Tree-Constrained Matching

Julián Mestre (University of Sydney)

Joint work with:

S. Canzar (CWI)
K. Elbassioni (MPI-INF)
G. Klau (CWI)

To appear in ICALP 2011

THE UNIVERSITY OF SYDNEY

Analysis of Live Cell Video

Analysis of Live Cell Video

Given live cell video, we want to track individual cells

Analysis of Live Cell Video

Given live cell video, we want to track individual cells

Mosig et al. introduce two-step approach:

Analysis of Live Cell Video

Given live cell video, we want to track individual cells

Mosig et al. introduce two-step approach:

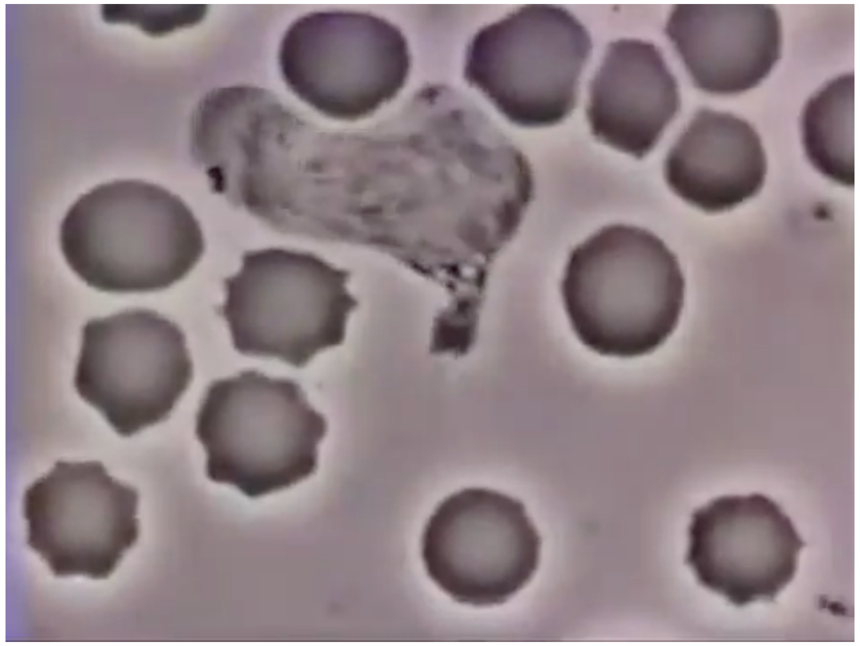
- Perform image segmentation for each frame

Analysis of Live Cell Video

Given live cell video, we want to track individual cells

Mosig et al. introduce two-step approach:

- Perform image segmentation for each frame

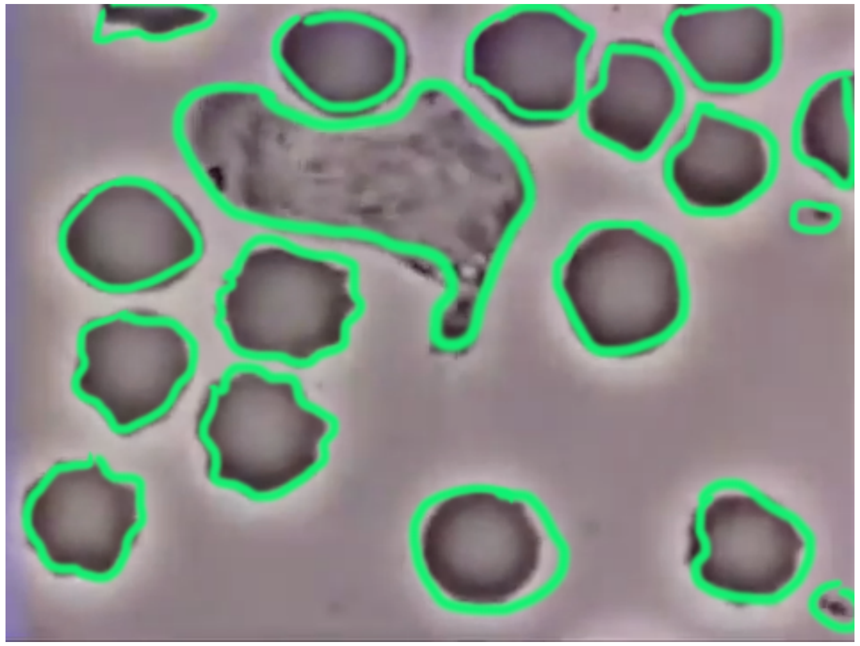


Analysis of Live Cell Video

Given live cell video, we want to track individual cells

Mosig et al. introduce two-step approach:

- Perform image segmentation for each frame

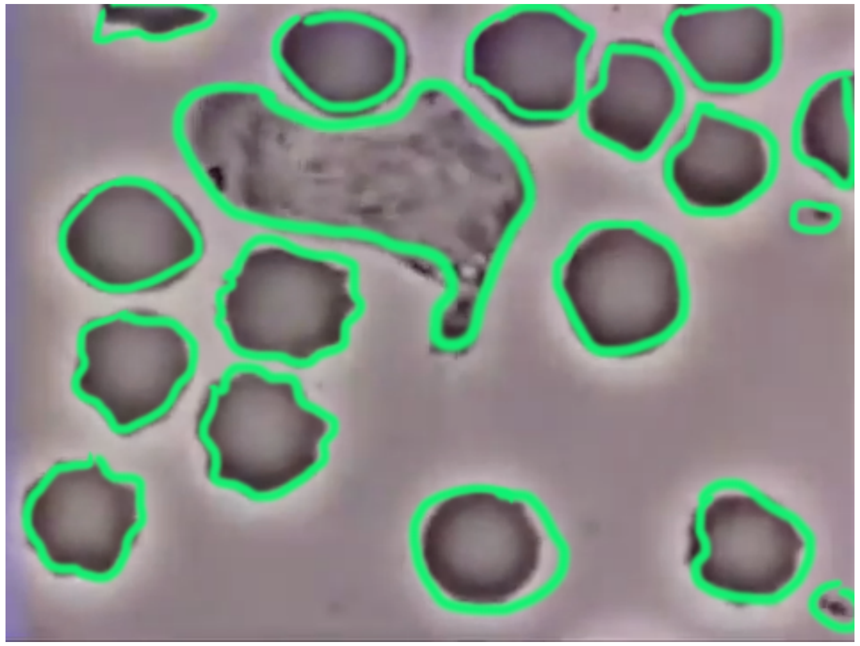


Analysis of Live Cell Video

Given live cell video, we want to track individual cells

Mosig et al. introduce two-step approach:

- Perform image segmentation for each frame
- Match segments from adjacent frames

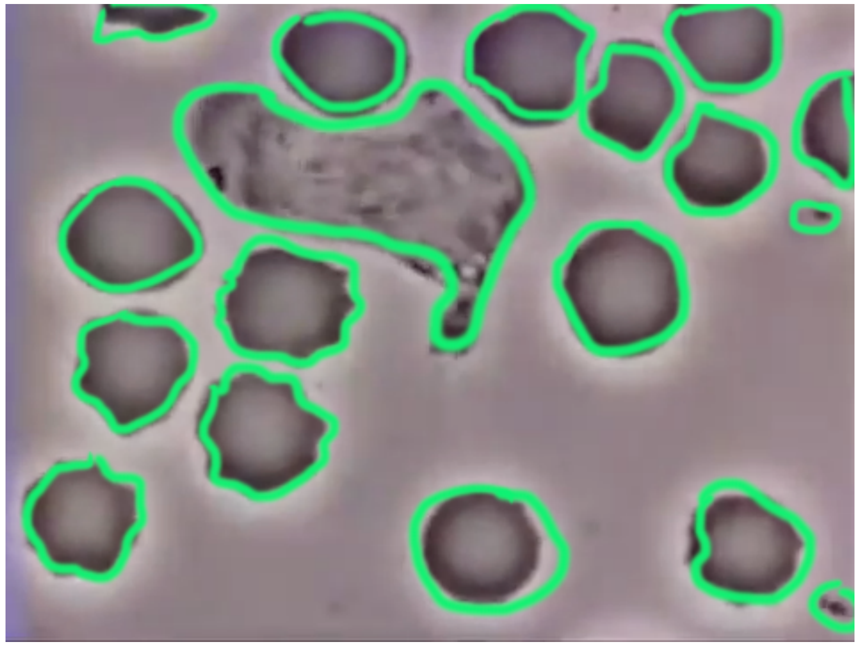


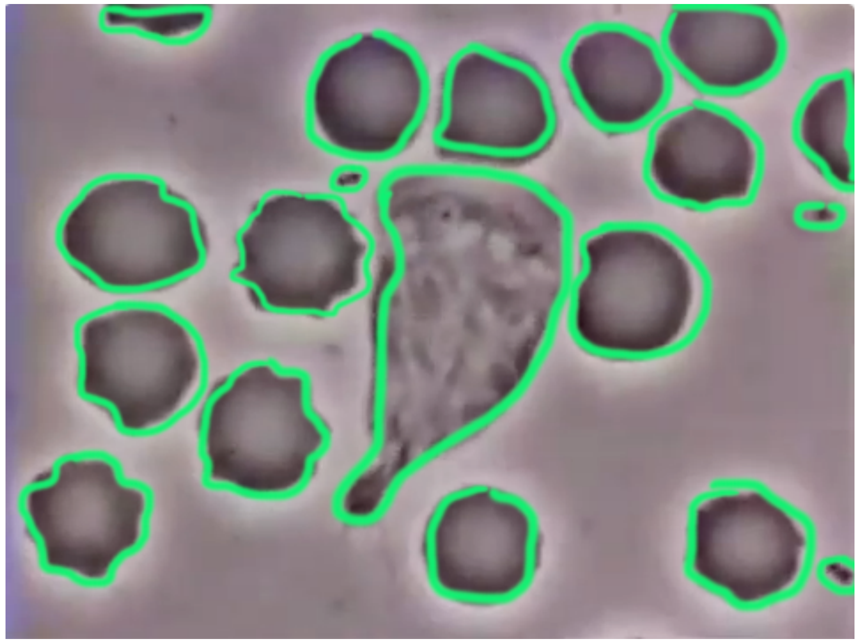
Analysis of Live Cell Video

Given live cell video, we want to track individual cells

Mosig et al. introduce two-step approach:

- Perform image segmentation for each frame
- Match segments from adjacent frames



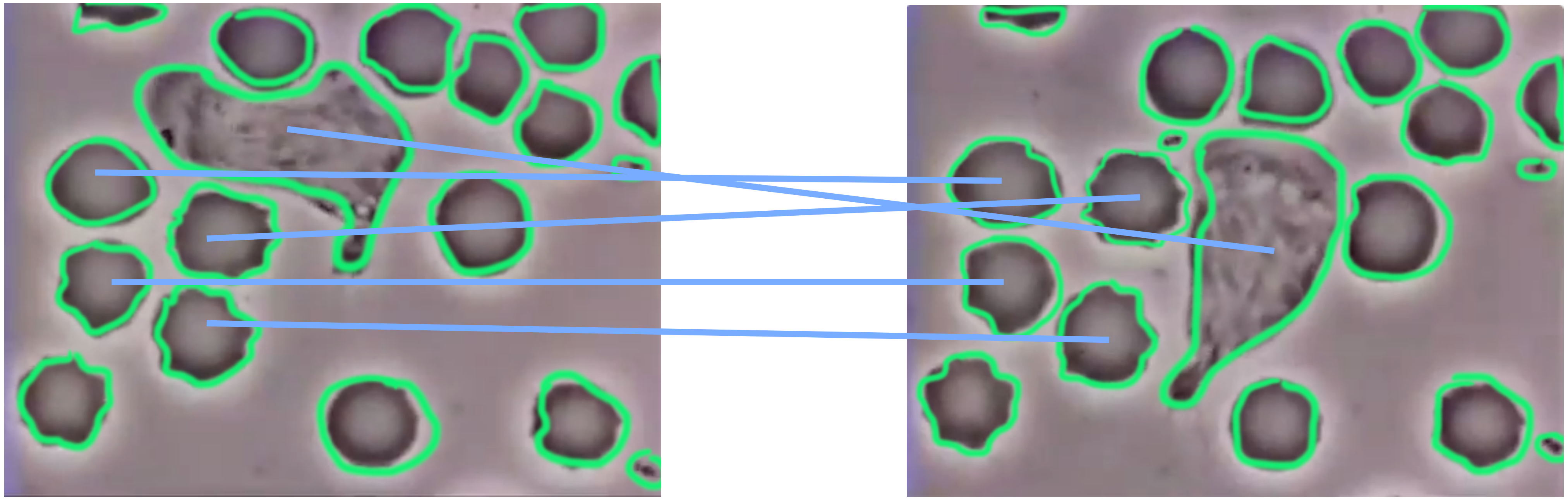


Analysis of Live Cell Video

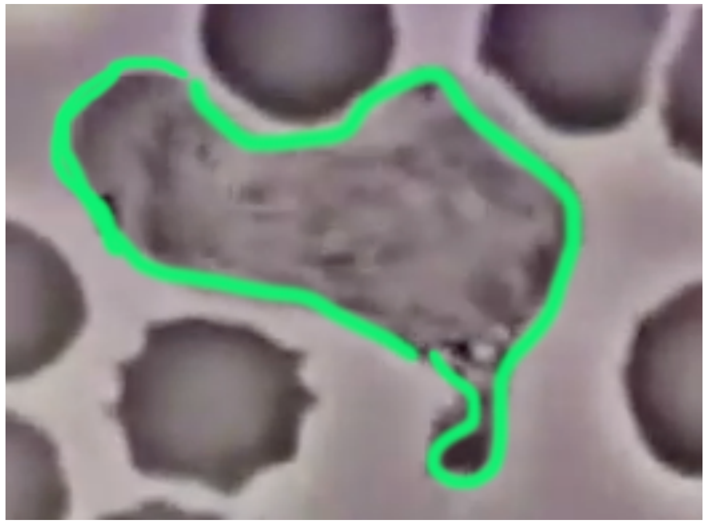
Given live cell video, we want to track individual cells

Mosig et al. introduce two-step approach:

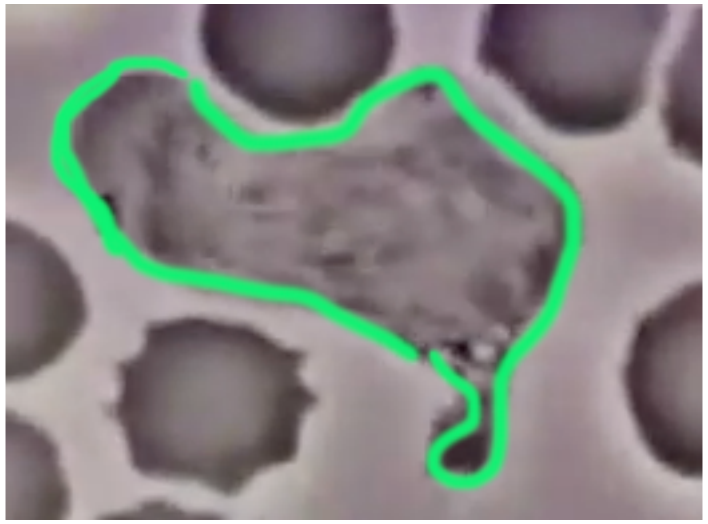
- Perform image segmentation for each frame
- Match segments from adjacent frames



Over/under segmentation

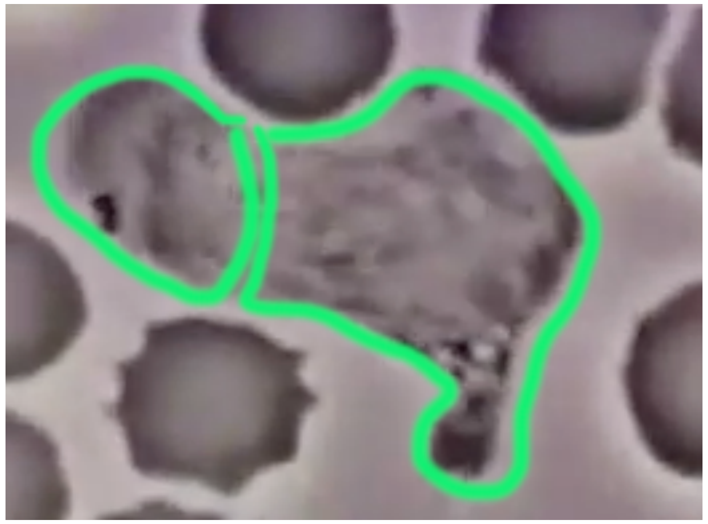


Over/under segmentation

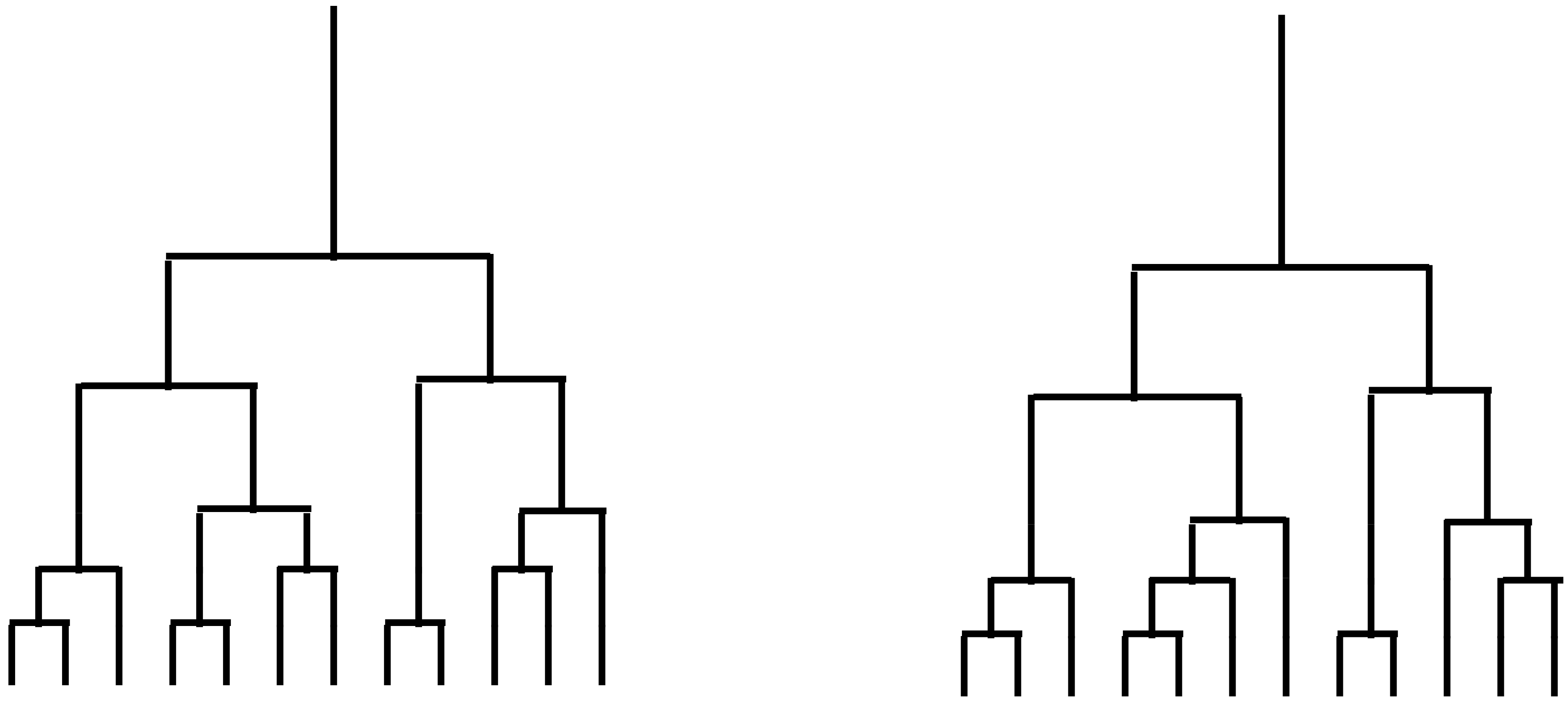


Over/under segmentation

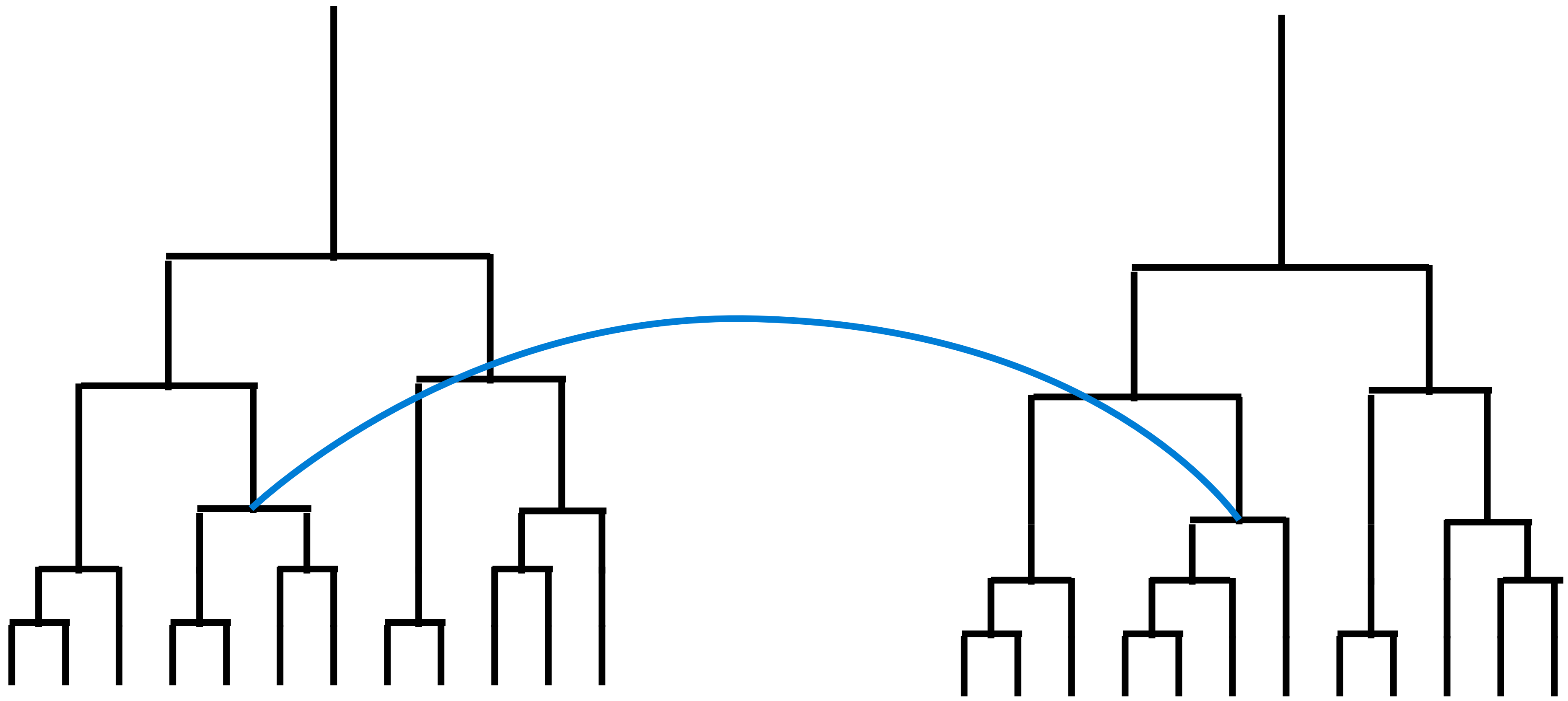
Over/under segmentation

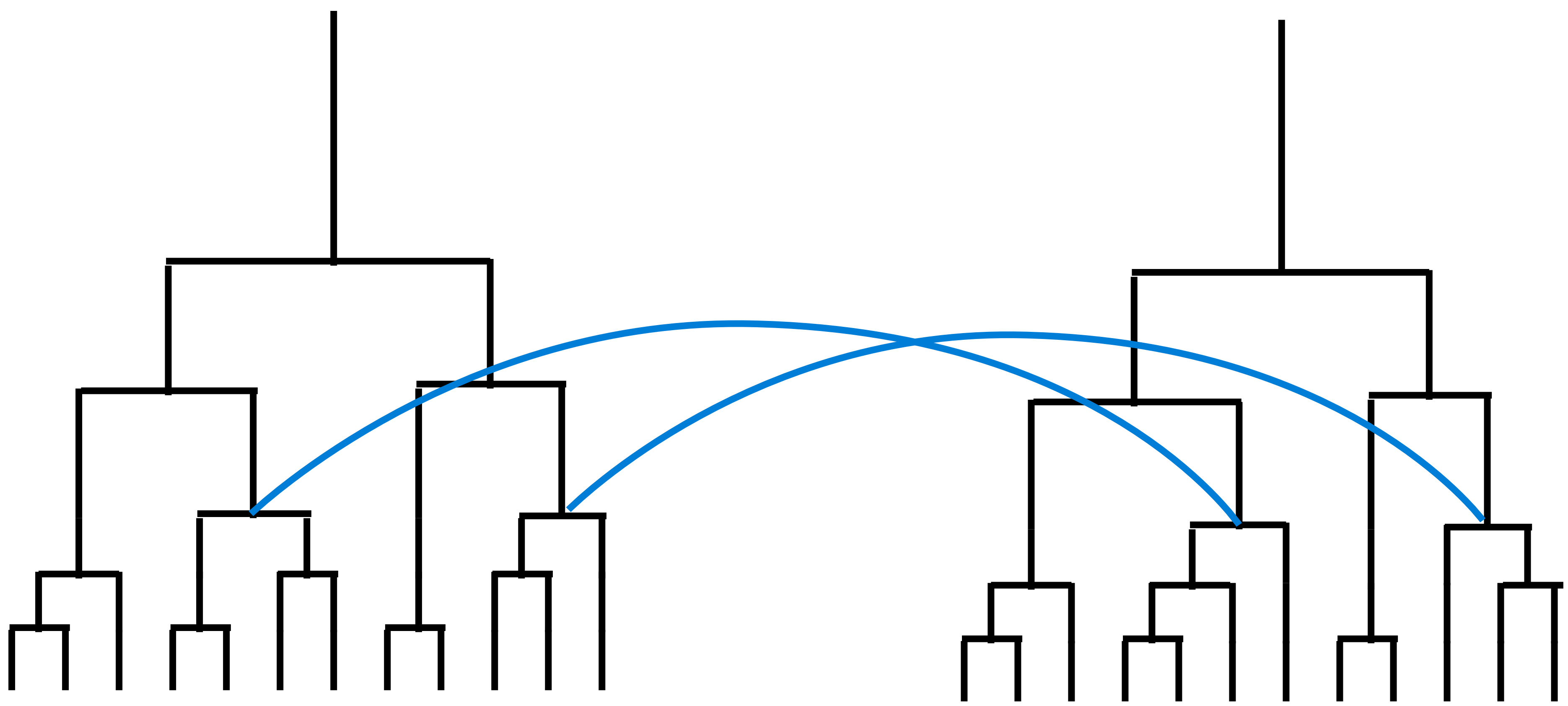
Hierarchical Clustering



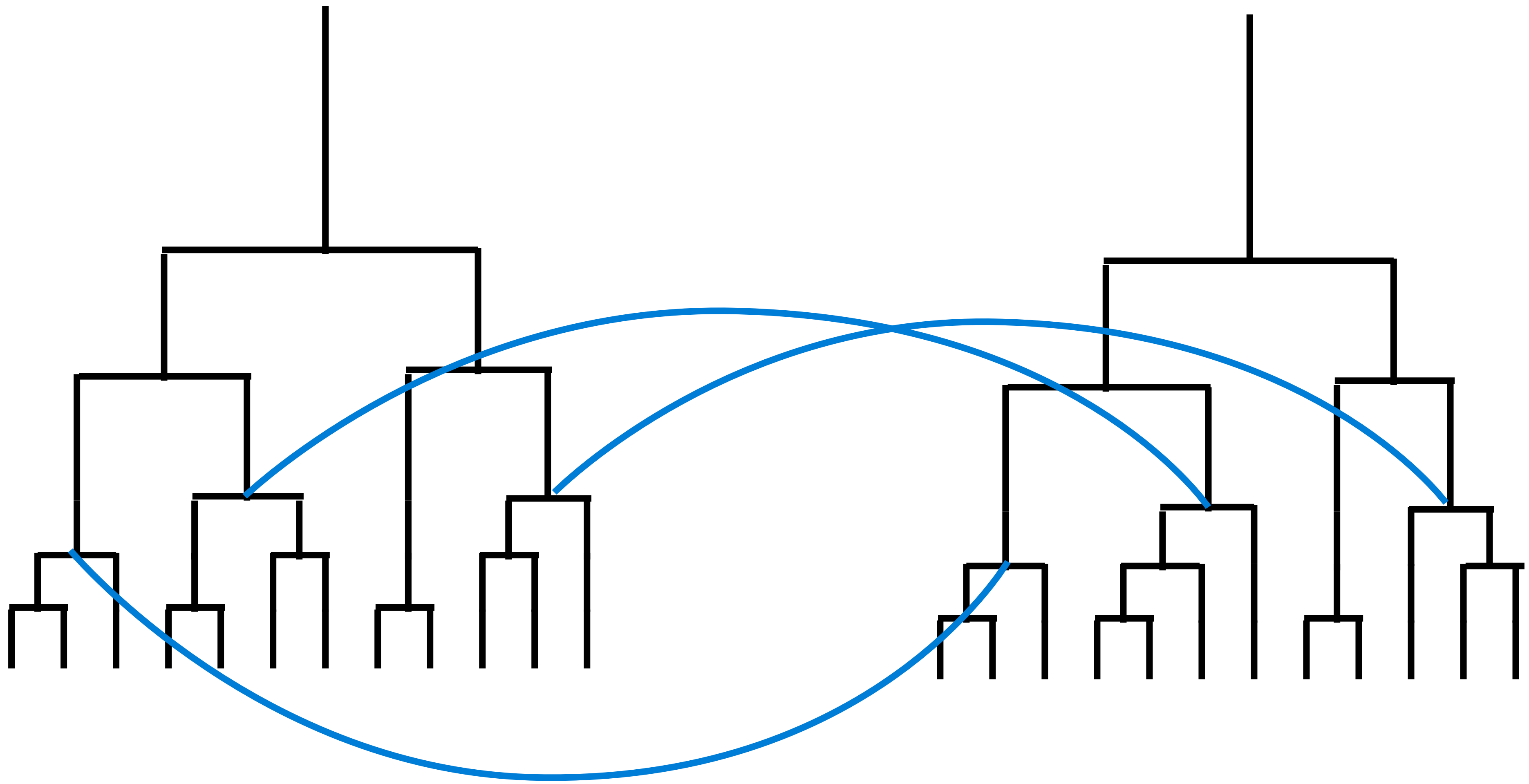
Hierarchical Clustering



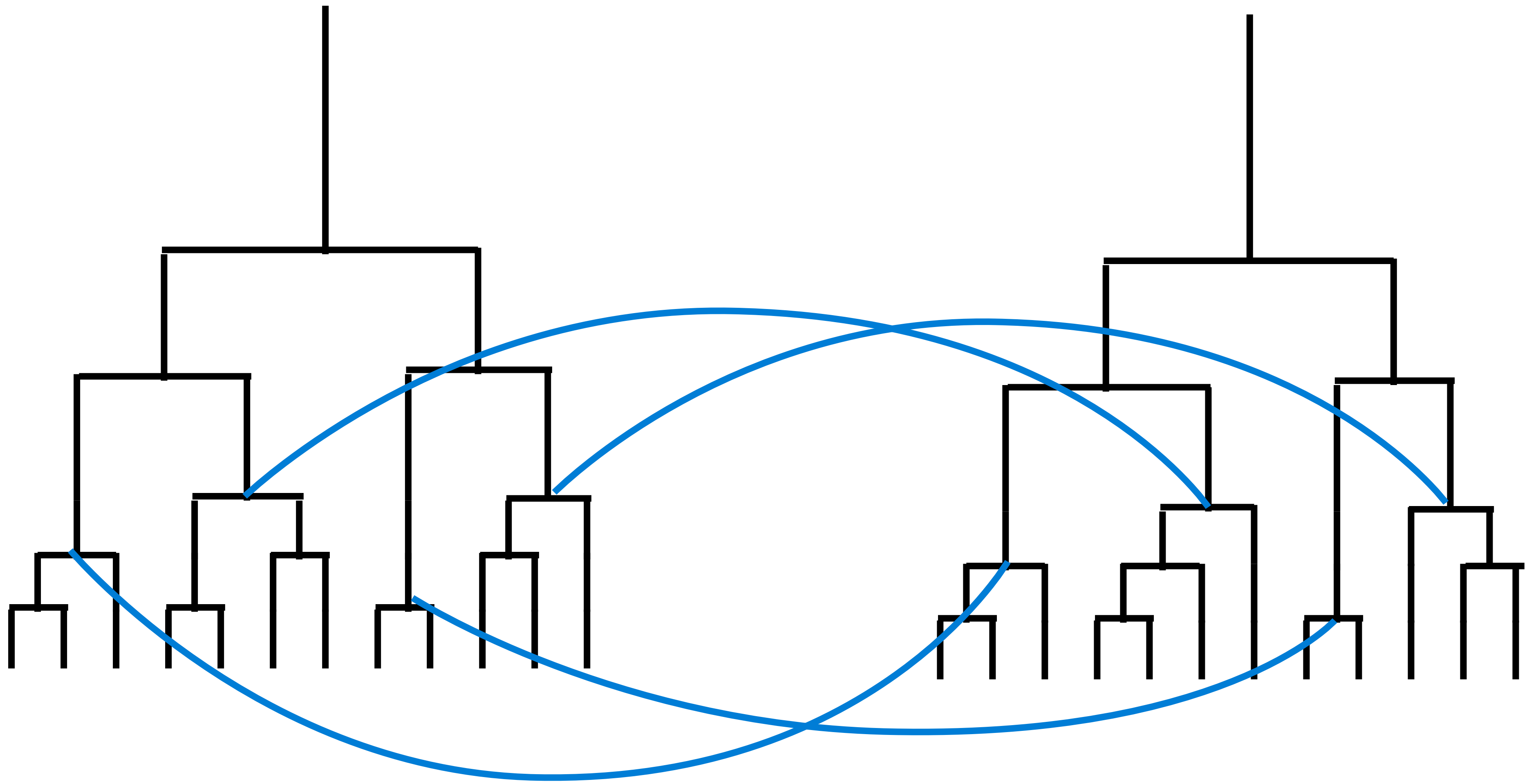
Hierarchical Clustering



Hierarchical Clustering



Hierarchical Clustering

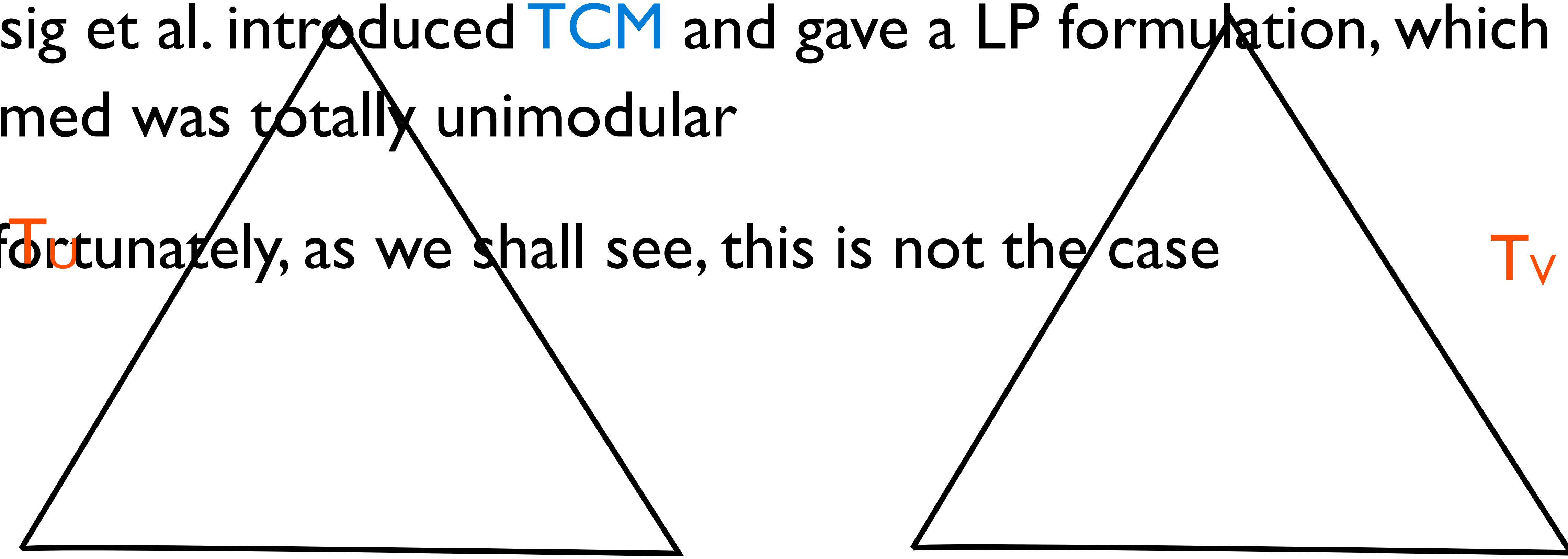


Tree-Constrained Matching

Given a weighted bipartite graph (U,V,E) and trees T_U and T_V over U and V, we want maximum weight matching M such that matched vertices in T_U and T_V are not comparable

Mosig et al. introduced TCM and gave a LP formulation, which they claimed was totally unimodular

Unfortunately, as we shall see, this is not the case

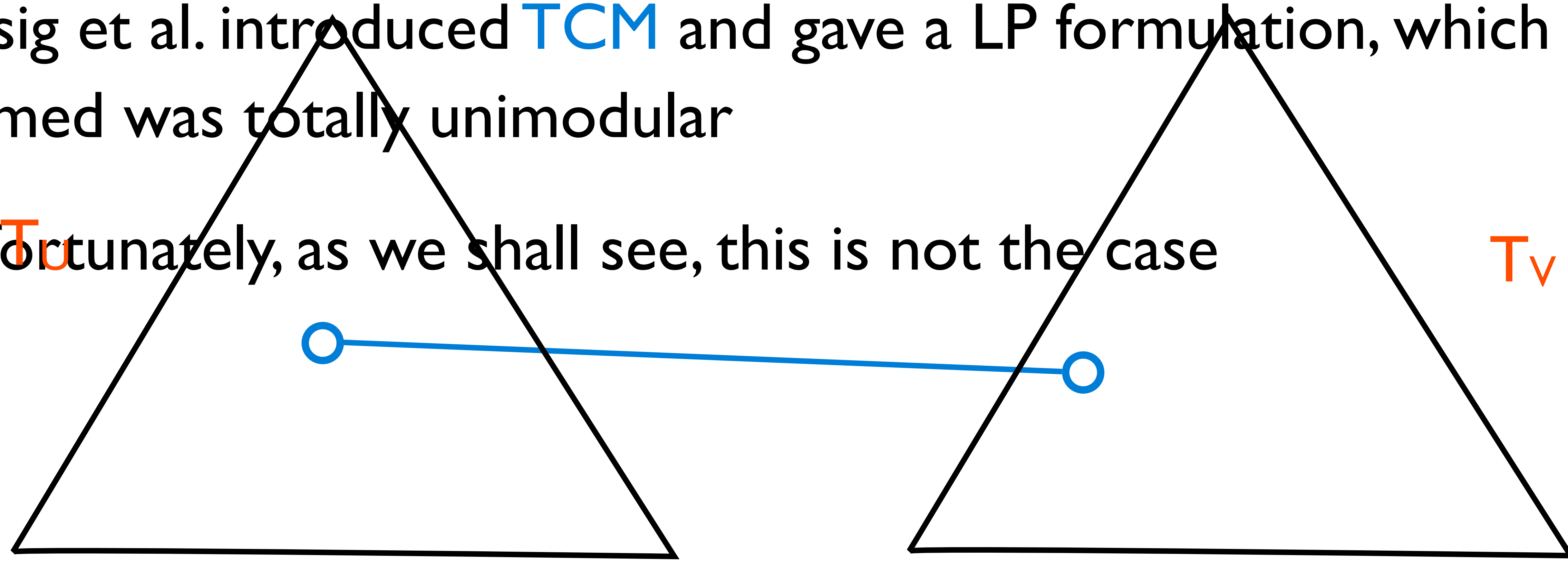


Tree-Constrained Matching

Given a weighted bipartite graph (U,V,E) and trees T_U and T_V over U and V, we want maximum weight matching M such that matched vertices in T_U and T_V are not comparable

Mosig et al. introduced TCM and gave a LP formulation, which they claimed was totally unimodular

Unfortunately, as we shall see, this is not the case

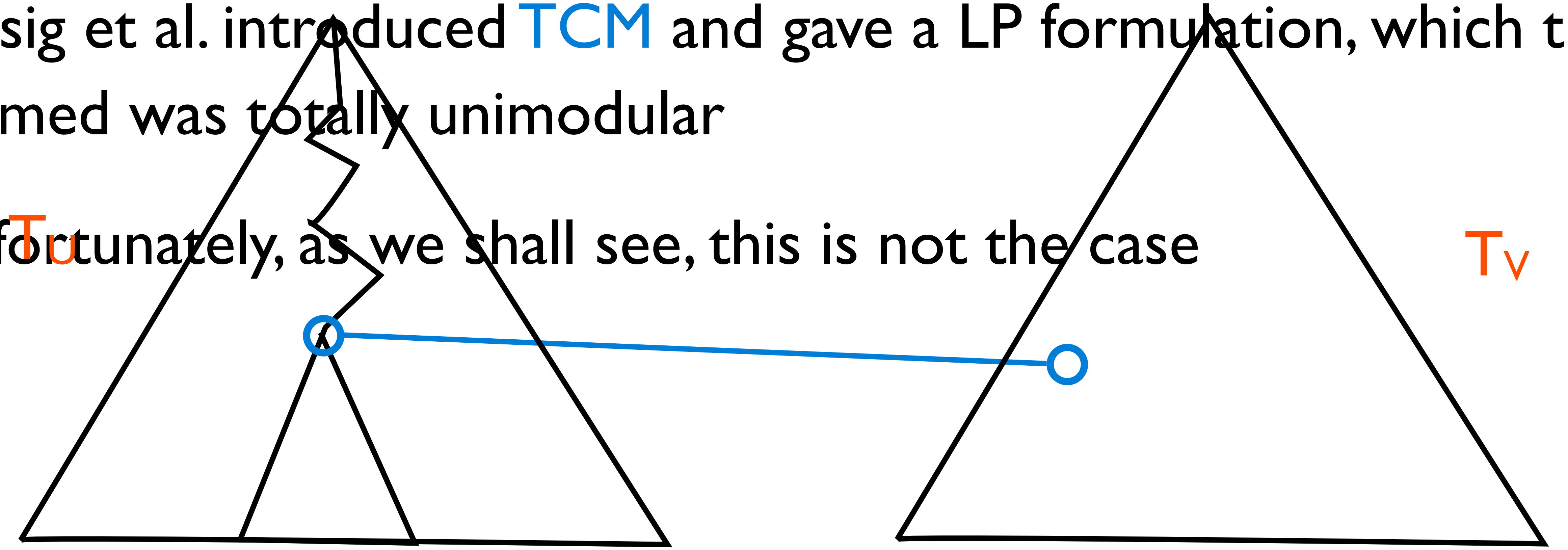


Tree-Constrained Matching

Given a weighted bipartite graph (U,V,E) and trees T_U and T_V over U and V, we want maximum weight matching M such that matched vertices in T_U and T_V are not comparable

Mosig et al. introduced TCM and gave a LP formulation, which they claimed was totally unimodular

Unfortunately, as we shall see, this is not the case

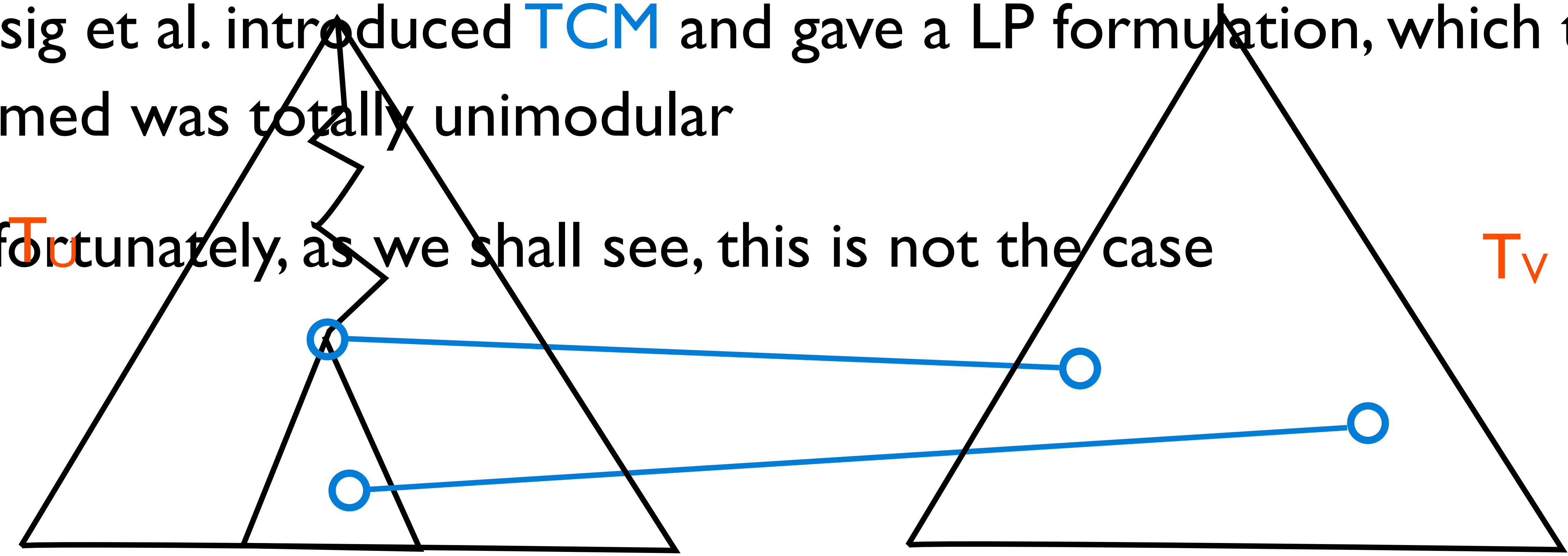


Tree-Constrained Matching

Given a weighted bipartite graph (U,V,E) and trees T_U and T_V over U and V, we want maximum weight matching M such that matched vertices in T_U and T_V are not comparable

Mosig et al. introduced TCM and gave a LP formulation, which they claimed was totally unimodular

Unfortunately, as we shall see, this is not the case

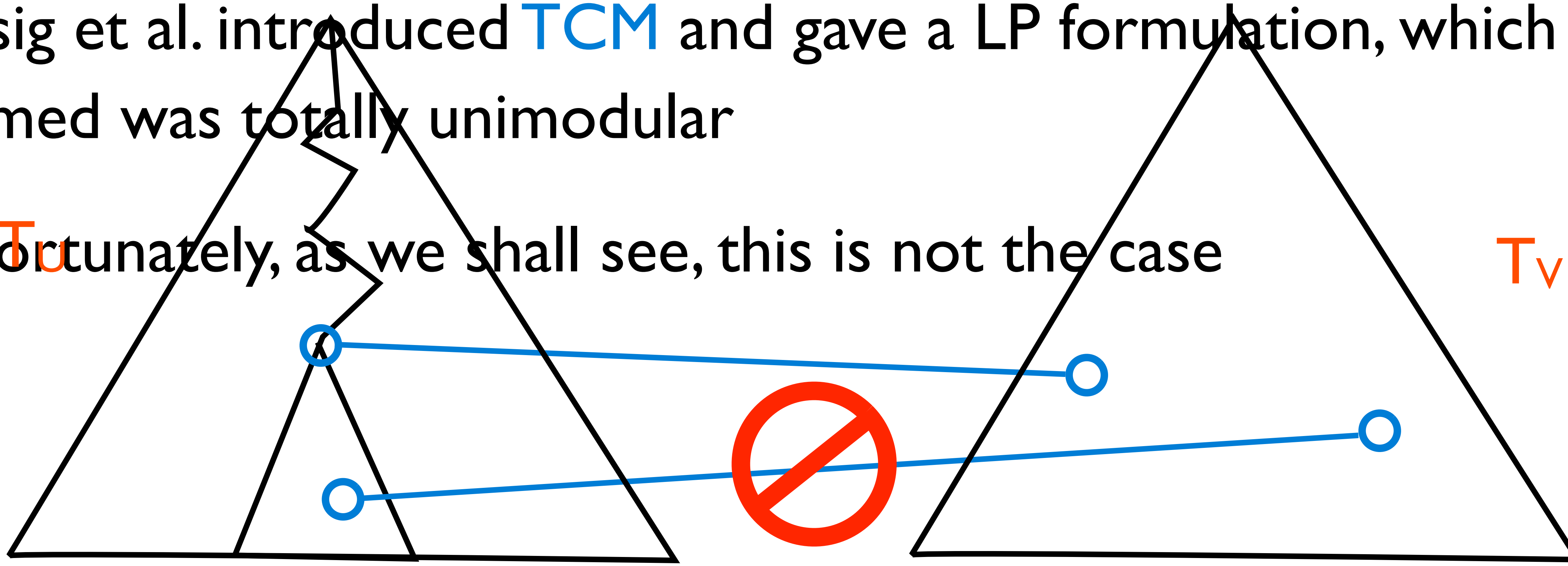


Tree-Constrained Matching

Given a weighted bipartite graph (U,V,E) and trees T_U and T_V over U and V, we want maximum weight matching M such that matched vertices in T_U and T_V are not comparable

Mosig et al. introduced TCM and gave a LP formulation, which they claimed was totally unimodular

Unfortunately, as we shall see, this is not the case

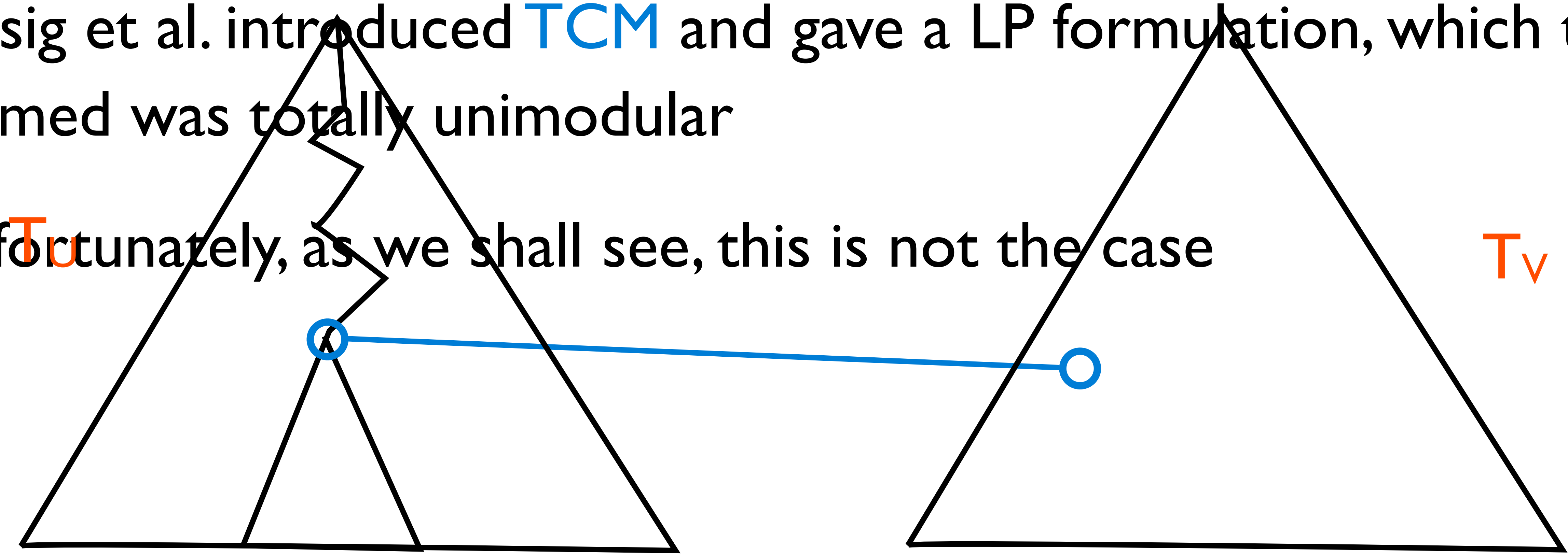


Tree-Constrained Matching

Given a weighted bipartite graph (U,V,E) and trees T_U and T_V over U and V, we want maximum weight matching M such that matched vertices in T_U and T_V are not comparable

Mosig et al. introduced TCM and gave a LP formulation, which they claimed was totally unimodular

Unfortunately, as we shall see, this is not the case

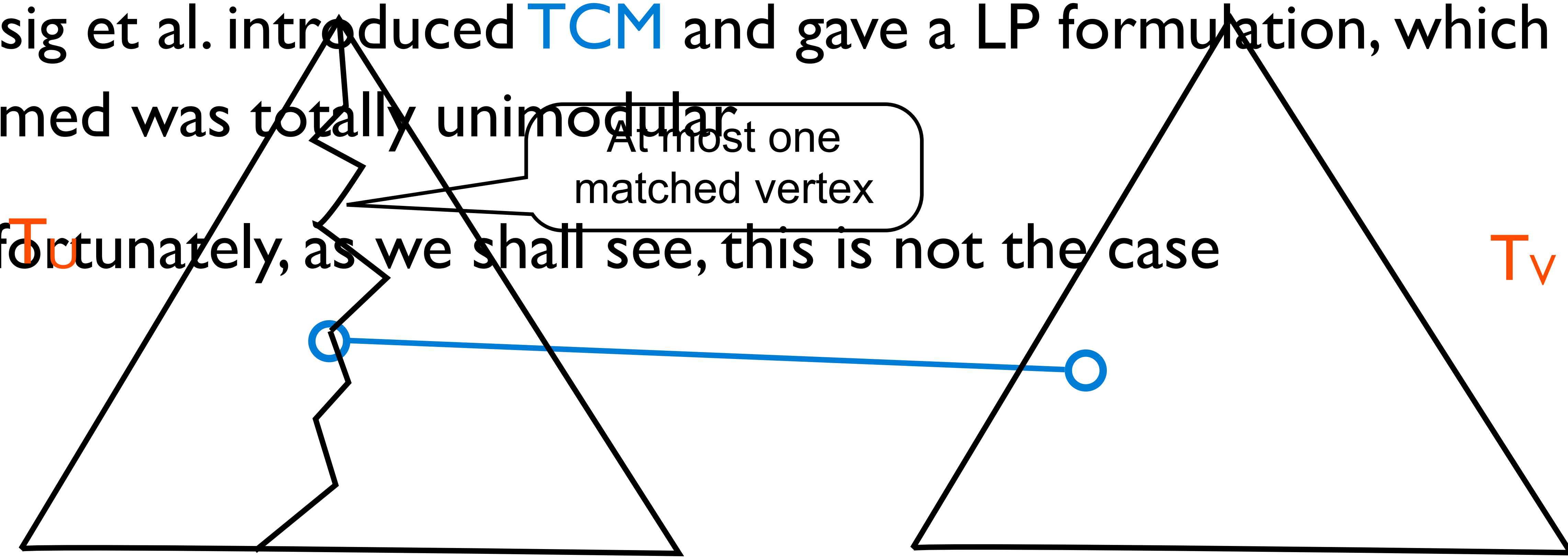


Tree-Constrained Matching

Given a weighted bipartite graph (U,V,E) and trees T_U and T_V over U and V, we want maximum weight matching M such that matched vertices in T_U and T_V are not comparable

Mosig et al. introduced TCM and gave a LP formulation, which they claimed was totally unimodular

Unfortunately, as we shall see, this is not the case

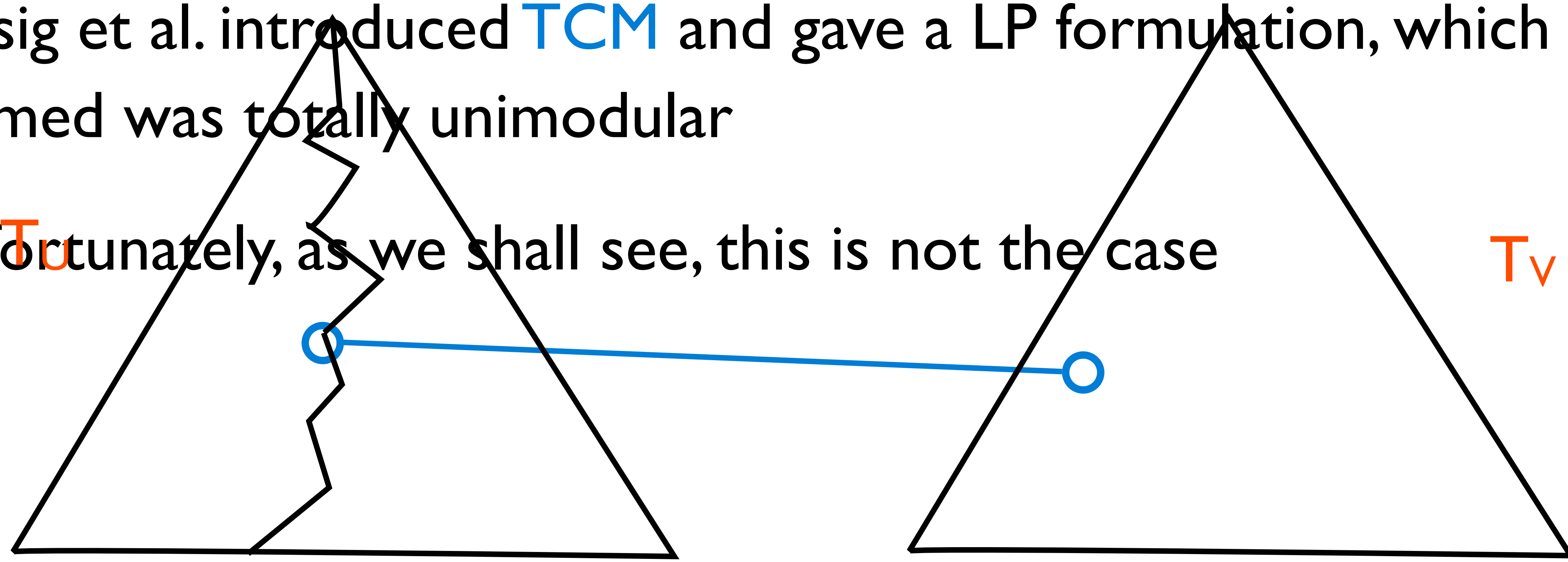


Tree-Constrained Matching

Given a weighted bipartite graph (U,V,E) and trees T_U and T_V over U and V, we want maximum weight matching M such that matched vertices in T_U and T_V are not comparable

Mosig et al. introduced TCM and gave a LP formulation, which they claimed was totally unimodular

Unfortunately, as we shall see, this is not the case



Tree-Constrained Matching

Given a weighted bipartite graph (U,V,E) and trees T_U and T_V over U and V, we want maximum weight matching M such that matched vertices in T_U and T_V are not comparable

Mosig et al. introduced TCM and gave a LP formulation, which they claimed was totally unimodular

Tree-Constrained Matching

Given a weighted bipartite graph (U,V,E) and trees T_U and T_V over U and V, we want maximum weight matching M such that matched vertices in T_U and T_V are not comparable

Mosig et al. introduced TCM and gave a LP formulation, which they claimed was totally unimodular

Unfortunately, as we shall see, this is not the case

MIS in d-Interval Graphs

MIS in d-Interval Graphs

There is a reduction from TCM to MIS in 2-interval graphs

MIS in d-Interval Graphs

There is a reduction from TCM to MIS in 2-interval graphs

A d-interval is $u = \{ [s_1, f_1], [s_2, f_2], \ldots, [s_d, f_d] \}$

MIS in d-Interval Graphs

There is a reduction from TCM to MIS in 2-interval graphs

A d-interval is $u = \{ [s_1, f_1], [s_2, f_2], \ldots, [s_d, f_d] \}$

A d-interval graph is the intersection graph of a set of d-intervals

MIS in d-Interval Graphs

There is a reduction from TCM to MIS in 2-interval graphs

A d-interval is $u = \{ [s_1, f_1], [s_2, f_2], \ldots, [s_d, f_d] \}$

A d-interval graph is the intersection graph of a set of d-intervals

Maximum weight independent set (MIS) in d-interval graphs:

MIS in d-Interval Graphs

There is a reduction from TCM to MIS in 2-interval graphs

A d-interval is $u = \{ [s_1, f_1], [s_2, f_2], \ldots, [s_d, f_d] \}$

A d-interval graph is the intersection graph of a set of d-intervals

Maximum weight independent set (MIS) in d-interval graphs:

- [Folk] If $d=1$, there is an exact algorithm

MIS in d-Interval Graphs

There is a reduction from TCM to MIS in 2-interval graphs

A d-interval is $u = \{ [s_1, f_1], [s_2, f_2], \ldots, [s_d, f_d] \}$

A d-interval graph is the intersection graph of a set of d-intervals

Maximum weight independent set (MIS) in d-interval graphs:

- [Folk] If $d=1$, there is an exact algorithm
- [BHNSS] If $d>1$, the problem is APX-hard, but there is a $2d$-approximation

MIS in d-Interval Graphs

There is a reduction from TCM to MIS in 2-interval graphs

A d-interval is $u = \{ [s_1, f_1], [s_2, f_2], ..., [s_d, f_d] \}$

A d-interval graph is the intersection graph of a set of d-intervals

Maximum weight independent set (MIS) in d-interval graphs:

- [Folk] If $d=1$, there is an exact algorithm
- [BHNSS] If $d>1$, the problem is APX-hard, but there is a 2d-approximation

Thus, there is a 4-approximation for TCM

Our results

Our results

Show that TCM is APX-hard and disprove claim of Mosig et al.

Our results

Show that TCM is APX-hard and disprove claim of Mosig et al.

Bar-Yehuda et al. algorithm is in fact a 3-approximation for TCM

Our results

Show that TCM is APX-hard and disprove claim of Mosig et al.

Bar-Yehuda et al. algorithm is in fact a 3-approximation for TCM

Give 2-approximation, matching integrality gap of LP formulation

Our results

Show that TCM is APX-hard and disprove claim of Mosig et al.

Bar-Yehuda et al. algorithm is in fact a 3-approximation for TCM

Give 2-approximation, matching integrality gap of LP formulation

Generalization to posets

Reducing TCM to MIS in 2-IG

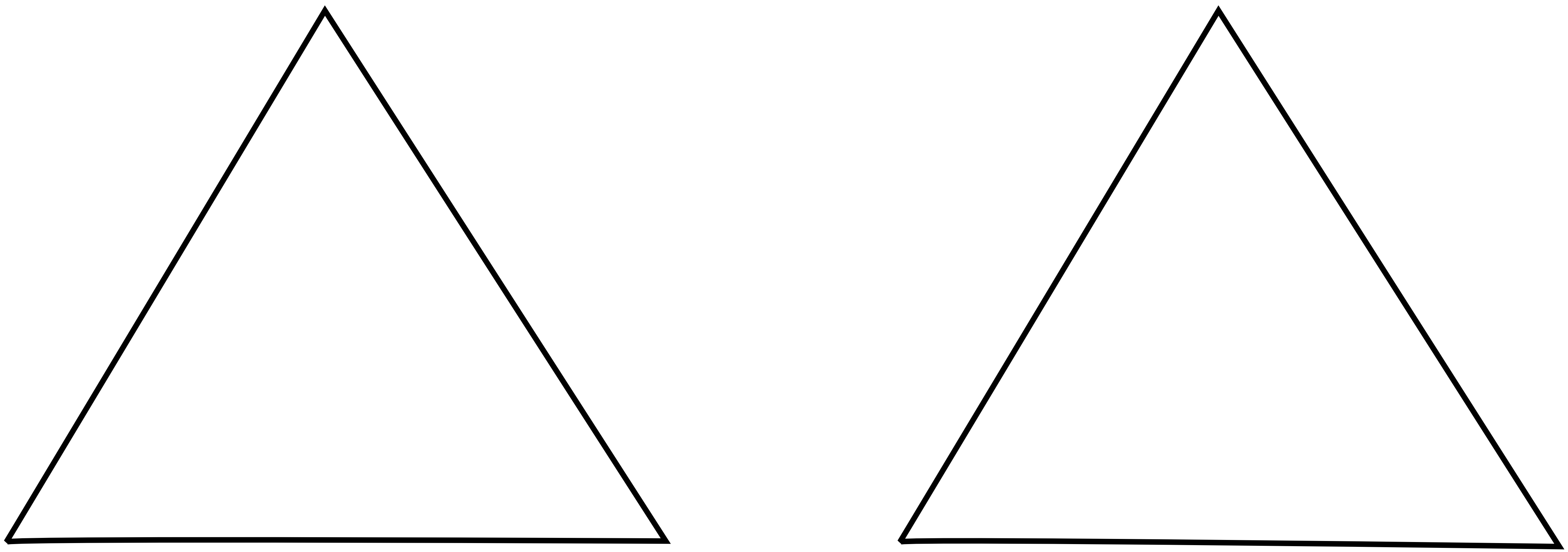
Every edge (u,v) is assigned one interval in TU and one in TV

Matching is feasible ⇔ corresp. set of 2-intervals is independent

Reducing TCM to MIS in 2-IG

Every edge (u,v) is assigned one interval in TU and one in TV

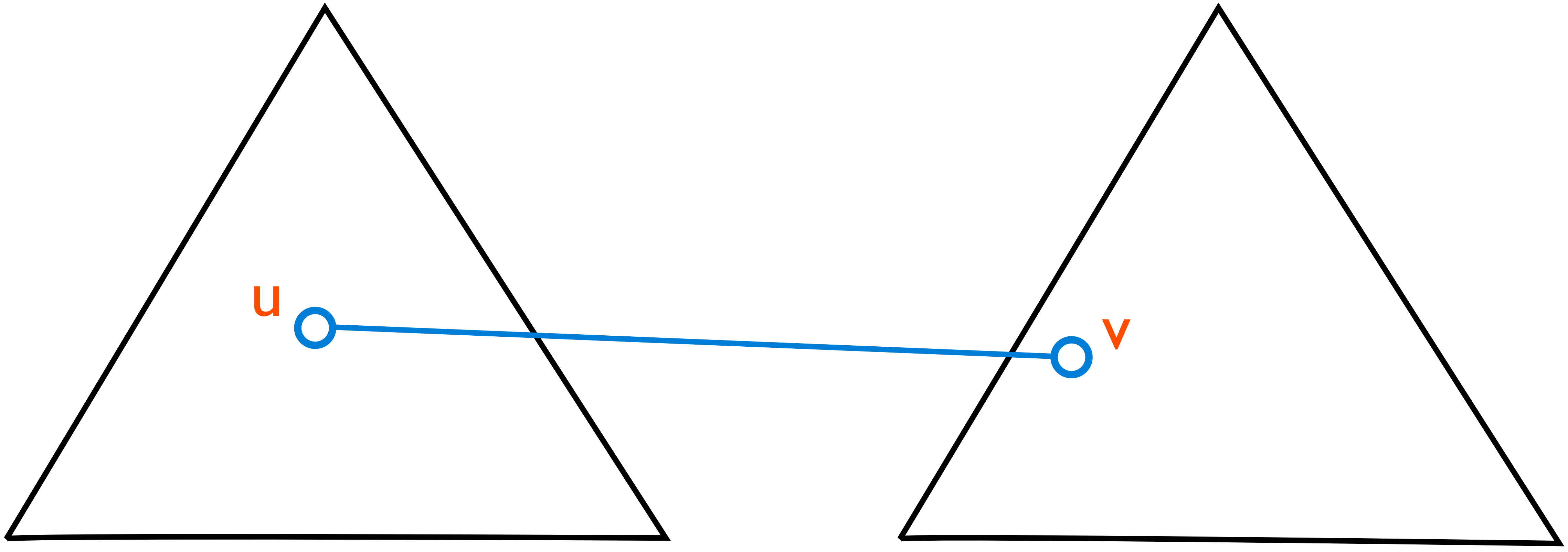
Matching is feasible ⇔ corresp. set of 2-intervals is independent



Reducing TCM to MIS in 2-IG

Every edge (u,v) is assigned one interval in TU and one in TV

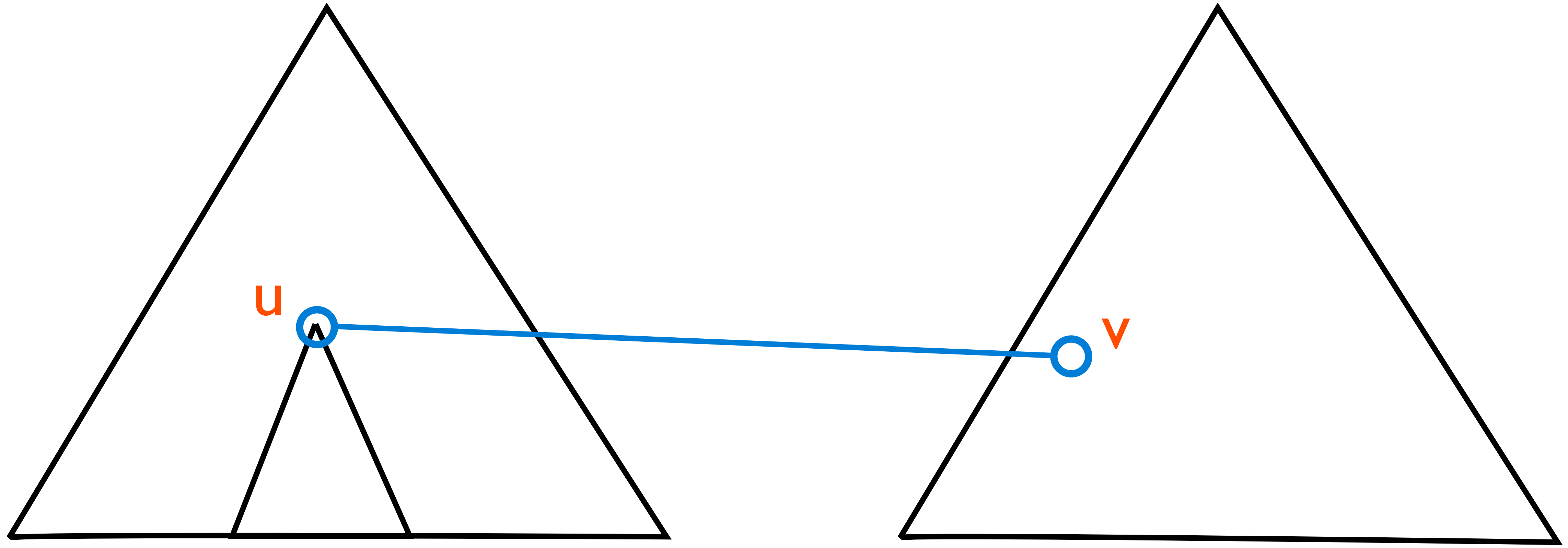
Matching is feasible ⇔ corresp. set of 2-intervals is independent



Reducing TCM to MIS in 2-IG

Every edge (u,v) is assigned one interval in TU and one in TV

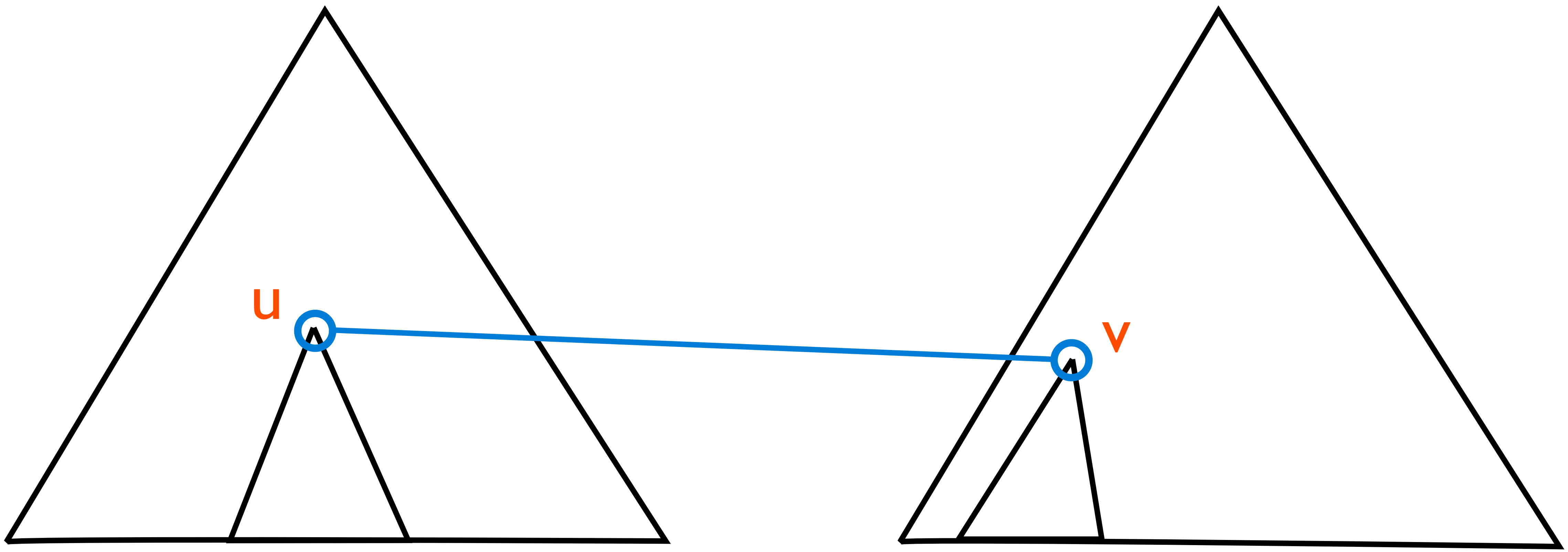
Matching is feasible ⇔ corresp. set of 2-intervals is independent



Reducing TCM to MIS in 2-IG

Every edge (u,v) is assigned one interval in TU and one in TV

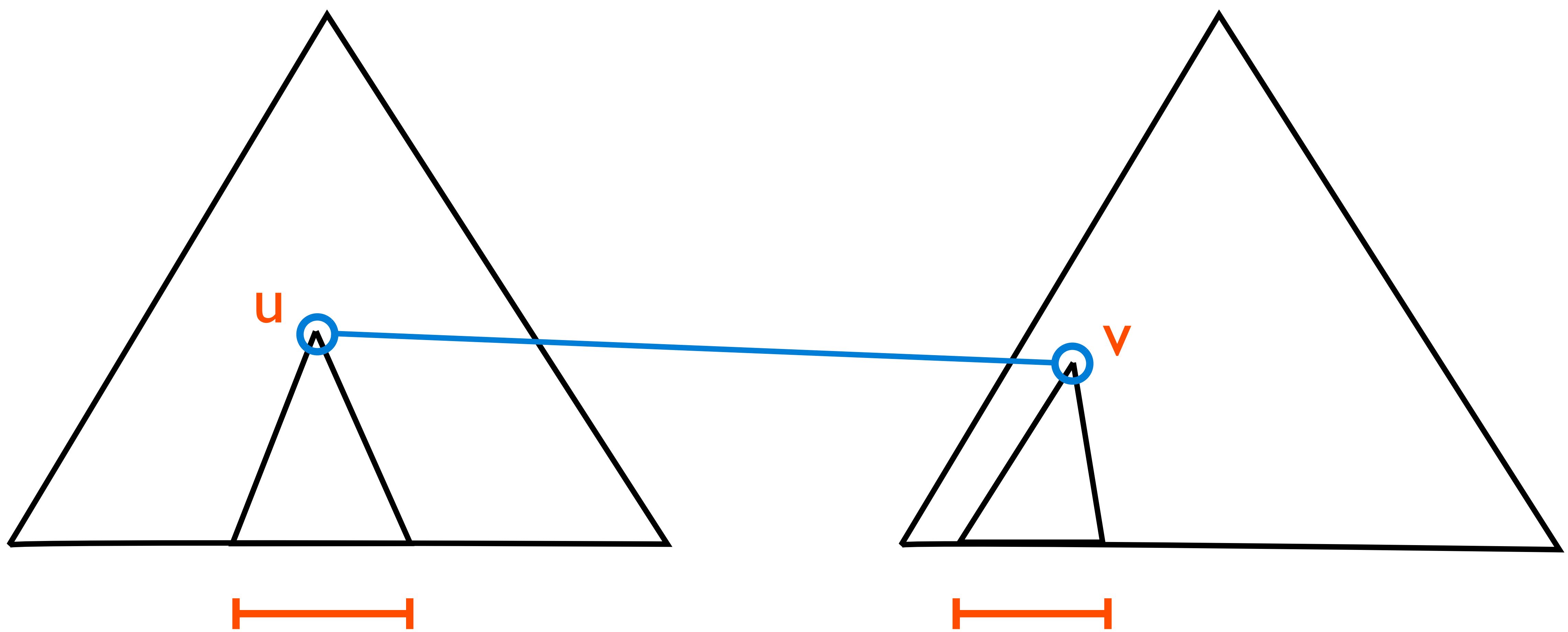
Matching is feasible ⇔ corresp. set of 2-intervals is independent



Reducing TCM to MIS in 2-IG

Every edge (u,v) is assigned one interval in TU and one in TV

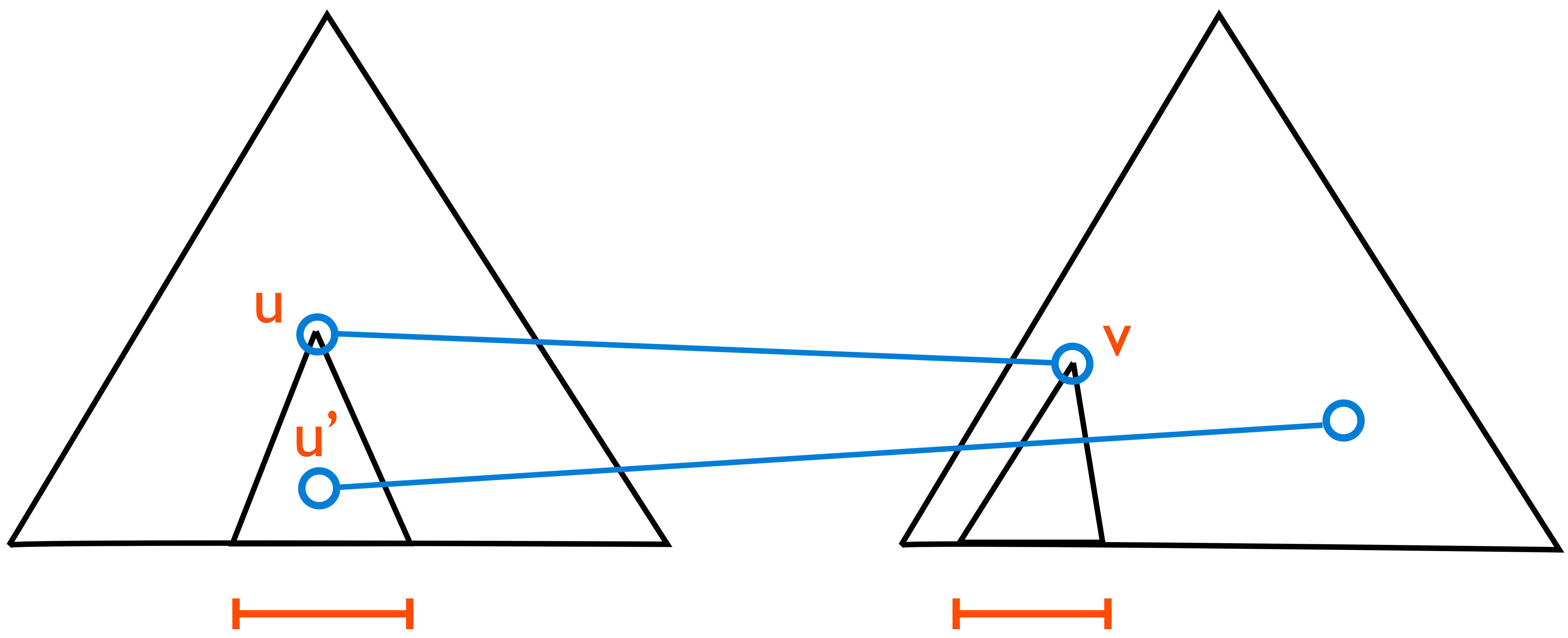
Matching is feasible ⇔ corresp. set of 2-intervals is independent



Reducing TCM to MIS in 2-IG

Every edge (u,v) is assigned one interval in TU and one in TV

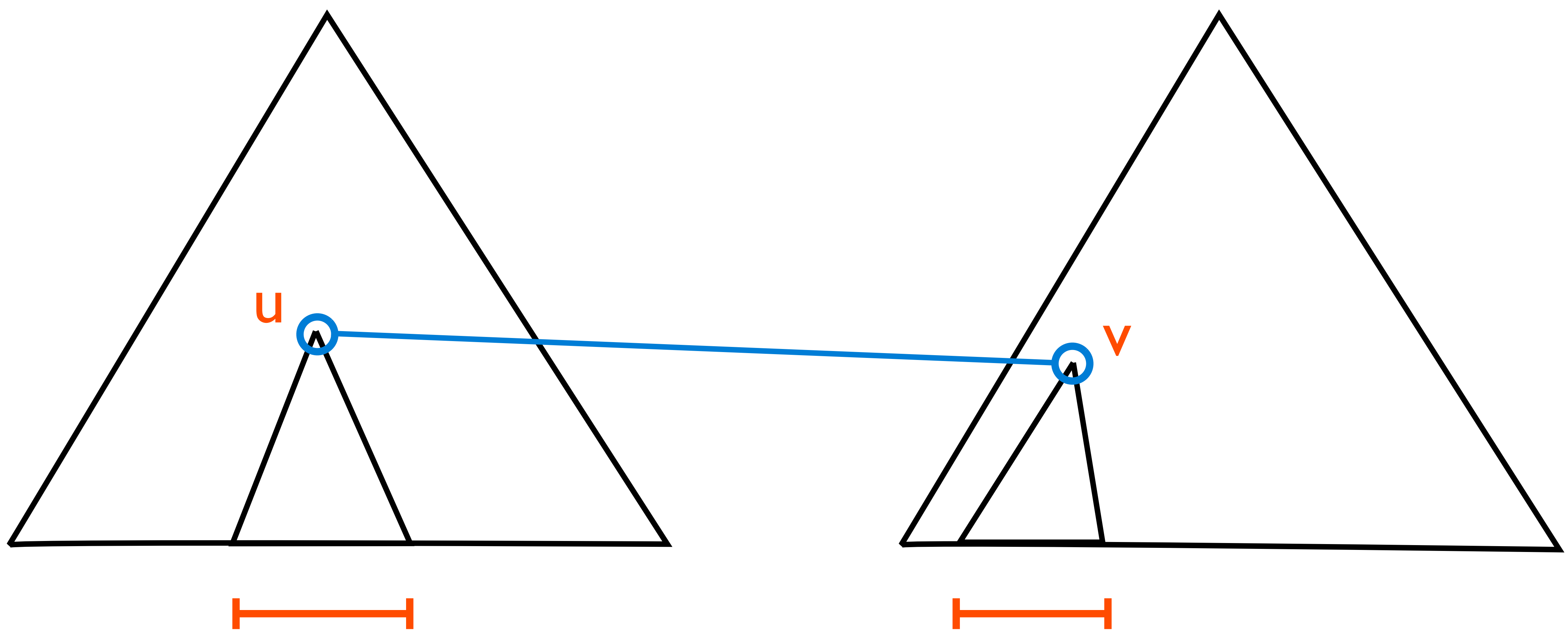
Matching is feasible ⇔ corresp. set of 2-intervals is independent



Reducing TCM to MIS in 2-IG

Every edge (u,v) is assigned one interval in TU and one in TV

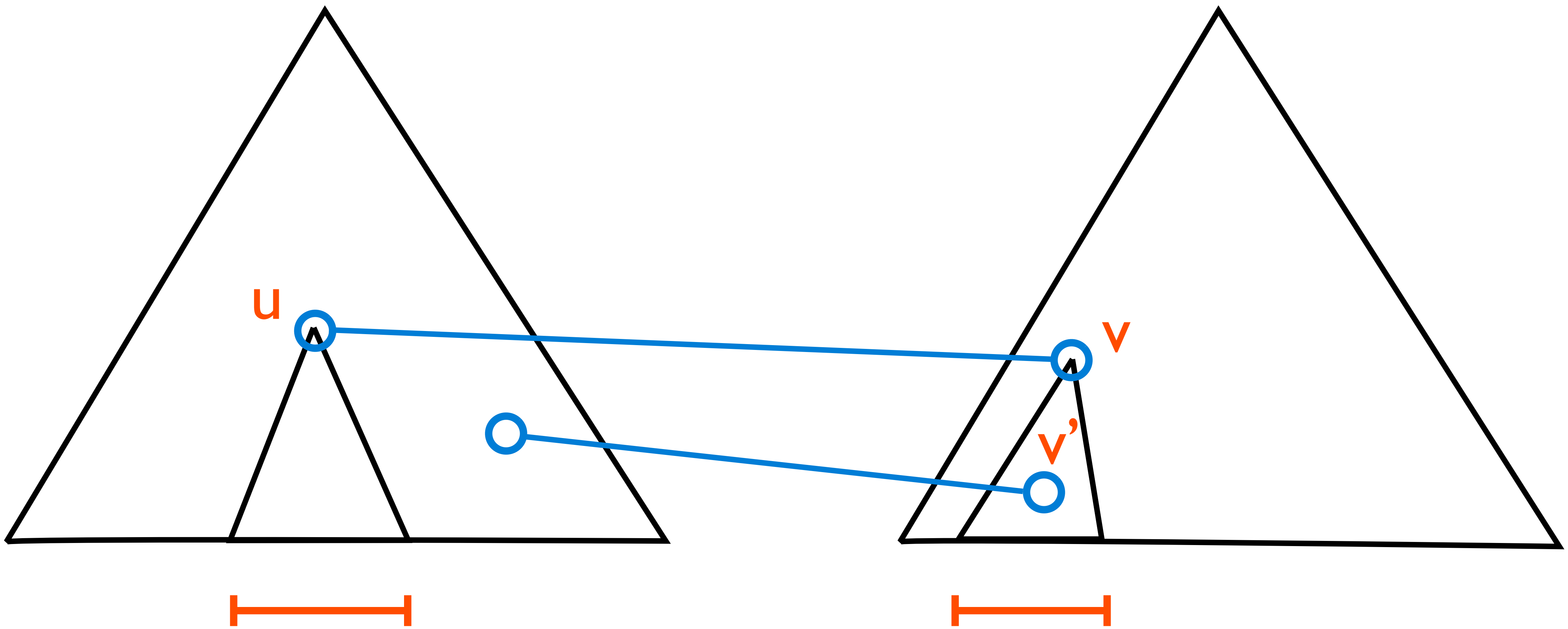
Matching is feasible ⇔ corresp. set of 2-intervals is independent



Reducing TCM to MIS in 2-IG

Every edge (u,v) is assigned one interval in TU and one in TV

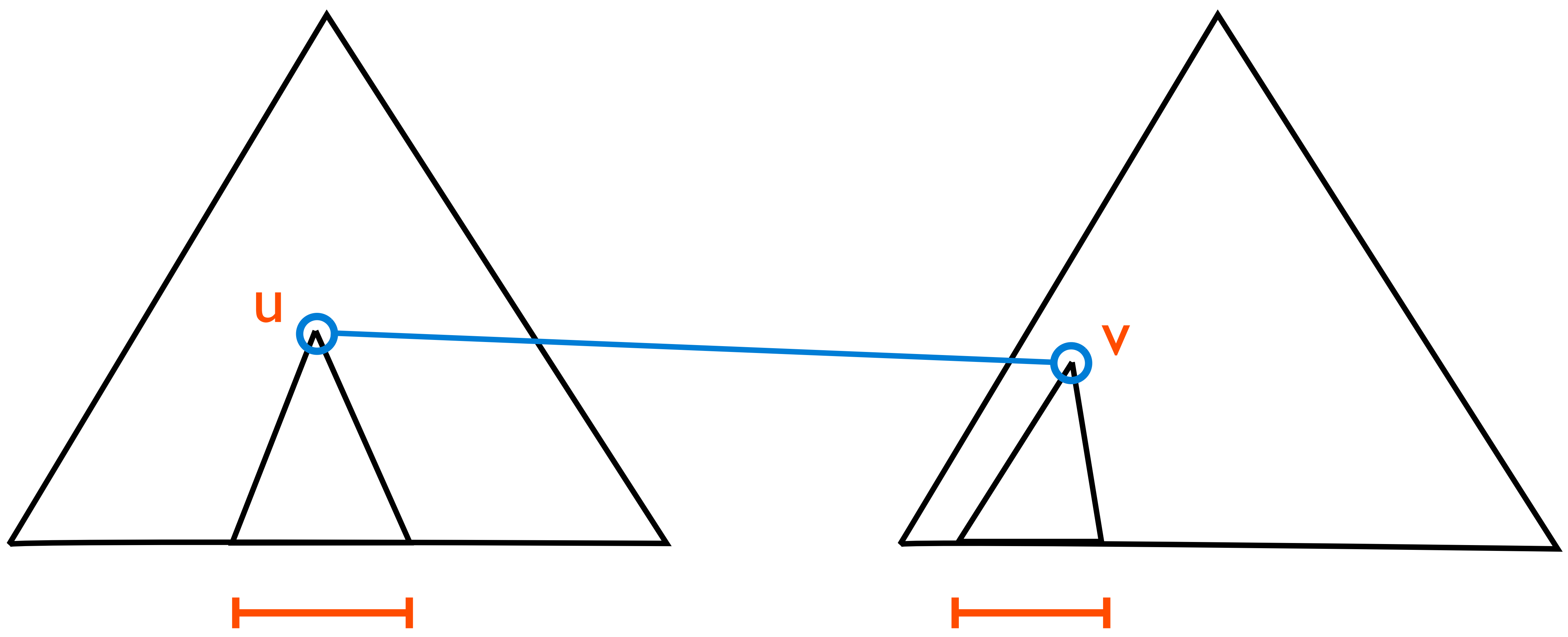
Matching is feasible ⇔ corresp. set of 2-intervals is independent



Reducing TCM to MIS in 2-IG

Every edge (u,v) is assigned one interval in TU and one in TV

Matching is feasible ⇔ corresp. set of 2-intervals is independent



Bar-Yehuda et al. algorithm

minimize $\sum_u x_u$

subject to

$$\sum_{u\,:\,p \text{ in } u} x_u \leq 1 \quad \forall \text{ point } p$$

$$x_u \geq 0 \quad \forall \text{ d-interval } u$$

Bar-Yehuda et al. algorithm

minimize $\sum_u x_u$

subject to

$\sum_{u\,:\,p \text{ in } u} x_u \leq 1 \;\forall$ point p

$x_u \geq 0 \;\forall$ d-interval u

MIS-d-interval(G)

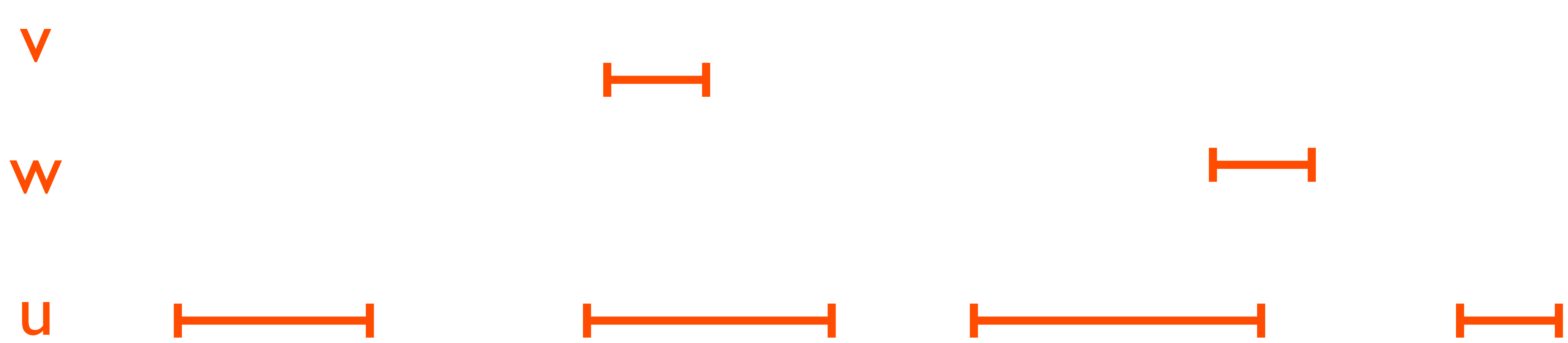
1. let x be optimal LP solution
2. let S be the empty set
3. while G is non empty
4. let u minimize x(N(u))
5. add u to S
6. remove N(u) + u from G
7. return S

$\forall$ feasible $x : \exists\, u : x(N(u)) \leq 2d$

$\sum_u x_u \sum_{v \text{ in } N(u)} x_v$

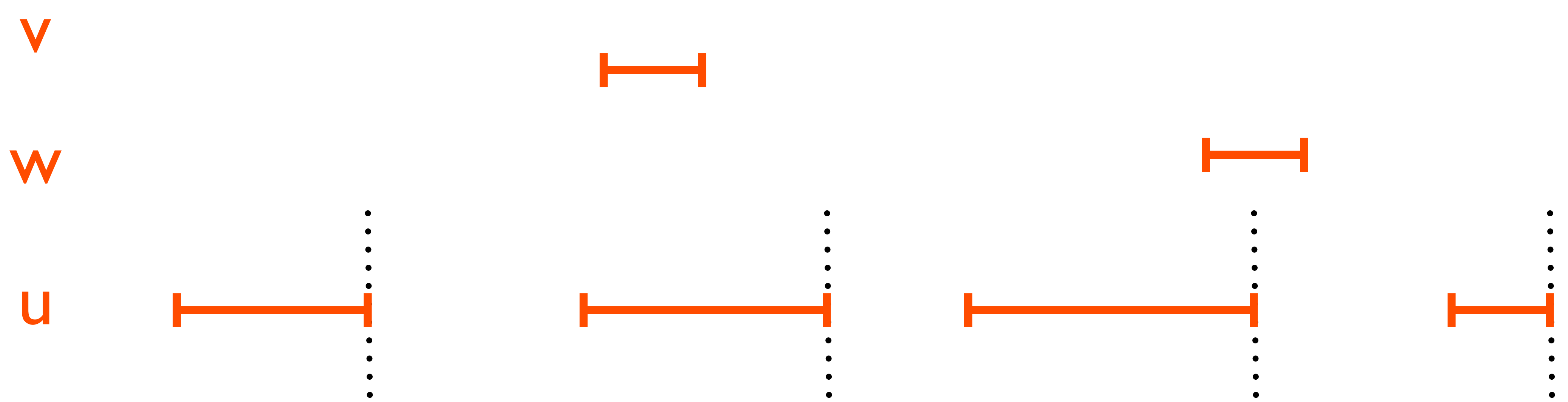
$\forall$ feasible x : $\exists$ u : x(N(u) $\leq$ 2d

$\sum_u x_u \sum_{v \text{ in } N(u)} x_v$



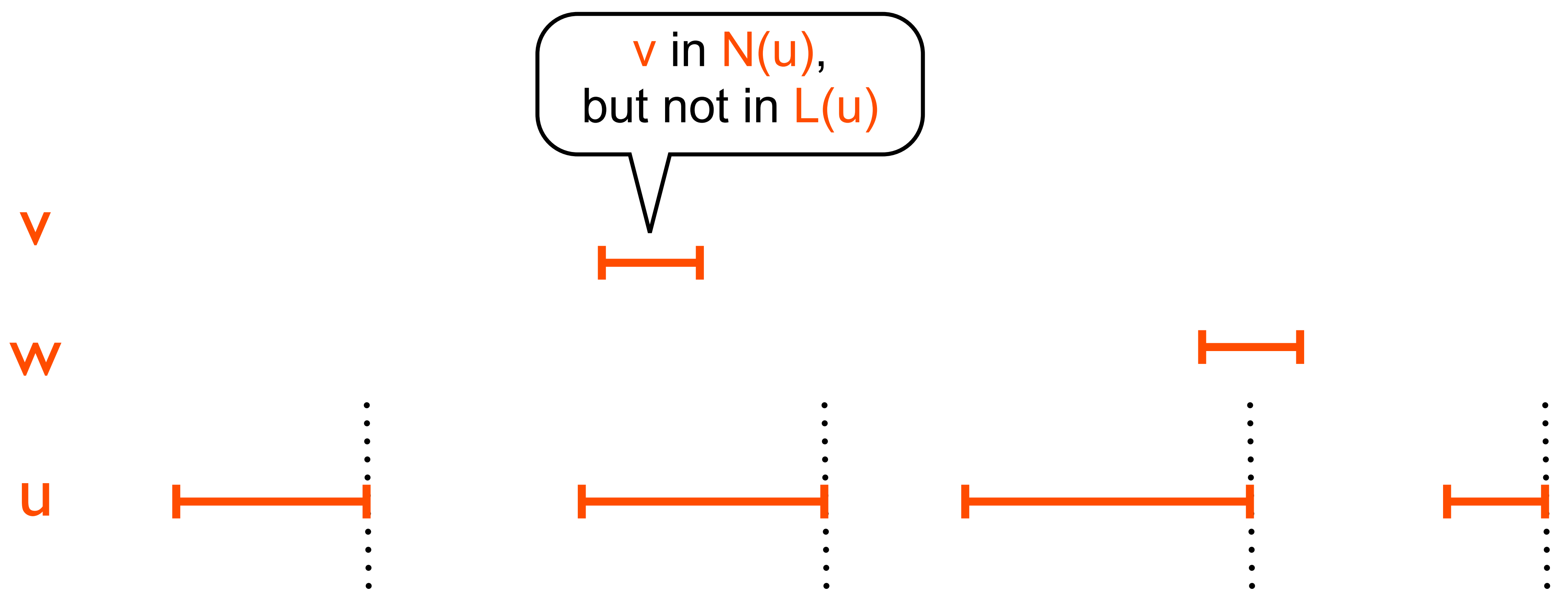
∀ feasible x : ∃ u : x(N(u)) ≤ 2d

$\sum_u x_u \sum_{v \text{ in } N(u)} x_v$



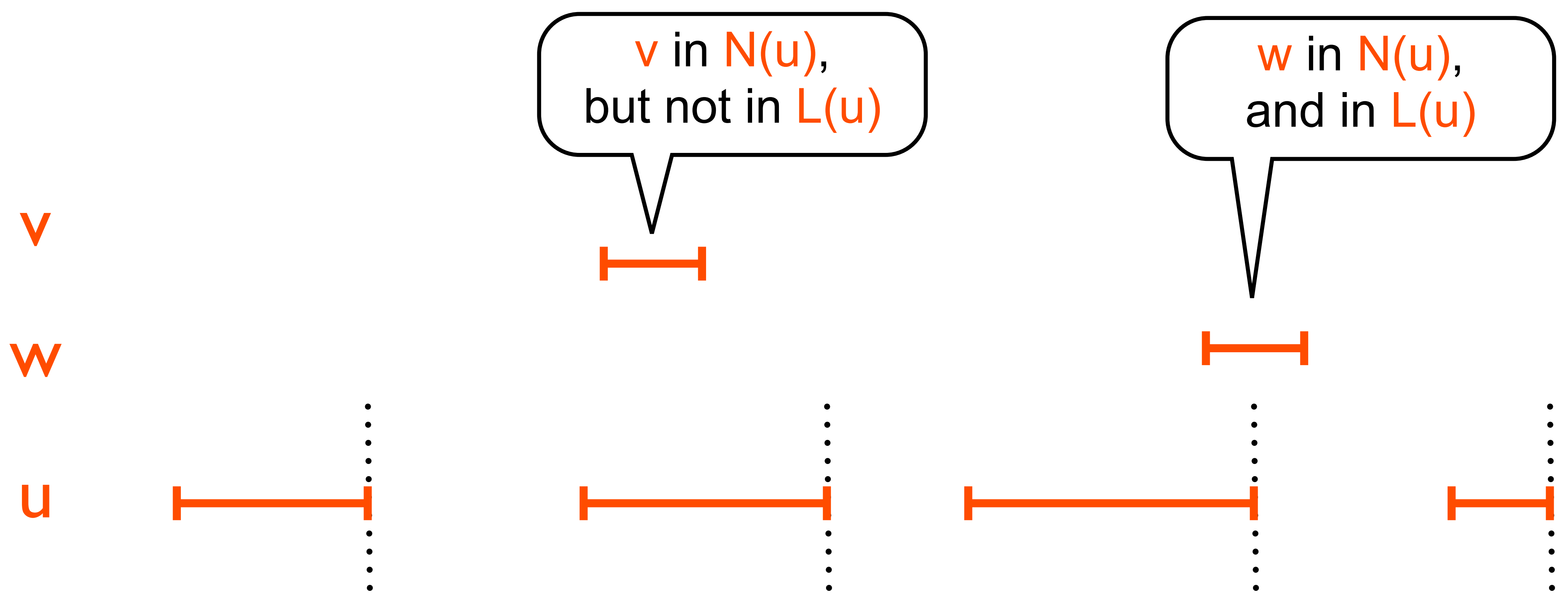
$\forall$ feasible x : $\exists$ u : x(N(u) $\leq$ 2d

$$\sum_u x_u \sum_{v \text{ in } N(u)} x_v$$



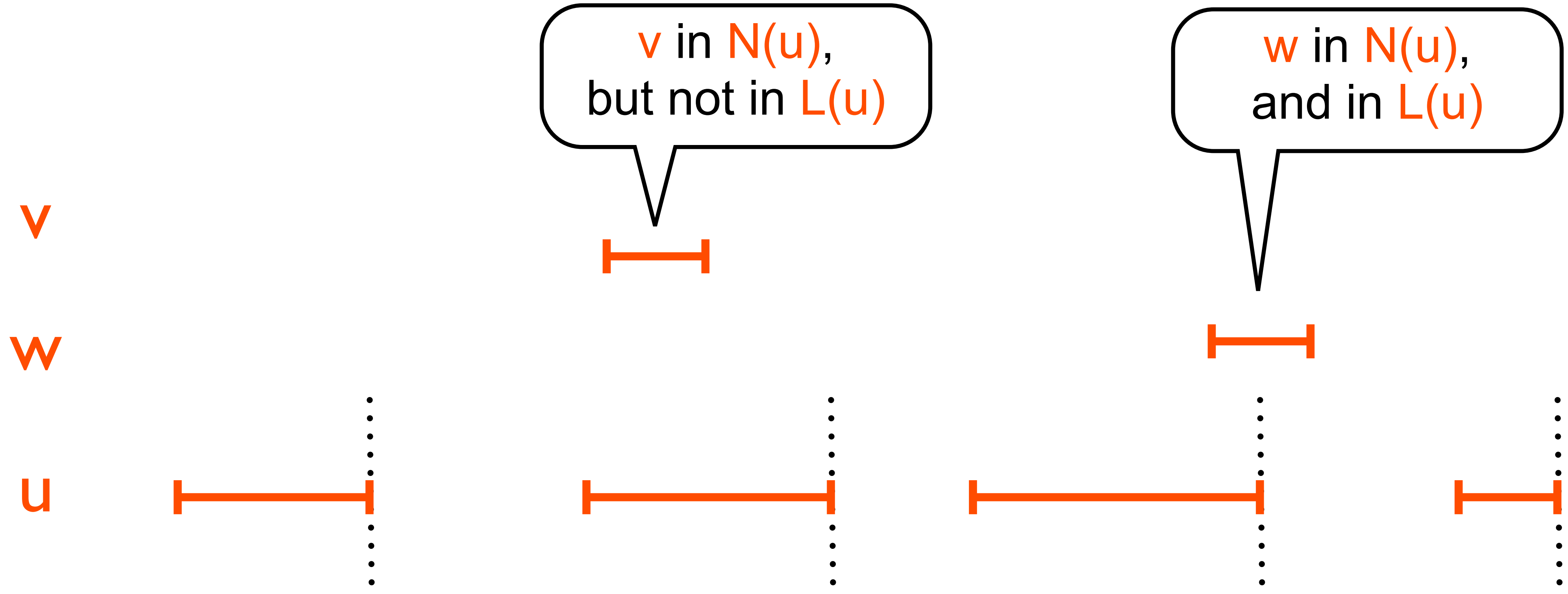
$\forall$ feasible x : $\exists$ u : $x(N(u)) \leq 2d$

$\sum_u x_u \sum_{v \text{ in } N(u)} x_v$



$\forall$ feasible $x : \exists\, u : x(N(u)) \leq 2d$

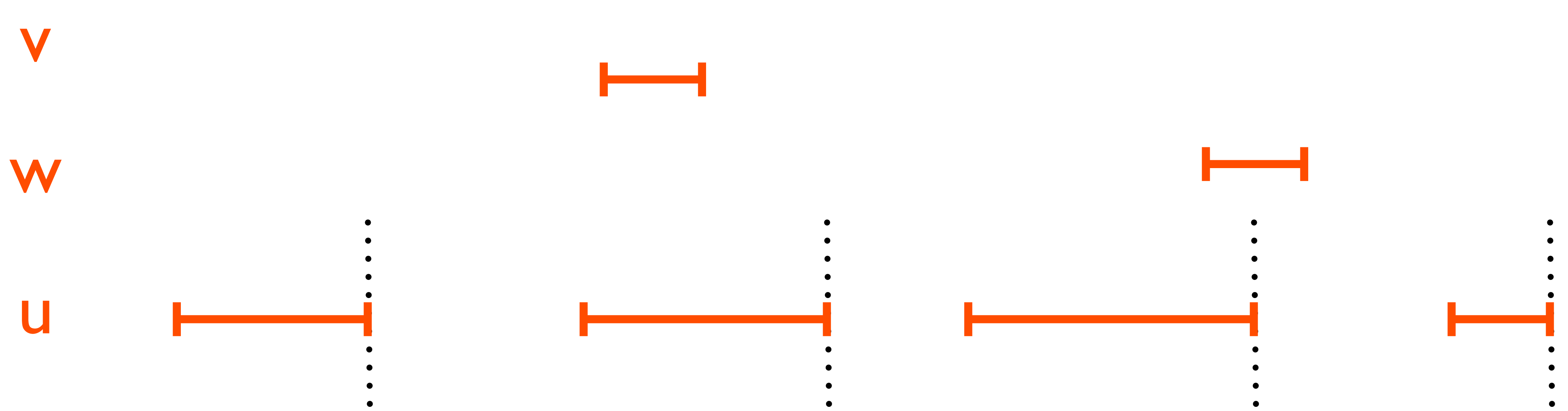
$$\sum_u x_u \sum_{v \text{ in } N(u)} x_v \leq 2 \sum_u x_u \sum_{v \text{ in } L(u)} x_v$$



$\forall$ feasible x : $\exists$ u : x(N(u)) $\leq$ 2d

$$\sum_u x_u \sum_{v \text{ in } N(u)} x_v \leq 2 \sum_u x_u \sum_{v \text{ in } L(u)} x_v$$

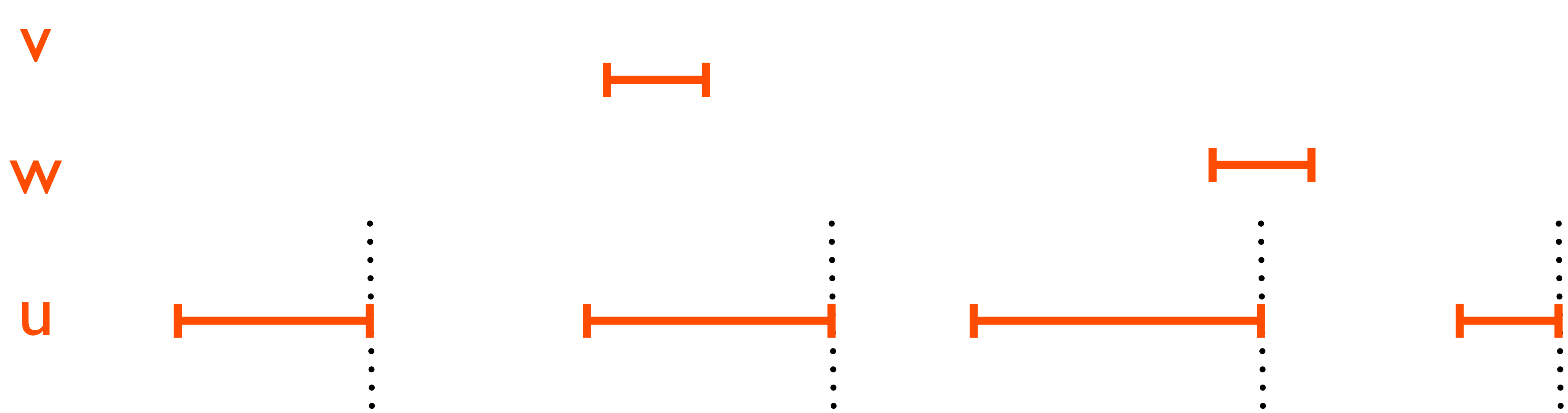
$$\leq 2 \sum_u x_u \, d$$



THE UNIVERSITY OF SYDNEY

$\forall$ feasible x : $\exists$ u : x(N(u)) $\leq$ 2d

$$\begin{aligned}\textstyle\sum_u x_u \sum_{v \text{ in } N(u)} x_v &\leq 2 \textstyle\sum_u x_u \sum_{v \text{ in } L(u)} x_v \\ &\leq 2 \textstyle\sum_u x_u\, d \\ &\leq 2d \textstyle\sum_u x_u\end{aligned}$$



4-approximation for TCM

minimize $\sum_e x_e$

subject to

$\sum_{e \text{ on } P} x_e \leq 1 \quad \forall$ desc. path P

$x_e \geq 0 \quad \forall$ edge e

4-approximation for TCM

minimize $\sum_e x_e$

subject to

$\sum_{e \text{ on } P} x_e \leq 1 \quad \forall$ desc. path P

$x_e \geq 0 \quad \forall$ edge e

TCM(U,V,E)

1. let x be optimal LP solution
2. let M be the empty set
3. while E is non empty
4. let e minimize x(N(e))
5. add e to M
6. remove N(e) + e from E
7. return M

4-approximation for TCM

minimize $\sum_e x_e$

subject to

$\sum_{e \text{ on } P} x_e \leq 1 \quad \forall$ desc. path P

$x_e \geq 0 \quad \forall$ edge e

N(e) is the set of edges in conflict with e

TCM(U,V,E)

1. let x be optimal LP solution
2. let M be the empty set
3. while E is non empty
4. let e minimize x(N(e))
5. add e to M
6. remove N(e) + e from E
7. return M

$\forall$ basic feasible x : $\exists$ e : x(N(e) < 3

$\forall$ basic feasible x : $\exists$ e : x(N(e) < 3

For ease of analysis, assume that

- All edges $x_e > 0$
- No leaf is unmatched

$\forall$ basic feasible x : $\exists$ e : x(N(e) < 3

For ease of analysis, assume that

- All edges $x_e > 0$
- No leaf is unmatched

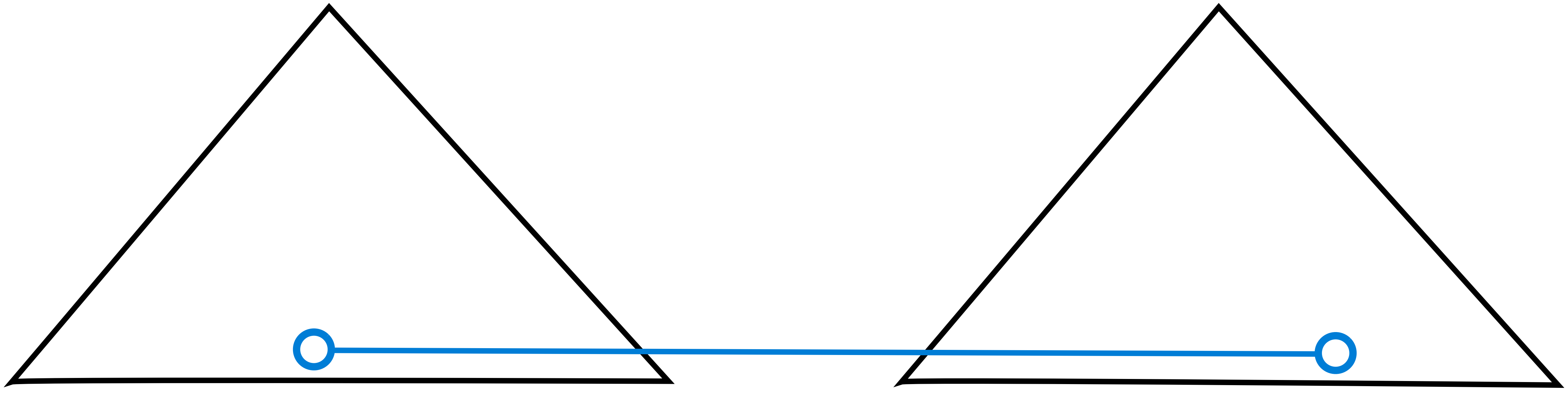
If we have a leaf-to-leaf edge, we are done

$\forall$ basic feasible x : $\exists$ e : $x(N(e)) < 3$

For ease of analysis, assume that

- All edges $x_e > 0$
- No leaf is unmatched

If we have a leaf-to-leaf edge, we are done



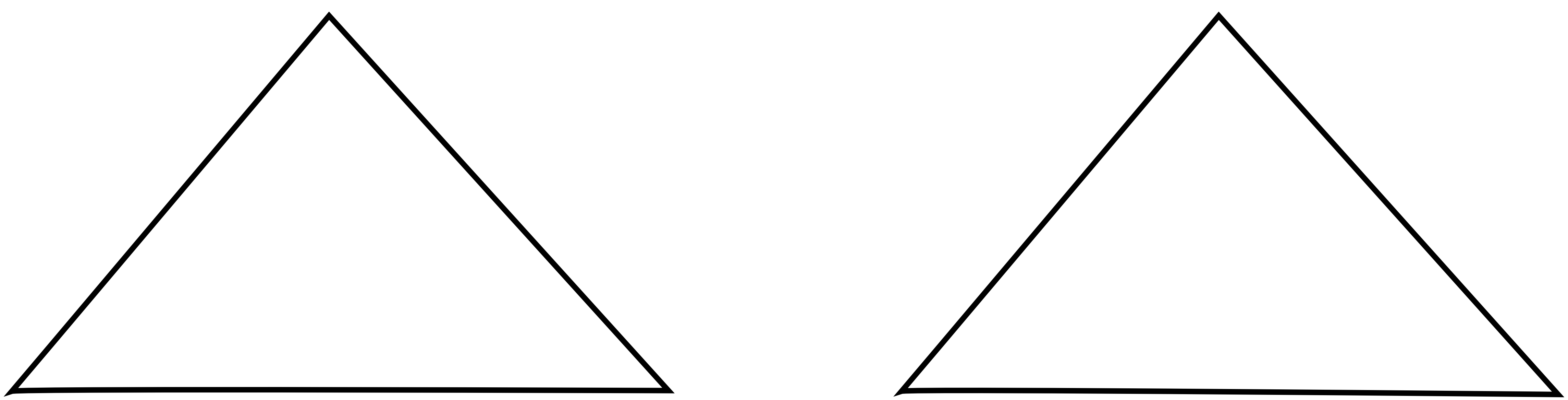
$\forall$ basic feasible $x : \exists\ e : x(N(e) < 3$

For ease of analysis, assume that

- All edges $x_e > 0$
- No leaf is unmatched

If we have a leaf-to-leaf edge, we are done

Otherwise, no internal-to-internal edges



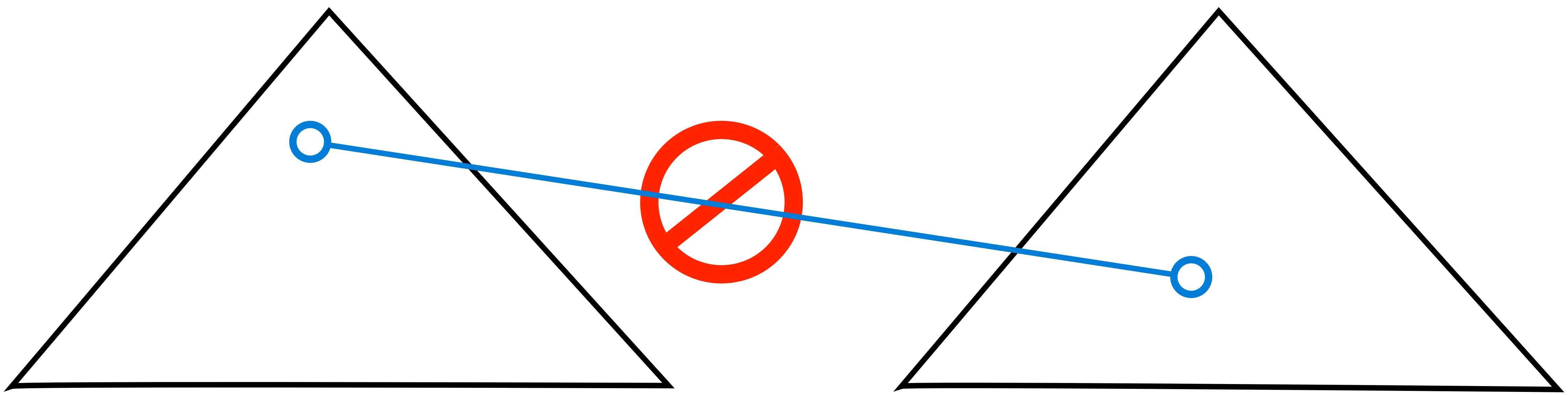
$\forall$ basic feasible x : $\exists$ e : x(N(e) < 3

For ease of analysis, assume that

- All edges $x_e > 0$
- No leaf is unmatched

If we have a leaf-to-leaf edge, we are done

Otherwise, no internal-to-internal edges



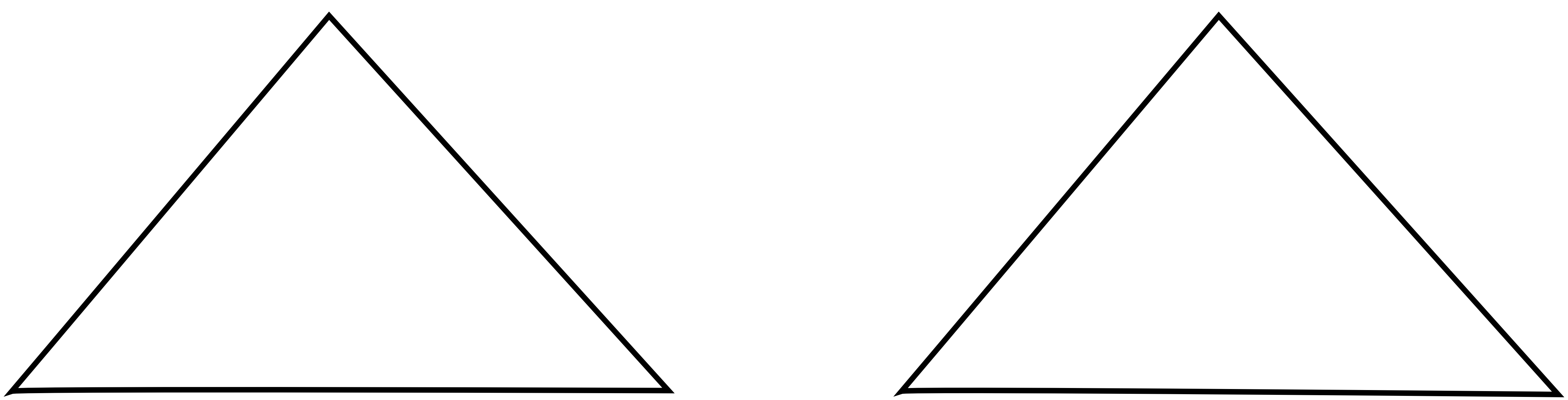
$\forall$ basic feasible $x : \exists\ e : x(N(e) < 3$

For ease of analysis, assume that

- All edges $x_e > 0$
- No leaf is unmatched

If we have a leaf-to-leaf edge, we are done

Otherwise, no internal-to-internal edges



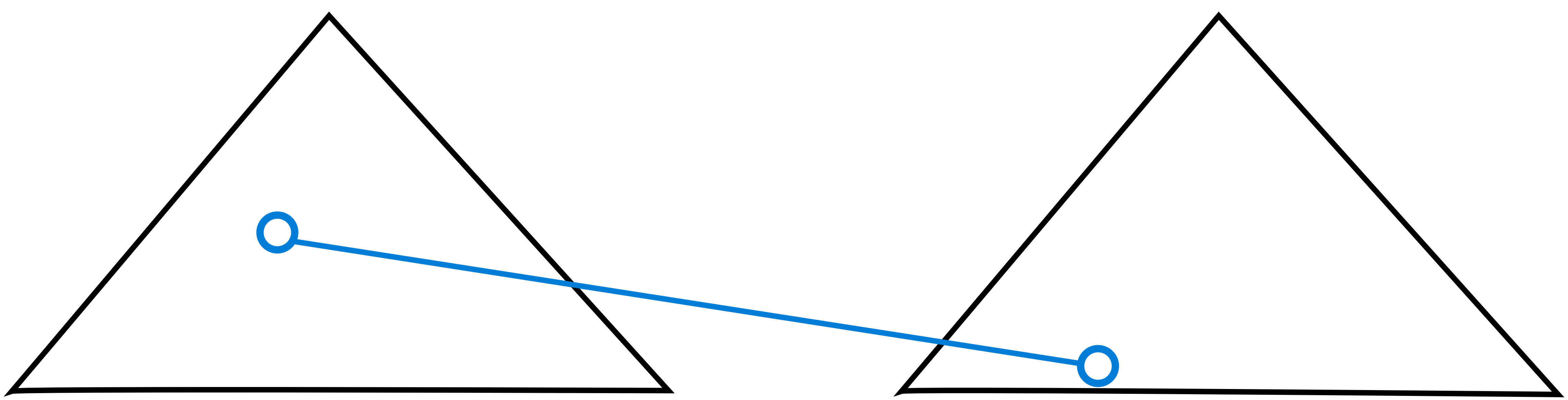
$\forall$ basic feasible $x : \exists\ e : x(N(e)) < 3$

For ease of analysis, assume that

- All edges $x_e > 0$
- No leaf is unmatched

If we have a leaf-to-leaf edge, we are done

Otherwise, no internal-to-internal edges



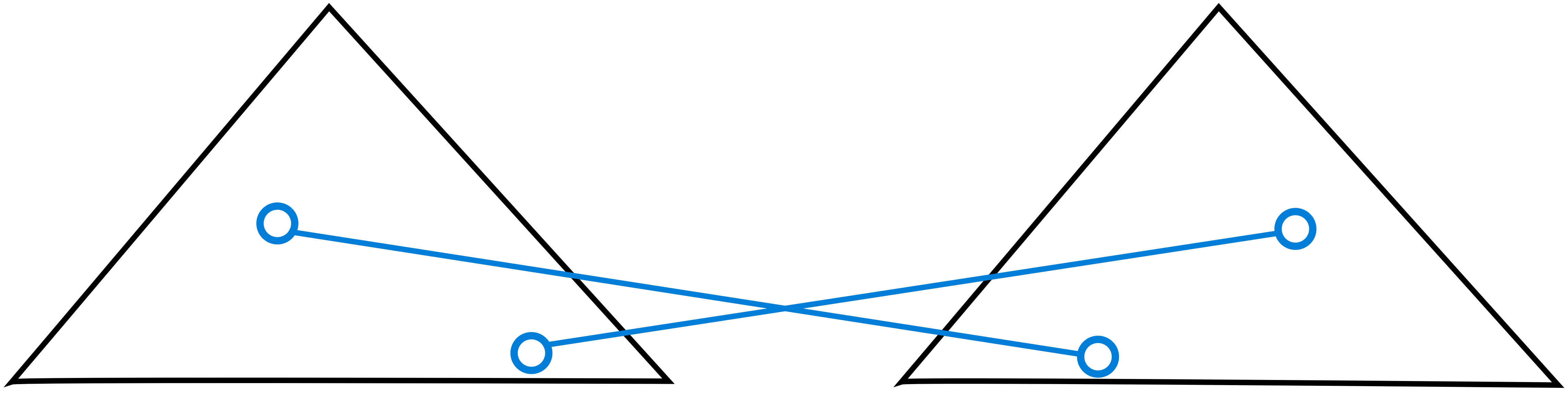
∀ basic feasible x : ∃ e : x(N(e) < 3

For ease of analysis, assume that

- All edges $x_e > 0$
- No leaf is unmatched

If we have a leaf-to-leaf edge, we are done

Otherwise, no internal-to-internal edges



Generalization to posets

Generalization to posets

To model uncertainty in hierarchical clustering, we can use posets instead of tress

Generalization to posets

To model uncertainty in hierarchical clustering, we can use posets instead of tress

We want matched vertices to be incomparable

Generalization to posets

To model uncertainty in hierarchical clustering, we can use posets instead of tress

We want matched vertices to be incomparable

Give 4 rho approximation, where rho is a parameter of poset

Generalization to posets

To model uncertainty in hierarchical clustering, we can use posets instead of tress

We want matched vertices to be incomparable

Give 4 rho approximation, where rho is a parameter of poset

It cannot be approximated to exp ($\log^{1-\varepsilon}$ rho), for any fixed $\varepsilon>0$, unless NP belongs to DTIME($n^{\text{polylog } n}$)

Thank you for your attention!