Bounded Degree Closest k-Tree Power is NP-Complete

Michael Dom, Jiong Guo, and Rolf Niedermeier

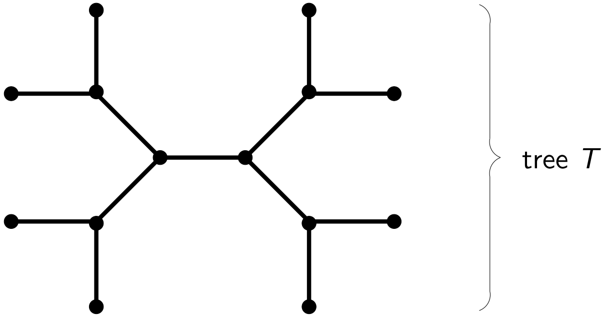
Friedrich-Schiller-Universität Jena, Germany

Structure of the Talk

- **Introduction: Tree power and tree root problems**
- A reduction from 3-Vertex Cover to Closest k-Tree Power with Bounded Degree

k-Tree Roots and k-Tree Powers

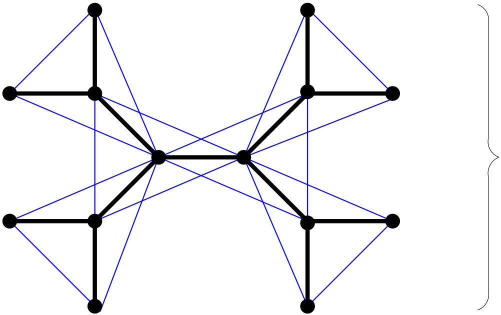
Example: A tree network consisting of 14 processors. Passing information from one processor to the next requires one timestep.



Black edges: Communication possible in one timestep

k-Tree Roots and k-Tree Powers

Example: A tree network consisting of 14 processors. Passing information from one processor to the next requires one timestep.

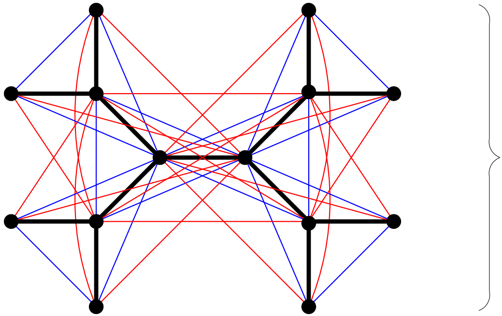


2-tree power T^2

Black edges: Communication possible in one timestep
Blue edges: Communication possible in two timesteps

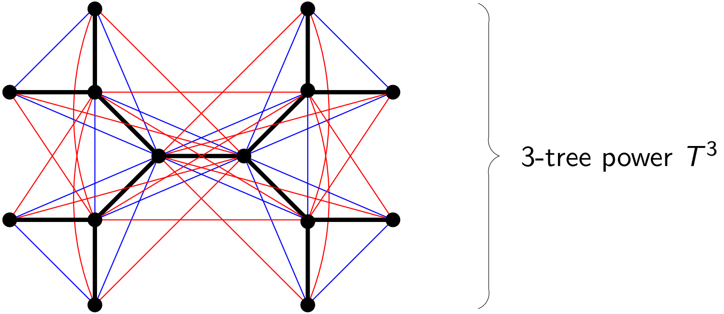
k-Tree Roots and k-Tree Powers

Example: A tree network consisting of 14 processors. Passing information from one processor to the next requires one timestep.



Black edges: Communication possible in one timestep
Blue edges: Communication possible in two timesteps
Red edges: Communication possible in three timesteps

k-Tree Roots and k-Tree Powers



Definition

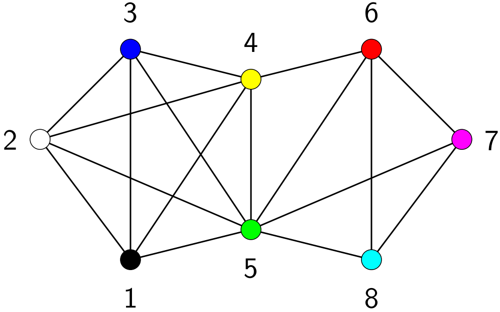
A graph $G = (V, E_G)$ is a *k-tree power* if there is a tree $T = (V, E_T)$ with

$$\forall u, v \in V : \; \mathrm{dist}_T \leq k \;\Leftrightarrow\; (u, v) \in E_G.$$

T is called the *k-tree root* of G.

Tree power recognition / Computing tree roots

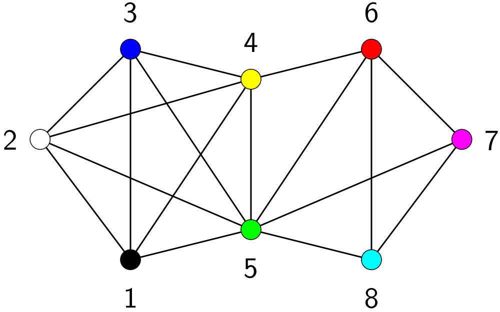
Example: Constructing phylogenies.
Given: A graph with node set V—denoting biological species—and an edge set denoting similarities between the species.
Question: Is there a tree with node set V in which two nodes u, v have distance at most k iff they are connected in the given graph?



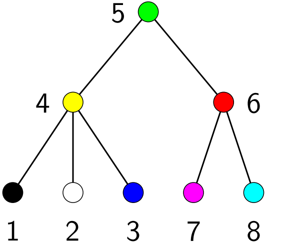
Set of species—edges denote similarities

Tree power recognition / Computing tree roots

Example: Constructing phylogenies.
Given: A graph with node set V—denoting biological species—and an edge set denoting similarities between the species.
Question: Is there a tree with node set V in which two nodes u, v have distance at most k iff they are connected in the given graph?



Set of species—edges denote similarities



Phylogeny: 2-tree root

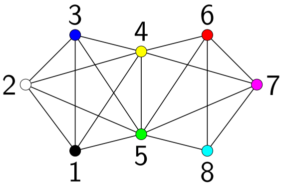
Tree power recognition / Computing tree roots

k-Tree Power
Instance: A graph G.
Question: Is there a tree T with $T^k = G$?

- Solvable in linear time for $k = 2$
  [Y. L. Lin, S. S. Skiena, *SIAM Journal on Discrete Mathematics*, 1995]
- Solvable in polynomial time for every $k \geq 3$
  [P. E. Kearney, D. G. Corneil, *Journal of Algorithms*, 1998]
- NP-complete for every $k \geq 2$ if T may be an arbitrary graph (Graph Power problem)
  [R. Motwani, M. Sudan, *Discrete Applied Mathematics*, 1994]

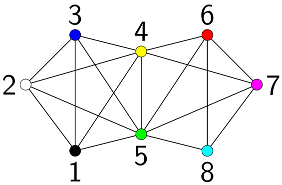
A graph modification problem

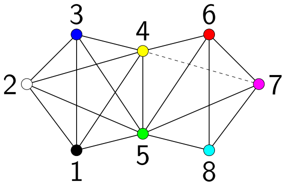
What to do if a given graph has no k-tree root?

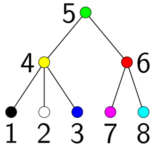


A graph modification problem

What to do if a given graph has no *k*-tree root?

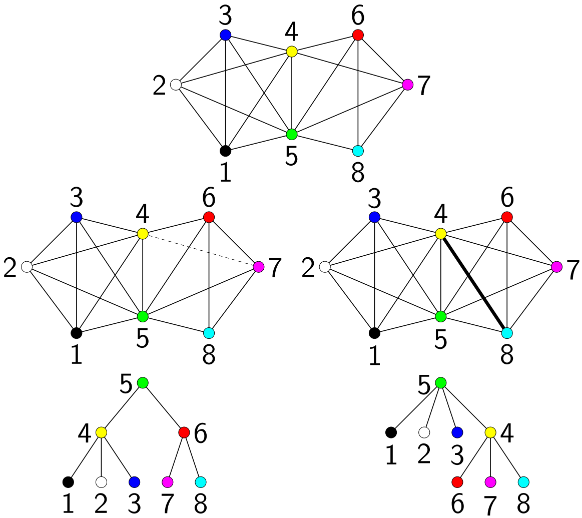






A graph modification problem

What to do if a given graph has no k-tree root?



A graph modification problem

CLOSEST k-TREE POWER (CTPk)
Instance: A graph G.
Question: Is there a k-tree power T^k such that T^k and G differ by at most ℓ edges: $|(E(T^k) \setminus E(G)) \cup (E(G) \setminus E(T^k))| \leq \ell$?

- NP-complete for every $k \geq 2$
  [P. E. Kearney, D. G. Corneil, *Journal of Algorithms*, 1998]

CLOSEST k-TREE POWER WITH BOUNDED DEGREE (Δ-CTPk)
The maximum vertex degree of T is at most Δ.

- Complexity open so far for every $k, \Delta \geq 2$. Open question at COCOON 2004! [T. Tsukiji, Z.-Z. Chen, *Proc. 10th COCOON*, 2004]

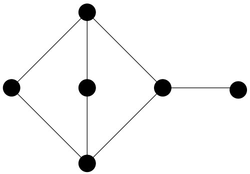
Structure of the Talk

- Introduction: Tree power and tree root problems
- **A reduction from 3-Vertex Cover to Closest k-Tree Power with Bounded Degree**

Idea of the reduction

We reduce the NP-complete [M. R. Garey, D. S. Johnson, L. J. Stockmeyer, *Theoretical Computer Science*, 1976] problem 3-VERTEX COVER to Δ-CTPk with $\Delta \geq 4$.

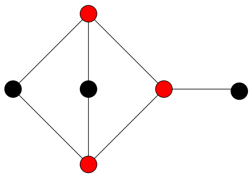
3-VERTEX COVER
Input: A graph $G = (V, E)$ with a maximum vertex degree 3 and a nonnegative integer ℓ.
Question: Is there a set $C \subseteq V$ of at most ℓ vertices such that each edge from E has at least one endpoint in C?



Idea of the reduction

We reduce the NP-complete [M. R. Garey, D. S. Johnson, L. J. Stockmeyer, *Theoretical Computer Science*, 1976] problem 3-VERTEX COVER to Δ-CTPk with $\Delta \geq 4$.

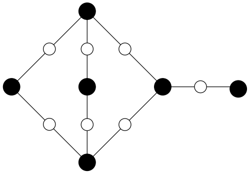
3-VERTEX COVER
Input: A graph $G = (V, E)$ with a maximum vertex degree 3 and a nonnegative integer ℓ.
Question: Is there a set $C \subseteq V$ of at most ℓ vertices such that each edge from E has at least one endpoint in C?



“Edge modification version” of 3-Vertex Cover

Reformulation of 3-Vertex Cover:

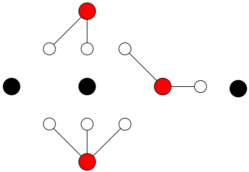
- We insert an additional node in the middle of each edge.
- The question now is: Delete edges such that
  1. each “additional” node is adjacent to exactly one “original” vertex, and
  2. the number of “original” vertices having degree at least one is minimized.



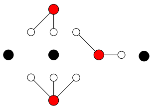
“Edge modification version” of 3-VERTEX COVER

Reformulation of 3-VERTEX COVER:

- We insert an additional node in the middle of each edge.
- The question now is: Delete edges such that
  1. each “additional” node is adjacent to exactly one “original” vertex, and
  2. the number of “original” vertices having degree at least one is minimized.

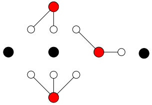


“Edge modification version” of 3-VERTEX COVER



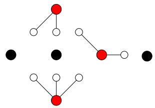
- We need $|E|$ edge deletions to transform each “additional” node into a degree-1-node—independent of the number of red vertices.

"Edge modification version" of 3-VERTEX COVER



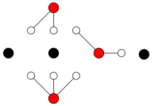
- We need $|E|$ edge deletions to transform each "additional" node into a degree-1-node—independent of the number of red vertices.
- Δ-$\mathrm{CTP}k$ minimizes only the number of modified edges, but not the number of red vertices.

“Edge modification version” of 3-Vertex Cover



- We need $|E|$ edge deletions to transform each “additional” node into a degree-1-node—independent of the number of red vertices.
- Δ-$\mathrm{CTP}k$ minimizes only the number of modified edges, but not the number of red vertices.
- $\Rightarrow$ Replace the “original” vertices of the 3-Vertex Cover instance by gadgets such that for each red vertex further edge modifications are necessary.

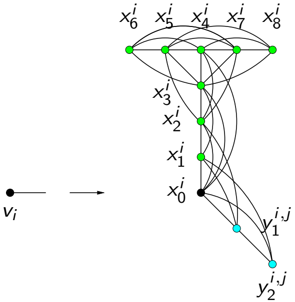
“Edge modification version” of 3-VERTEX COVER



- We need $|E|$ edge deletions to transform each “additional” node into a degree-1-node—independent of the number of red vertices.
- Δ-$\mathrm{CTP}k$ minimizes only the number of modified edges, but not the number of red vertices.
- $\Rightarrow$ Replace the “original” vertices of the 3-VERTEX COVER instance by gadgets such that for each red vertex further edge modifications are necessary.
- Replace the edges of the 3-VERTEX COVER instance (and the “additional” nodes) by gadgets.

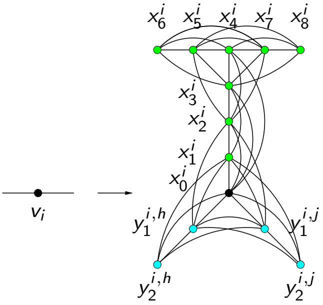
Construction of a vertex gadget (case $k = 3$)

For each "original" vertex $v_i \in V$, $i = 1, \ldots, n$ of the 3-Vertex Cover instance $G = (V, E)$ we insert into the Δ-CTPk instance G_{CTP} a *vertex gadget*:



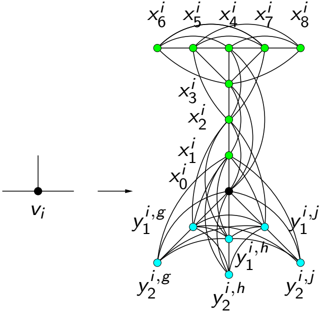
Construction of a vertex gadget (case $k = 3$)

For each "original" vertex $v_i \in V$, $i = 1, \ldots, n$ of the 3-VERTEX COVER instance $G = (V, E)$ we insert into the Δ-CTPk instance G_{CTP} a *vertex gadget*:



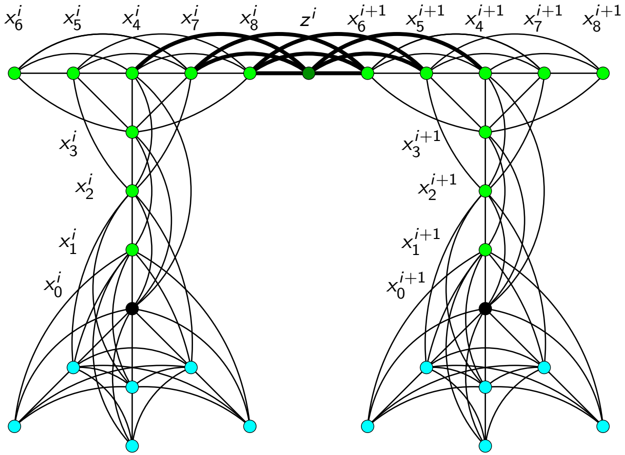
Construction of a vertex gadget (case $k = 3$)

For each "original" vertex $v_i \in V$, $i = 1, \ldots, n$ of the 3-Vertex Cover instance $G = (V, E)$ we insert into the Δ-CTPk instance G_{CTP} a *vertex gadget*:



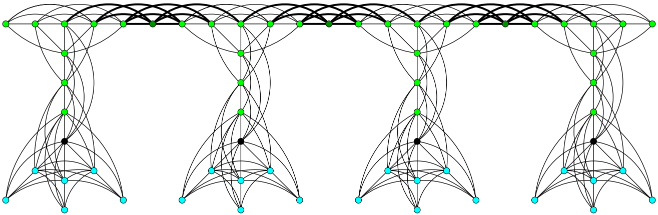
Making G_{CTP} a connected component (case $k = 3$)

To guarantee that G_{CTP} is connected we add $n - 1$ *connecting nodes* $z^i, \ldots, z^{n-1}$, and for all $1 \leq i < n$ we connect the gadgets of v_1 and v_{i+1}:



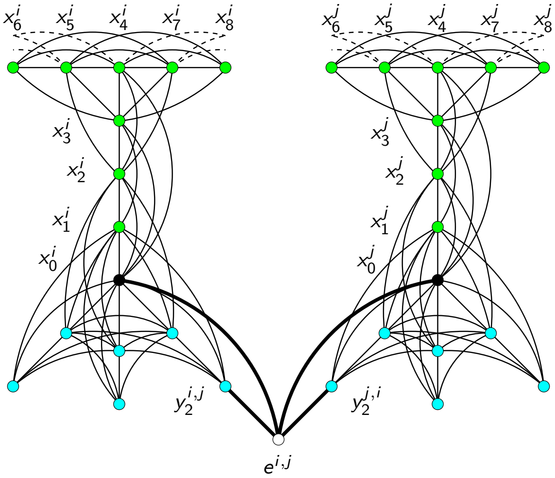
Making G_{CTP} a connected component (case $k = 3$)

To guarantee that G_{CTP} is connected we add $n-1$ *connecting nodes* $z^i, \ldots, z^{n-1}$, and for all $1 \leq i < n$ we connect the gadgets of v_1 and v_{i+1}:

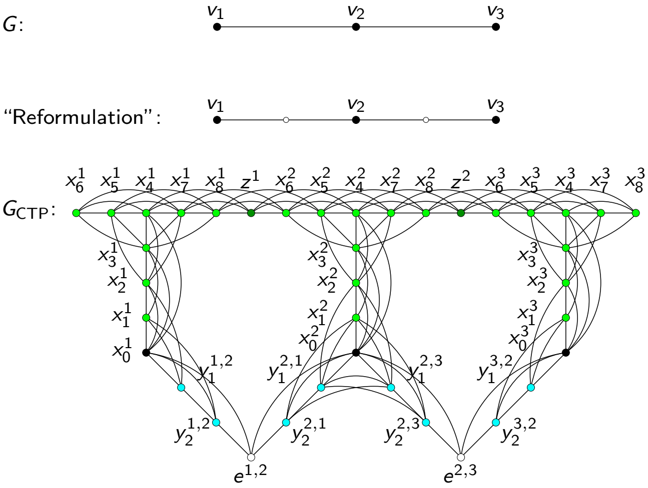


Construction of an edge gadget (case $k = 3$)

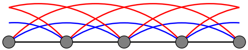
For each edge (v_i, v_j) of the 3-Vertex Cover instance $G = (V, E)$ we insert into the Δ-CTPk instance G_{CTP} an *edge gadget* consisting of an *edge node* $e^{i,j}$ and four edges:



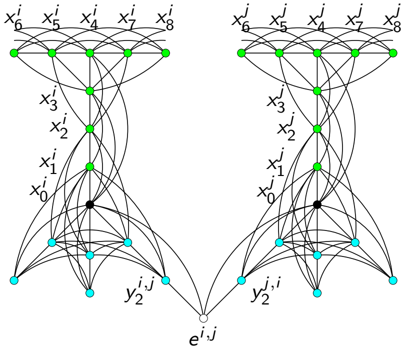
Example (case $k = 3$)



Observation for edge gadgets (case $k = 3$)

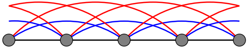
Structure of a 3-tree power: 

In order to obtain a 3-tree power, we have to insert one edge and to delete two edges in G_{CTP} for every edge in G:

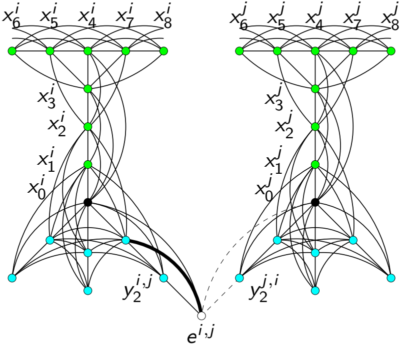


Altogether: $3 \cdot |E|$ edge modifications.

Observation for edge gadgets (case $k = 3$)

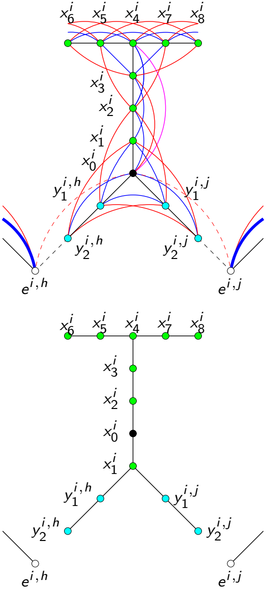
Structure of a 3-tree power: 

In order to obtain a 3-tree power, we have to insert one edge and to delete two edges in G_{CTP} for every edge in G:

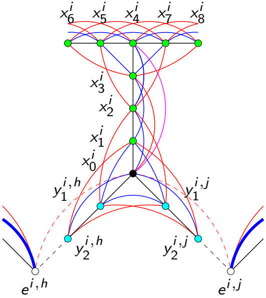


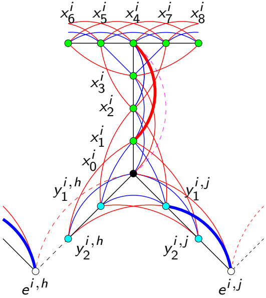
Altogether: $3 \cdot |E|$ edge modifications.

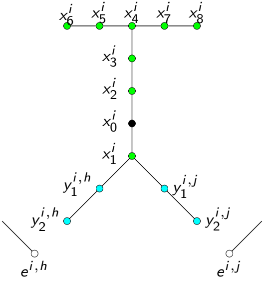
A detailed look on vertex gadgets (case $k = 3$)

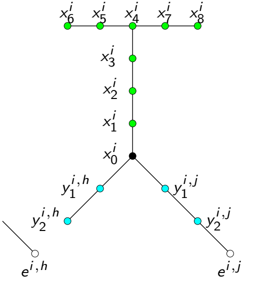


A detailed look on vertex gadgets (case $k = 3$)

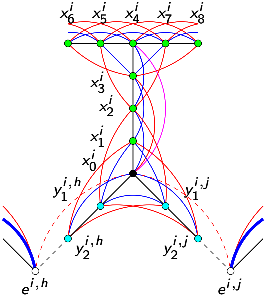






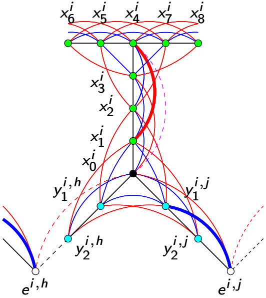


A detailed look on vertex gadgets (case $k = 3$)



Vertex gadget connected to no edge node:

No edge modification in the vertex gadget

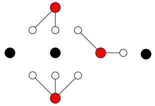


Vertex gadget connected to at least one edge node:

Two edge modifications in the vertex gadget

Counting the edge modifications

Altogether:
#modified edges $=$
$3 \cdot |E| + 2 \cdot$ #vertex gadgets connected to edge nodes



$\Rightarrow$ Number of vertex gadgets corresponding to red vertices are minimized.

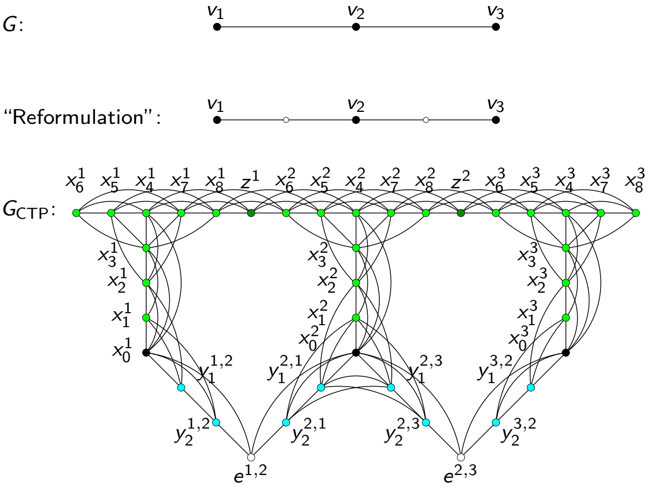
Theorem

G has a vertex cover of size x

$\Leftrightarrow$

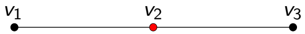
G_{CTP} has a solution of size $3 \cdot |E| + 2 \cdot x$

Example (case $k = 3$)

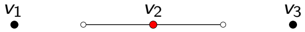


Example (case $k = 3$)

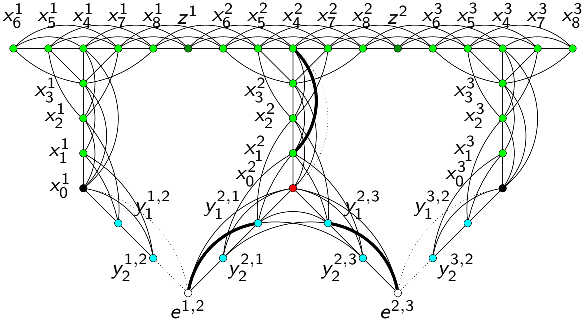
G:



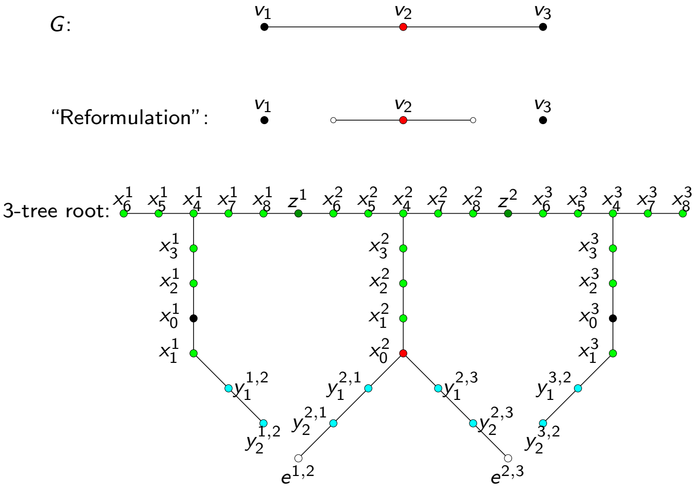
"Reformulation":



Solution:



Example (case $k = 3$)



Open questions

- NP-completeness is only shown for $\Delta \geq 4$. What about $\Delta = 3$?
- What about the hardness if only edge deletions/insertions are allowed?
- Approximation or fixed-parameter tractability results for (Δ)-$\mathrm{CTP}k$?