)$ returns
Strict Steiner Forest Algorithms

Given a Steiner forest instance $I = (G, c, R)$.

Have an $\alpha$-approximate algorithm $\mathcal{A}$. $\mathcal{A}(I)$ returns

1. A feasible Steiner forest $F$ with $c(F) \leq \alpha \cdot \text{opt}_I$

$\text{opt}_I$: cost of min-cost feasible Steiner forest for $I$
Strict Steiner Forest Algorithms

- Given a Steiner forest instance $I = (G, c, R)$.
- Have an $\alpha$-approximate algorithm $A$. $A(I)$ returns
  1. A feasible Steiner forest $F$ with $c(F) \leq \alpha \cdot \text{opt}_I$
      $\text{opt}_I$: cost of min-cost feasible Steiner forest for $I$
  2. A cost-share $\chi_R(s, t)$ for each $(s, t) \in R$ such that
      $\sum_{(s, t) \in R} \chi_R(s, t) \leq \text{opt}_I$
Strict Steiner Forest Algorithms

- Given a Steiner forest instance \( I = (G, c, R) \).
- Have an \( \alpha \)-approximate algorithm \( \mathcal{A} \). \( \mathcal{A}(I) \) returns
  1. A feasible Steiner forest \( F \) with \( c(F) \leq \alpha \cdot \text{opt}_I \)
     \( \text{opt}_I \): cost of min-cost feasible Steiner forest for \( I \)
  2. A cost-share \( \chi_R(s, t) \) for each \((s, t) \in R\) such that \( \sum_{(s, t) \in R} \chi_R(s, t) \leq \text{opt}_I \)
- Notation: \((F, \chi_R) \leftarrow \mathcal{A}(G, c, R)\)
Strict Steiner Forest Algorithms

$G|F$: Graph obtained from contracting edges in $F$
Strict Steiner Forest Algorithms

- $G | F$: Graph obtained from contracting edges in $F$
- Let $(s, t) \in R$ and let

$$F_{st} \leftarrow A(G, c, R \setminus \{(s, t)\})$$
Strict Steiner Forest Algorithms

- $G|F$: Graph obtained from contracting edges in $F$
- Let $(s, t) \in R$ and let
  \[
  F_{st} \leftarrow \mathcal{A}(G, c, R \setminus \{(s, t)\})
  \]
- $\mathcal{A}$ is $\beta$-strict if
  \[
  c_{G|F_{st}}(s, t) \leq \beta \cdot \chi_R(s, t)
  \]
  for all $(s, t) \in R$. 


Strict Steiner Forest Algorithms

- $G|F$: Graph obtained from contracting edges in $F$
- Let $(s, t) \in R$ and let
  \[
  F_{st} \leftarrow A(G, c, R \setminus \{(s, t)\})
  \]
- $A$ is $\beta$-strict if
  \[
  c_{G|F_{st}}(s, t) \leq \beta \cdot \chi_R(s, t)
  \]
  for all $(s, t) \in R$.
  In other words: $\beta \cdot \chi_R(s, t)$ pays $s, t$-path in $G|F_{st}$
Strictness: Example

- Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)
- All edges have unit cost and \( M = 2 \)

Compute Steiner forest and cost-shares:
Strictness: Example

- Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)
- All edges have unit cost and \( M = 2 \)

Compute Steiner forest and cost-shares:
\[
\sum_{i=1}^{4} \chi_{R}(s_i, t_i) = 4 = \text{opt}
\]
Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)

All edges have unit cost and \( M = 2 \)

Compute Steiner forest and cost-shares:
\[
\sum_{i=1}^{4} \chi_R(s_i, t_i) = 4 = \text{opt}
\]

Strictness for \((s_2, t_2)\). Compute Steiner forest \( F_2 \) for \( R \setminus \{(s_2, t_2)\} \).
**Strictness: Example**

- Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)
- All edges have unit cost and \( M = 2 \)

Compute Steiner forest and cost-shares:
\[
\sum_{i=1}^{4} \chi_{R}(s_i, t_i) = 4 = \text{opt}
\]

Strictness for \((s_2, t_2)\). Compute Steiner forest \( F_2 \) for \( R \setminus \{(s_2, t_2)\} \).
Strictness: Example

- Example with four terminal pairs: \( R = \{(s_i, t_i)\}_{1 \leq i \leq 4} \)
- All edges have unit cost and \( M = 2 \)

Compute Steiner forest and cost-shares:
\[
\sum_{i=1}^{4} \chi_R(s_i, t_i) = 4 = \text{opt}
\]

Strictness for \((s_2, t_2)\). Compute Steiner forest \( F_2 \) for \( R \setminus \{(s_2, t_2)\} \).
\[
c_{G|F_2}(s_2, t_2) = 1 \leq \chi_R(s_2, t_2)
\]
Strictness and MRoB

**MRoB-Theorem:** [Gupta et al. ’03]

An $\alpha$-approximate and $\beta$-strict approximation algorithm for the Steiner forest problem implies an $(\alpha + \beta)$-approximation for MRoB.
A roadmap for this talk

- MRoB: Problem definition and previous work
- Approximating MRoB: Recap of [Gupta et al.’03]
- Our algorithm and its analysis
  - Primal-Dual algorithm for Steiner forest
    - Strictness: Some intuition
    - Our Algorithm
Towards an LP formulation

A Steiner cut is a set $U \subseteq V$ that separates at least one terminal pair.

$\mathcal{U}$ is the set of all Steiner cuts:

$$\mathcal{U} = \{U \subseteq V : \exists (s,t) \in R : |\{s,t\} \cap U| = 1\}$$
Steiner forests: \textbf{Primal and dual LP’s}

\textbf{minimize} \quad \sum_{e \in E} c_e \cdot x_e \quad \quad (P)

\textbf{s.t.} \quad \sum_{e \in \delta(U)} x_e \geq 1 \quad \forall U \in \mathcal{U}

\quad x_e \geq 0 \quad \forall e \in E

\textbf{maximize} \quad \sum_{U \in \mathcal{U}} y_U \quad \quad (D)

\textbf{s.t.} \quad \sum_{U : e \in \delta(U)} y_U \leq c_e \quad \forall e \in E

\quad y_U \geq 0 \quad \forall U \in \mathcal{U}
Steiner trees: Moats

Can think of $y_U$ as moat around $U$ of radius $y_U$. Example: Steiner cuts $U$ and $W$, min-cost $U, W$-path has cost 4.

\[ y_U = y_W = 2 \]
Algorithm $S_F$ ([AKR '95],[GW '95]) construct primal and dual solution at the same time.
Algorithm $S_F$ ([AKR '95],[GW '95]) construct primal and dual solution at the same time.
Algorithm $SF$ ([AKR '95], [GW '95]) construct primal and dual solution at the same time.
Steiner Forests: PD-Algorithm $SF$

Algorithm $SF$ ([AKR ’95], [GW ’95]) construct primal and dual solution at the same time.
Algorithm $SF$ ([AKR '95],[GW '95]) construct primal and dual solution at the same time.
Properties of $\text{SF}$

- Let $F$ be the Steiner forest returned by $\text{SF}(R, c, G)$
- Consider tree $T$ in $F$
  - $\mathcal{U}_T$: Moats grown by $\text{SF}$ around active subsets of $T$
  - $\text{age}_T$: Time when $T$ dies in $\text{SF}$
Properties of $\text{SF}$

- Let $F$ be the Steiner forest returned by $\text{SF}(R, c, G)$
- Consider tree $T$ in $F$
  - $\mathcal{U}_T$: Moats grown by $\text{SF}$ around active subsets of $T$
  - $\text{age}_T$: Time when $T$ dies in $\text{SF}$

**Theorem:** [AKR ’95]

For all trees $T$ in $F$ we must have

$$\sum_{e \in T} c_e \leq \left( 2 \cdot \sum_{U \in \mathcal{U}_T} y_U \right) - 2 \cdot \text{age}_T$$

**Consequence:** $c(F') \leq 2 \cdot \text{opt}$
Computing Cost-Shares

Let $\text{age}_s = \text{age}_t$ be time when $s$ meets $t$ in SF.
Computing Cost-Shares

- Let $\text{age}_s = \text{age}_t$ be time when $s$ meets $t$ in SF
- $\chi_R(s) =$ total time in SF where $s$ is terminal of maximum age in its moat
  (break ties arbitrarily)
Computing Cost-Shares

- Let \( \text{age}_s = \text{age}_t \) be time when \( s \) meets \( t \) in \( SF \)
- \( \chi_R(s) = \) total time in \( SF \) where \( s \) is terminal of maximum age in its moat
  (break ties arbitrarily)
- \( \chi_R(s, t) = \chi_R(s) + \chi_R(t) \)
Computing Cost-Shares

Let \( \text{age}_s = \text{age}_t \) be time when \( s \) meets \( t \) in \( SF \)

\( \chi_R(s) = \) total time in \( SF \) where \( s \) is terminal of maximum age in its moat
(break ties arbitrarily)

\( \chi_R(s, t) = \chi_R(s) + \chi_R(t) \)

\[ \sum_{(s, t) \in R} \chi_R(s, t) = \sum_{U \in \mathcal{U}} y_U \leq \text{opt } I \]
A roadmap for this talk

- MRoB: Problem definition and previous work
- Approximating MRoB: Recap of [Gupta et al.'03]
- Our algorithm and its analysis
  - Primal-Dual algorithm for Steiner forest
  - Strictness: Some intuition
  - Our Algorithm
**Strictness: Intuition**

\[ F_{st} \leftarrow SF(G, c, R \setminus \{(s, t)\} \]

- \( U_1 \) and \( U_2 \) are outermost moats around trees in \( F_{st} \)
Strictness: Intuition

\[ F_{st} \leftarrow SF(G, c, R \setminus \{(s, t)\}) \]

- \( U_1 \) and \( U_2 \) are outermost moats around trees in \( F_{st} \)
- \( s \) and \( t \) collect cost-share for black parts of path but not for red parts!
Strictness: Intuition

\[ F_{st} \leftarrow SF(G, c, R \setminus \{(s, t)\}) \]

- \( U_1 \) and \( U_2 \) are outermost moats around trees in \( F_{st} \)
- \( s \) and \( t \) collect cost-share for black parts of path but not for red parts!
- Idea: Can we force the black parts to be at least as long as the red parts? \( 2 \)-Strictness?
Strictness: Intuition

Consider outermost moat $U$ of $F_{st}$ that is on $s, t$-path

$h_s + h_t$: Parts of $U$ that feel dual in $SF(G, c, R \setminus \{(s, t)\})$

$b_s^0 + b_t^0$: Cost-share of path $P$ reserved for moat $U$

Achieve $\beta$-strictness by proving

$$b_t + b_s \leq \beta \cdot (b_t^0 + b_s^0)$$

for all components $U$ on path $P$
Intuition: Reserving Cost-Share

Suppose component $U$ has just died in SF
Intuition: Reserving Cost-Share

Suppose component $U$ has just died in SF.

Define budget of $U$ as

$$b_U^0 = \gamma \cdot \sum_{S \in U_U} y_S$$

for some parameter $\gamma \geq .5$.
Intuition: Reserving Cost-Share

Suppose component $U$ has just died in SF

Define budget of $U$ as

$$b_U^0 = \gamma \cdot \sum_{S \in \mathcal{U}_U} y_S$$

for some parameter $\gamma \geq 0.5$

Keep on growing $U$ and consume $b_U^0$ at the rate of growth.
Intuition: Reserving Cost-Share

- Suppose component $U$ has just died in SF
- Define budget of $U$ as
  
  $$b^0_U = \gamma \cdot \sum_{S \in \mathcal{U}_U} y_S$$

  for some parameter $\gamma \geq 0.5$
- Keep on growing $U$ and consume $b^0_U$
  at the rate of growth.

Cost-share reservation
A roadmap for this talk

- MRoB: Problem definition and previous work
- Approximating MRoB: Recap of [Gupta et al.'03]
- Our algorithm and its analysis
  - Primal-Dual algorithm for Steiner forest
  - Strictness: Some intuition
- Our Algorithm
Algorithm

- Algorithm works like $SF$:
  - Grow moats around active connected components
  - Merge two moats $U_1$ and $U_2$ when a path between them becomes tight
  - A moat is dead when it does not separate terminal pairs
Algorithm

Algorithm works like SF:
- Grow moats around active connected components
- Merge two moats $U_1$ and $U_2$ when a path between them becomes tight
- A moat is dead when it does not separate terminal pairs

**Crucial difference:** Grow a dead moat $U$ and increase its spent budget $b_U$ at the same speed.

Continue as long as $b_U \leq \gamma \cdot \sum_{S \in U} y_S$
Algorithm

- Algorithm works like SF:
  - Grow moats around active connected components
  - Merge two moats $U_1$ and $U_2$ when a path between them becomes tight
  - A moat is dead when it does not separate terminal pairs

- Crucial difference: Grow a dead moat $U$ and increase its spent budget $b_U$ at the same speed.

- Continue as long as $b_U \leq \gamma \cdot \sum_{S \in U} y_S$

- Initially let $b_U = 0$ for all $U \subseteq V$
Algorithm

- Algorithm works like SF:
  - Grow moats around active connected components
  - Merge two moats $U_1$ and $U_2$ when a path between them becomes tight
  - A moat is dead when it does not separate terminal pairs

- Crucial difference: Grow a dead moat $U$ and increase its spent budget $b_U$ at the same speed.

- Continue as long as $b_U \leq \gamma \cdot \sum_{S \in \mathcal{U}_U} y_S$

- Initially let $b_U = 0$ for all $U \subseteq V$

- When two moats $U_1$ and $U_2$ merge, let $b_{U_1 \cup U_2} = b_{U_1} + b_{U_2}$. 
Cost of the final forest

[AKR ’95] immediately implies

\[ c(F') \leq 2 \cdot \sum_{U \subseteq V} y_U \leq (2 + 2\gamma) \cdot \sum_{U \in \mathcal{U}} y_U \leq (2 + 2\gamma) \cdot \text{opt}_I \]
The Strictness of the algorithm

Consider outermost moat $M$ of $F_{st}$ that is on $s,t$-path

\[ b_s^0 \quad h_s \quad v_1 \quad v_2 \quad h_t \quad b_t^0 \]

- $\gamma$ controls $b_s^0$ and $b_t^0$
- Can show that $\gamma \geq 1 / (\beta - 2)$ suffices to obtain

\[ b_t + b_s \leq \beta \cdot (b_t^0 + b_s^0) \]

for all components $U$ on path $P$
Putting things together

- **Cost:** The final forest $F$ costs at most 
  $$(2 + 2\gamma) \cdot \text{opt } I \leq (2 + 2/(\beta - 2)) \cdot \text{opt } I$$

- **Strictness:** The algorithm is $\beta$-strict

- **MRoB-Theorem $\implies$ Algorithm is**
  $$(2 + 2/(\beta - 2) + \beta)\text{-approximation for MRoB.}$$

Choosing $\beta = 2 + \sqrt{2}$ yields 
$$(2 + 2/(\beta - 2) + \beta) = 4 + 2\sqrt{2}.$$
Open issues

- Close gap between SRoB and MRoB: 3.55 vs 6.82. Our algorithm is 5-approximate in some special cases!

- Is $\beta$-Strictness too a strong property? Can the MRoB analysis in [Gupta et al. ’03] be improved by a using a weakly-strict algorithm $A$?

For $R' \subseteq R$ let $F_{R'}$ be the Steiner forest computed by $A(G, c, R \setminus R')$.

Require

$$c_{G|F_{R'}}(R') \leq \beta \cdot \sum_{(s,t) \in R'} \chi_R(s, t)$$