Lower bounds for tropical circuits and the power of pure dynamic programming

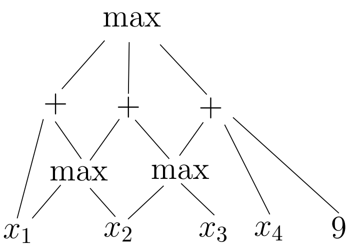
Tuukka Korhonen

UNIVERSITY OF COPENHAGEN

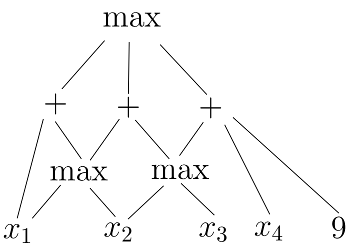
WACT 2026

5 June 2026

Tropical circuits

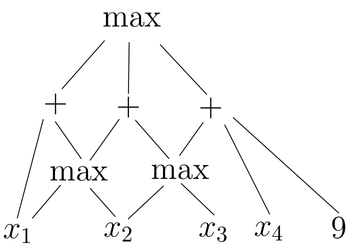


Tropical circuits



$$\max(2x_1, x_1 + x_2, x_1 + x_3, 2x_2, x_2 + x_3, x_2 + x_4 + 9, x_3 + x_4 + 9)$$

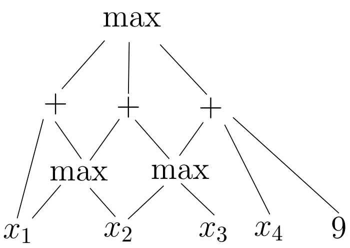
Tropical circuits



$$\max(2x_1, x_1 + x_2, x_1 + x_3, 2x_2, x_2 + x_3, x_2 + x_4 + 9, x_3 + x_4 + 9)$$

- Tropical circuits: Circuits over the $(\max, +)$-semiring (or the $(\min, +)$-semiring)

Tropical circuits



$$\max(2x_1, x_1 + x_2, x_1 + x_3, 2x_2, x_2 + x_3, x_2 + x_4 + 9, x_3 + x_4 + 9)$$

- Tropical circuits: Circuits over the $(\max, +)$-semiring (or the $(\min, +)$-semiring)
- First studied by [Jerrum & Snir, '82], recently by [Jukna '16, Jukna '17, Jukna & Seiwert '19, Jukna & Seiwert '20, K. '21, Jukna '23, Kluk & Nederlof '25])

Pure dynamic programming

Motivation: Tropical circuits model *pure dynamic programming algorithms* for optimization problems

Pure dynamic programming

Motivation: Tropical circuits model *pure dynamic programming algorithms* for optimization problems

- Shortest s-t-path: $\min_P \sum_{e \in P} x_e$, where x_e are weights of edges of a complete n-vertex graph with two designated vertices s and t, and P ranges over s-t-paths

Pure dynamic programming

Motivation: Tropical circuits model *pure dynamic programming algorithms* for optimization problems

- Shortest s-t-path: $\min_P \sum_{e \in P} x_e$, where x_e are weights of edges of a complete n-vertex graph with two designated vertices s and t, and P ranges over s-t-paths
- Maximum perfect bipartite matching: $\max_M \sum_{e \in M} x_e$, where x_e are weights of edges of a complete bipartite graph with n vertices on each side, and M ranges over perfect matchings

Pure dynamic programming

Motivation: Tropical circuits model *pure dynamic programming algorithms* for optimization problems

- Shortest s-t-path: $\min_P \sum_{e \in P} x_e$, where x_e are weights of edges of a complete n-vertex graph with two designated vertices s and t, and P ranges over s-t-paths
- Maximum perfect bipartite matching: $\max_M \sum_{e \in M} x_e$, where x_e are weights of edges of a complete bipartite graph with n vertices on each side, and M ranges over perfect matchings

Both polynomial-time solvable, but only shortest s-t-path has polynomial-size tropical circuits

Pure dynamic programming

Motivation: Tropical circuits model *pure dynamic programming algorithms* for optimization problems

- Shortest s-t-path: $\min_P \sum_{e \in P} x_e$, where x_e are weights of edges of a complete n-vertex graph with two designated vertices s and t, and P ranges over s-t-paths
- Maximum perfect bipartite matching: $\max_M \sum_{e \in M} x_e$, where x_e are weights of edges of a complete bipartite graph with n vertices on each side, and M ranges over perfect matchings

Both polynomial-time solvable, but only shortest s-t-path has polynomial-size tropical circuits

- Shortest s-t-path: $\mathcal{O}(n^3)$-size circuit from the Bellman-Ford algorithm

Pure dynamic programming

Motivation: Tropical circuits model *pure dynamic programming algorithms* for optimization problems

- Shortest s-t-path: $\min_P \sum_{e \in P} x_e$, where x_e are weights of edges of a complete n-vertex graph with two designated vertices s and t, and P ranges over s-t-paths
- Maximum perfect bipartite matching: $\max_M \sum_{e \in M} x_e$, where x_e are weights of edges of a complete bipartite graph with n vertices on each side, and M ranges over perfect matchings

Both polynomial-time solvable, but only shortest s-t-path has polynomial-size tropical circuits

- Shortest s-t-path: $\mathcal{O}(n^3)$-size circuit from the Bellman-Ford algorithm
- Maximum perfect bipartite matching: $2^{\Omega(n)}$-size required [Jerrum & Snir, '82]

Pure dynamic programming

Motivation: Tropical circuits model *pure dynamic programming algorithms* for optimization problems

- Shortest s-t-path: $\min_P \sum_{e \in P} x_e$, where x_e are weights of edges of a complete n-vertex graph with two designated vertices s and t, and P ranges over s-t-paths
- Maximum perfect bipartite matching: $\max_M \sum_{e \in M} x_e$, where x_e are weights of edges of a complete bipartite graph with n vertices on each side, and M ranges over perfect matchings

Both polynomial-time solvable, but only shortest s-t-path has polynomial-size tropical circuits

- Shortest s-t-path: $\mathcal{O}(n^3)$-size circuit from the Bellman-Ford algorithm
- Maximum perfect bipartite matching: $2^{\Omega(n)}$-size required [Jerrum & Snir, '82]

Important: Can inputs be negative?

Pure dynamic programming

Motivation: Tropical circuits model *pure dynamic programming algorithms* for optimization problems

- Shortest s-t-path: $\min_P \sum_{e \in P} x_e$, where x_e are weights of edges of a complete n-vertex graph with two designated vertices s and t, and P ranges over s-t-paths
- Maximum perfect bipartite matching: $\max_M \sum_{e \in M} x_e$, where x_e are weights of edges of a complete bipartite graph with n vertices on each side, and M ranges over perfect matchings

Both polynomial-time solvable, but only shortest s-t-path has polynomial-size tropical circuits

- Shortest s-t-path: $\mathcal{O}(n^3)$-size circuit from the Bellman-Ford algorithm
- Maximum perfect bipartite matching: $2^{\Omega(n)}$-size required [Jerrum & Snir, '82]

Important: Can inputs be negative?

- If yes, shortest s-t-path becomes hard [Jerrum & Snir, '82]

Pure dynamic programming

Motivation: Tropical circuits model *pure dynamic programming algorithms* for optimization problems

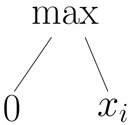
- Shortest s-t-path: $\min_P \sum_{e \in P} x_e$, where x_e are weights of edges of a complete n-vertex graph with two designated vertices s and t, and P ranges over s-t-paths
- Maximum perfect bipartite matching: $\max_M \sum_{e \in M} x_e$, where x_e are weights of edges of a complete bipartite graph with n vertices on each side, and M ranges over perfect matchings

Both polynomial-time solvable, but only shortest s-t-path has polynomial-size tropical circuits

- Shortest s-t-path: $\mathcal{O}(n^3)$-size circuit from the Bellman-Ford algorithm
- Maximum perfect bipartite matching: $2^{\Omega(n)}$-size required [Jerrum & Snir, '82]

Important: Can inputs be negative?

- If yes, shortest s-t-path becomes hard [Jerrum & Snir, '82]
- For $(\max, +)$ over downward-closed solution sets, can assume wlog that yes



Pure dynamic programming

Motivation: Tropical circuits model *pure dynamic programming algorithms* for optimization problems

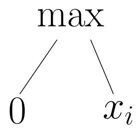
- Shortest s-t-path: $\min_P \sum_{e \in P} x_e$, where x_e are weights of edges of a complete n-vertex graph with two designated vertices s and t, and P ranges over s-t-paths
- Maximum perfect bipartite matching: $\max_M \sum_{e \in M} x_e$, where x_e are weights of edges of a complete bipartite graph with n vertices on each side, and M ranges over perfect matchings

Both polynomial-time solvable, but only shortest s-t-path has polynomial-size tropical circuits

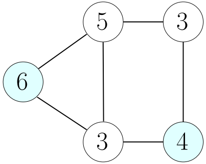
- Shortest s-t-path: $\mathcal{O}(n^3)$-size circuit from the Bellman-Ford algorithm
- Maximum perfect bipartite matching: $2^{\Omega(n)}$-size required [Jerrum & Snir, '82]

Important: Can inputs be negative?

- If yes, shortest s-t-path becomes hard [Jerrum & Snir, '82]
- For $(\max, +)$ over downward-closed solution sets, can assume wlog that yes
- Rest of this talk: Assume inputs can be negative (they are arbitrary reals)

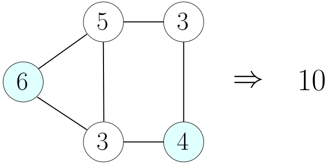


Maximum weight independent set (MWIS)

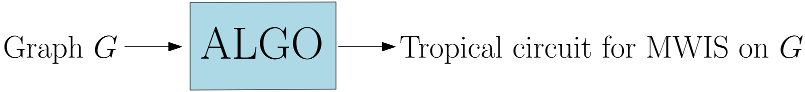


$\Rightarrow \quad 10$

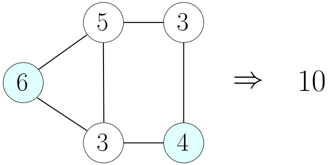
Maximum weight independent set (MWIS)



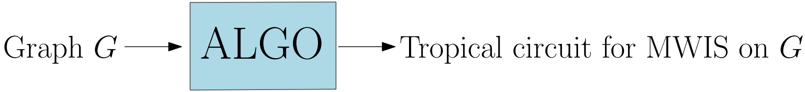
Many algorithms for MWIS produce tropical circuits (implicitly)



Maximum weight independent set (MWIS)



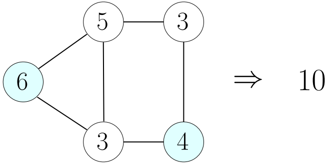
Many algorithms for MWIS produce tropical circuits (implicitly)



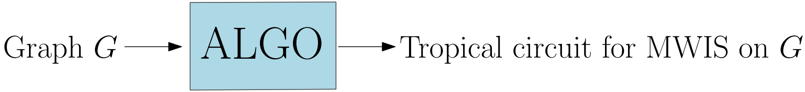
Examples:

- Bounded treewidth
- Bounded cliquewidth
- Chordal graphs
- P_5-free graphs
- Graphs with polynomial number of minimal separators

Maximum weight independent set (MWIS)



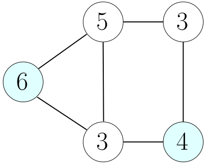
Many algorithms for MWIS produce tropical circuits (implicitly)



Examples:

- Bounded treewidth
- Bounded cliquewidth
- Chordal graphs
- P_5-free graphs
- Graphs with polynomial number of minimal separators

Maximum weight independent set (MWIS)



$\Rightarrow \quad 10$

Many algorithms for MWIS produce tropical circuits (implicitly)

Graph G → ALGO → Tropical circuit for MWIS on G

Examples:

- Bounded treewidth
- Bounded cliquewidth
- Chordal graphs
- P_5-free graphs
- Graphs with polynomial number of minimal separators

Non-examples:

- Perfect graphs (incl. bipartite)
- Line graphs (and their generalizations)

Lower bounds for MWIS

(My) ultimate goal: Understand how the hardness of a problem depends on input structure

Lower bounds for MWIS

(My) ultimate goal: Understand how the hardness of a problem depends on input structure

Very difficult!

Lower bounds for MWIS

(My) ultimate goal: Understand how the hardness of a problem depends on input structure

Very difficult!

Easier goal: Understand how the complexity of pure dynamic programming for MWIS depends on the graph G

Lower bounds for MWIS

(My) ultimate goal: Understand how the hardness of a problem depends on input structure

Very difficult!

Easier goal: Understand how the complexity of pure dynamic programming for MWIS depends on the graph G

Let $\tau(G)$ be the size of the smallest tropical circuit for MWIS on G

Lower bounds for MWIS

(My) ultimate goal: Understand how the hardness of a problem depends on input structure

Very difficult!

Easier goal: Understand how the complexity of pure dynamic programming for MWIS depends on the graph G

Let $\tau(G)$ be the size of the smallest tropical circuit for MWIS on G

Theorem (K., '21)

For **every** n-vertex graph G with treewidth k and max degree Δ:

$$2^{\Omega(k/\Delta)} \leq \tau(G) \leq 2^{\mathcal{O}(k)} n$$

Lower bounds for MWIS

(My) ultimate goal: Understand how the hardness of a problem depends on input structure

Very difficult!

Easier goal: Understand how the complexity of pure dynamic programming for MWIS depends on the graph G

Let $\tau(G)$ be the size of the smallest tropical circuit for MWIS on G

Theorem (K., '21)

For **every** n-vertex graph G with treewidth k and max degree Δ:

$$2^{\Omega(k/\Delta)} \leq \tau(G) \leq 2^{\mathcal{O}(k)} n$$

Theorem (K., '21)

For **every** n-vertex planar graph G with treewidth k:

$$2^{\Omega(k)} \leq \tau(G) \leq 2^{\mathcal{O}(k)} n$$

Proof techniques

Proof techniques

Step 1: Write the tropical polynomial as sum-of-products

$$\max(A_1 + B_1, A_2 + B_2, \ldots, A_s + B_s)$$

Proof techniques

Step 1: Write the tropical polynomial as sum-of-products

$$\max(A_1 + B_1, A_2 + B_2, \ldots, A_s + B_s)$$

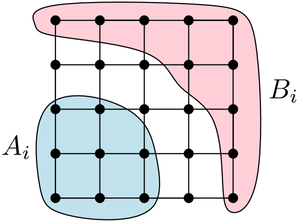
where:

- $s = \tau(G)$ is the size of the circuit (e.g. [Hyafil'79, Valiant'80])

Proof techniques

Step 1: Write the tropical polynomial as sum-of-products

$$\max(A_1 + B_1, A_2 + B_2, \ldots, A_s + B_s)$$

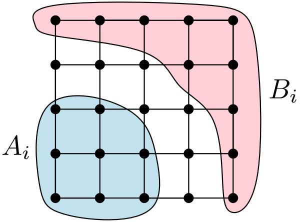


where:

- $s = \tau(G)$ is the size of the circuit (e.g. [Hyafil'79, Valiant'80])
- each product $A_i + B_i$ corresponds to a vertex-separator of size $\geq \Omega(\text{tw}(G))$

Proof techniques

Step 1: Write the tropical polynomial as sum-of-products



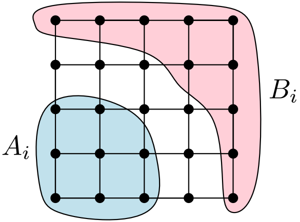
$$\max(A_1 + B_1, A_2 + B_2, \dots, A_s + B_s)$$

where:

- $s = \tau(G)$ is the size of the circuit (e.g. [Hyafil'79, Valiant'80])
- each product $A_i + B_i$ corresponds to a vertex-separator of size $\geq \Omega(\mathrm{tw(G)})$
- $2^{\Omega(\mathrm{tw}(G)/\Delta(G))}$ lower bound via Lovász local lemma

Proof techniques

Step 1: Write the tropical polynomial as sum-of-products



$$\max(A_1 + B_1, A_2 + B_2, \ldots, A_s + B_s)$$

where:

- $s = \tau(G)$ is the size of the circuit (e.g. [Hyafil'79, Valiant'80])
- each product $A_i + B_i$ corresponds to a vertex-separator of size $\geq \Omega(\text{tw(G)})$

- $2^{\Omega(\text{tw}(G)/\Delta(G))}$ lower bound via Lovász local lemma
- $2^{\Omega(\text{tw}(G))}$ lower bound for planar graphs from $\Omega(\text{tw}(G)) \times \Omega(\text{tw}(G))$ grid induced minors

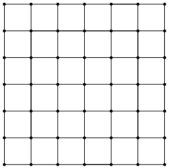
Open problems

Open: Understand $\tau(G)$ for all graphs G

Open problems

Open: Understand $\tau(G)$ for all graphs G

Conjecture: (Quasi)polynomial $\tau(G)$ characterized by excluded *grid induced minors*

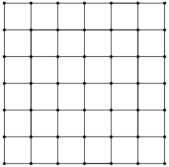


Open problems

Open: Understand $\tau(G)$ for all graphs G

Conjecture: (Quasi)polynomial $\tau(G)$ characterized by excluded *grid induced minors*

Open: Show that MWIS is (quasi)polynomial-time on graphs excluding constant-size grid induced minors [Dallard-Milanič-Štorgel ’21, Gartland-Lokshtanov ’21, K. ’21]

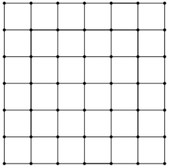


Open problems

Open: Understand $\tau(G)$ for all graphs G

Conjecture: (Quasi)polynomial $\tau(G)$ characterized by excluded *grid induced minors*

Open: Show that MWIS is (quasi)polynomial-time on graphs excluding constant-size grid induced minors [Dallard-Milanič-Štorgel '21, Gartland-Lokshtanov '21, K. '21]



Family of related conjectures and lot of partial progress [K. '23], [Bonamy, Bonnet, Déprés, Esperet, Geniet, Hilaire, Thomassé, Wesolek '23], [K. & Lokshtanov '24], [Chudnovsky, Gartland, Hajebi, Lokshtanov, Spirkl '25]

TSP on graphs of bounded treewidth

TSP on graphs of bounded treewidth

- "Straightforward" DP for solving TSP on treewidth-k graphs: $2^{\mathcal{O}(k \log k)} n$

TSP on graphs of bounded treewidth

- "Straightforward" DP for solving TSP on treewidth-k graphs: $2^{\mathcal{O}(k \log k)} n$
- Advanced techniques: $2^{\mathcal{O}(k)} n$ [Cygan, Nederlof, Pilipczuk, Pilipczuk, Rooij, Wojtaszczyk '11]

TSP on graphs of bounded treewidth

- "Straightforward" DP for solving TSP on treewidth-k graphs: $2^{\mathcal{O}(k \log k)} n$
- Advanced techniques: $2^{\mathcal{O}(k)} n$ [Cygan, Nederlof, Pilipczuk, Pilipczuk, Rooij, Wojtaszczyk '11]

Theorem (Kluk, Nederlof '25)

For each k, exists a treewidth-k graph with $k^{\mathcal{O}(1)}$ vertices that requires $2^{\Omega(k \log \log k)}$ size tropical circuits for TSP

TSP on graphs of bounded treewidth

- "Straightforward" DP for solving TSP on treewidth-k graphs: $2^{\mathcal{O}(k \log k)} n$
- Advanced techniques: $2^{\mathcal{O}(k)} n$ [Cygan, Nederlof, Pilipczuk, Pilipczuk, Rooij, Wojtaszczyk '11]

Theorem (Kluk, Nederlof '25)

For each k, exists a treewidth-k graph with $k^{\mathcal{O}(1)}$ vertices that requires $2^{\Omega(k \log \log k)}$ size tropical circuits for TSP

Proven by using $2^{\Omega(k \log \log k)}$ lower bound on the nondeterministic communication complexity of the matchings compatibility matrix [Raz & Spieker '95]

TSP on graphs of bounded treewidth

- "Straightforward" DP for solving TSP on treewidth-k graphs: $2^{\mathcal{O}(k \log k)} n$
- Advanced techniques: $2^{\mathcal{O}(k)} n$ [Cygan, Nederlof, Pilipczuk, Pilipczuk, Rooij, Wojtaszczyk '11]

Theorem (Kluk, Nederlof '25)

For each k, exists a treewidth-k graph with $k^{\mathcal{O}(1)}$ vertices that requires $2^{\Omega(k \log \log k)}$ size tropical circuits for TSP

Proven by using $2^{\Omega(k \log \log k)}$ lower bound on the nondeterministic communication complexity of the matchings compatibility matrix [Raz & Spieker '95]

Open: Improve to $2^{\Omega(k \log k)}$

Tropical vs monotone arithmetic

Lower bound for tropical circuits $\Rightarrow$ lower bound for monotone arithmetic circuits

Tropical vs monotone arithmetic

Lower bound for tropical circuits $\Rightarrow$ lower bound for monotone arithmetic circuits

Most of upper bounds for tropical circuits $\Rightarrow$ upper bounds for monotone arithmetic

Tropical vs monotone arithmetic

Lower bound for tropical circuits $\Rightarrow$ lower bound for monotone arithmetic circuits

Most of upper bounds for tropical circuits $\Rightarrow$ upper bounds for monotone arithmetic

- If we can optimize efficiently, can we count efficiently?

Tropical vs monotone arithmetic

Lower bound for tropical circuits $\Rightarrow$ lower bound for monotone arithmetic circuits

Most of upper bounds for tropical circuits $\Rightarrow$ upper bounds for monotone arithmetic

- If we can optimize efficiently, can we count efficiently?
- Exists graph classes with poly-size tropical circuits for MWIS, but for which counting independent sets in polynomial time is open

Tropical vs monotone arithmetic

Lower bound for tropical circuits $\Rightarrow$ lower bound for monotone arithmetic circuits

Most of upper bounds for tropical circuits $\Rightarrow$ upper bounds for monotone arithmetic

- If we can optimize efficiently, can we count efficiently?
- Exists graph classes with poly-size tropical circuits for MWIS, but for which counting independent sets in polynomial time is open
  - No such example where counting is known to be #P-hard!

Tropical vs monotone arithmetic

Lower bound for tropical circuits ⇒ lower bound for monotone arithmetic circuits

Most of upper bounds for tropical circuits ⇒ upper bounds for monotone arithmetic

- If we can optimize efficiently, can we count efficiently?
- Exists graph classes with poly-size tropical circuits for MWIS, but for which counting independent sets in polynomial time is open
  - No such example where counting is known to be #P-hard!
- Does there exists a polynomial s.t:

Tropical vs monotone arithmetic

Lower bound for tropical circuits $\Rightarrow$ lower bound for monotone arithmetic circuits

Most of upper bounds for tropical circuits $\Rightarrow$ upper bounds for monotone arithmetic

- If we can optimize efficiently, can we count efficiently?
- Exists graph classes with poly-size tropical circuits for MWIS, but for which counting independent sets in polynomial time is open
  - No such example where counting is known to be #P-hard!
- Does there exists a polynomial s.t:
  - Multilinear, $0, 1$-coefficients

Tropical vs monotone arithmetic

Lower bound for tropical circuits $\Rightarrow$ lower bound for monotone arithmetic circuits

Most of upper bounds for tropical circuits $\Rightarrow$ upper bounds for monotone arithmetic

- If we can optimize efficiently, can we count efficiently?
- Exists graph classes with poly-size tropical circuits for MWIS, but for which counting independent sets in polynomial time is open
  - No such example where counting is known to be #P-hard!
- Does there exists a polynomial s.t:
  - Multilinear, $0, 1$-coefficients
  - On the $(+, \times)$-ring, hard for monotone arithmetic circuits

Tropical vs monotone arithmetic

Lower bound for tropical circuits $\Rightarrow$ lower bound for monotone arithmetic circuits

Most of upper bounds for tropical circuits $\Rightarrow$ upper bounds for monotone arithmetic

- If we can optimize efficiently, can we count efficiently?
- Exists graph classes with poly-size tropical circuits for MWIS, but for which counting independent sets in polynomial time is open
  - No such example where counting is known to be #P-hard!
- Does there exists a polynomial s.t:
  - Multilinear, $0, 1$-coefficients
  - On the $(+, \times)$-ring, hard for monotone arithmetic circuits
  - On the $(\max, +)$-semiring, easy for tropical circuits (that work for negative weights)

Tropical vs monotone arithmetic

Lower bound for tropical circuits $\Rightarrow$ lower bound for monotone arithmetic circuits

Most of upper bounds for tropical circuits $\Rightarrow$ upper bounds for monotone arithmetic

- If we can optimize efficiently, can we count efficiently?
- Exists graph classes with poly-size tropical circuits for MWIS, but for which counting independent sets in polynomial time is open
  - No such example where counting is known to be #P-hard!
- Does there exists a polynomial s.t:
  - Multilinear, $0, 1$-coefficients
  - On the $(+, \times)$-ring, hard for monotone arithmetic circuits
  - On the $(\max, +)$-semiring, easy for tropical circuits (that work for negative weights)
- The power of idempotence

Tropical vs monotone arithmetic

Lower bound for tropical circuits $\Rightarrow$ lower bound for monotone arithmetic circuits

Most of upper bounds for tropical circuits $\Rightarrow$ upper bounds for monotone arithmetic

- If we can optimize efficiently, can we count efficiently?
- Exists graph classes with poly-size tropical circuits for MWIS, but for which counting independent sets in polynomial time is open
  - No such example where counting is known to be #P-hard!
- Does there exists a polynomial s.t:
  - Multilinear, $0, 1$-coefficients
  - On the $(+, \times)$-ring, hard for monotone arithmetic circuits
  - On the $(\max, +)$-semiring, easy for tropical circuits (that work for negative weights)
- The power of idempotence
- Quasipolynomial separation should be doable, but exponential not clear

Conclusion

People care about tropical circuits because they model pure dynamic programming

Conclusion

People care about tropical circuits because they model pure dynamic programming

Open problems:

Conclusion

People care about tropical circuits because they model pure dynamic programming

Open problems:

1. Characterize which graphs have poly-size tropical circuits for MWIS

Conclusion

People care about tropical circuits because they model pure dynamic programming

Open problems:

1. Characterize which graphs have poly-size tropical circuits for MWIS
2. Prove $2^{\Omega(k \log k)}$ lower bound for TSP on treewidth-k graphs

Conclusion

People care about tropical circuits because they model pure dynamic programming

Open problems:

1. Characterize which graphs have poly-size tropical circuits for MWIS
2. Prove $2^{\Omega(k \log k)}$ lower bound for TSP on treewidth-k graphs
3. Separation between tropical and monotone arithmetic circuits for multilinear polynomials with $0, 1$-coefficients

Conclusion

People care about tropical circuits because they model pure dynamic programming

Open problems:

1. Characterize which graphs have poly-size tropical circuits for MWIS
2. Prove $2^{\Omega(k \log k)}$ lower bound for TSP on treewidth-k graphs
3. Separation between tropical and monotone arithmetic circuits for multilinear polynomials with $0, 1$-coefficients

Thank you!