Polynomial Time Algorithms in Invariant Theory for Torus Actions

M. Levent Doğan

21.04.2022

Based on a joint work with P. Bürgisser, V. Makam, M. Walter, A. Wigderson

Permanent vs Determinant

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \cdots & \cdots & a_{nn} \end{bmatrix}$$

Permanent vs Determinant

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix}$$

- $\text{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$
- $\det(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} \text{sgn}(\sigma) a_{i\sigma(i)}$

Permanent vs Determinant

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix}$$

- $\text{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$
- $\text{per}\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = ad + bc$

- $\det(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} \text{sgn}(\sigma) a_{i\sigma(i)}$
- $\det\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = ad - bc$

Permanent vs Determinant

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix}$$

- $\mathrm{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$
- $\mathrm{per}(\begin{bmatrix} a & b \\ c & d \end{bmatrix}) = ad + bc$
- $\mathrm{per}(LAR) = \mathrm{per}(L)\mathrm{per}(A)\mathrm{per}(R)$, whenever L, R are diagonal matrices.

- $\det(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} \mathrm{sgn}(\sigma) a_{i\sigma(i)}$
- $\det(\begin{bmatrix} a & b \\ c & d \end{bmatrix}) = ad - bc$
- $\det(LAR) = \det(L)\det(A)\det(R)$, for every $L, R \in \mathbb{C}^{n \times n}$

Permanent vs Determinant

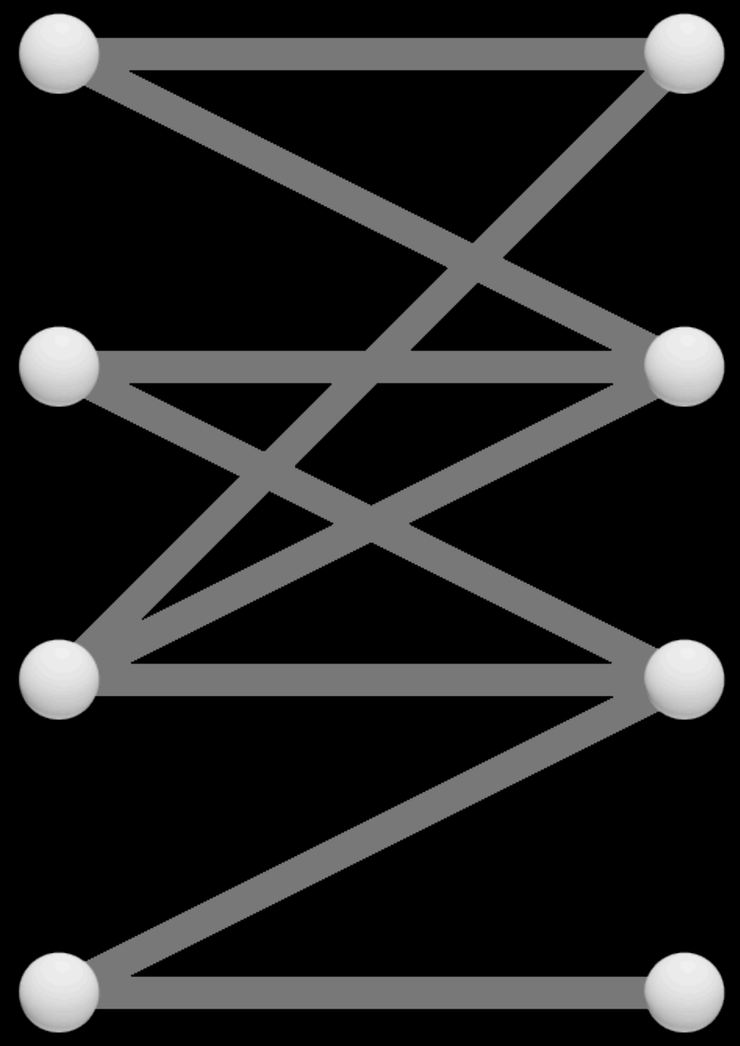
$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix}$$

- $\mathrm{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$
- $\mathrm{per}(\begin{bmatrix} a & b \\ c & d \end{bmatrix}) = ad + bc$
- $\mathrm{per}(LAR) = \mathrm{per}(L)\mathrm{per}(A)\mathrm{per}(R)$, whenever L, R are diagonal matrices.
- Matrix scaling

- $\det(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} \mathrm{sgn}(\sigma) a_{i\sigma(i)}$
- $\det(\begin{bmatrix} a & b \\ c & d \end{bmatrix}) = ad - bc$
- $\det(LAR) = \det(L)\det(A)\det(R)$, for every $L, R \in \mathbb{C}^{n \times n}$
- Gaussian elimination

$$\text{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$$

G



Biadjacency matrix →

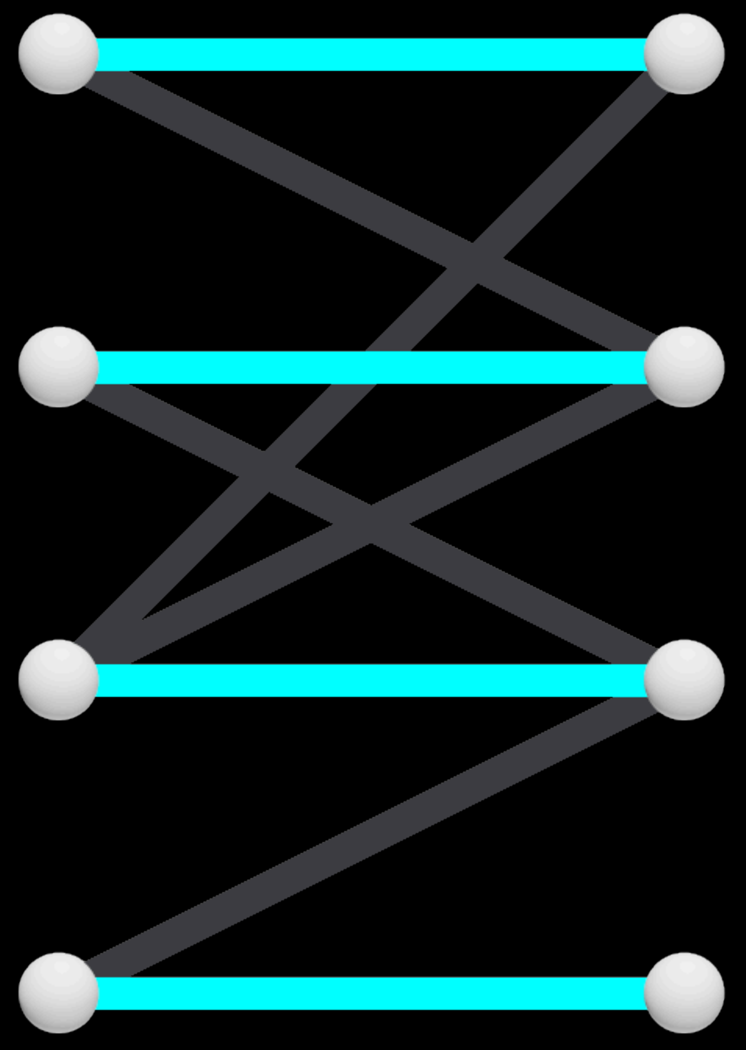
$\text{Adj}(G)$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$\text{per}(\text{Adj}(G)) =$

$$\mathrm{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$$

G



Biadjacency matrix →

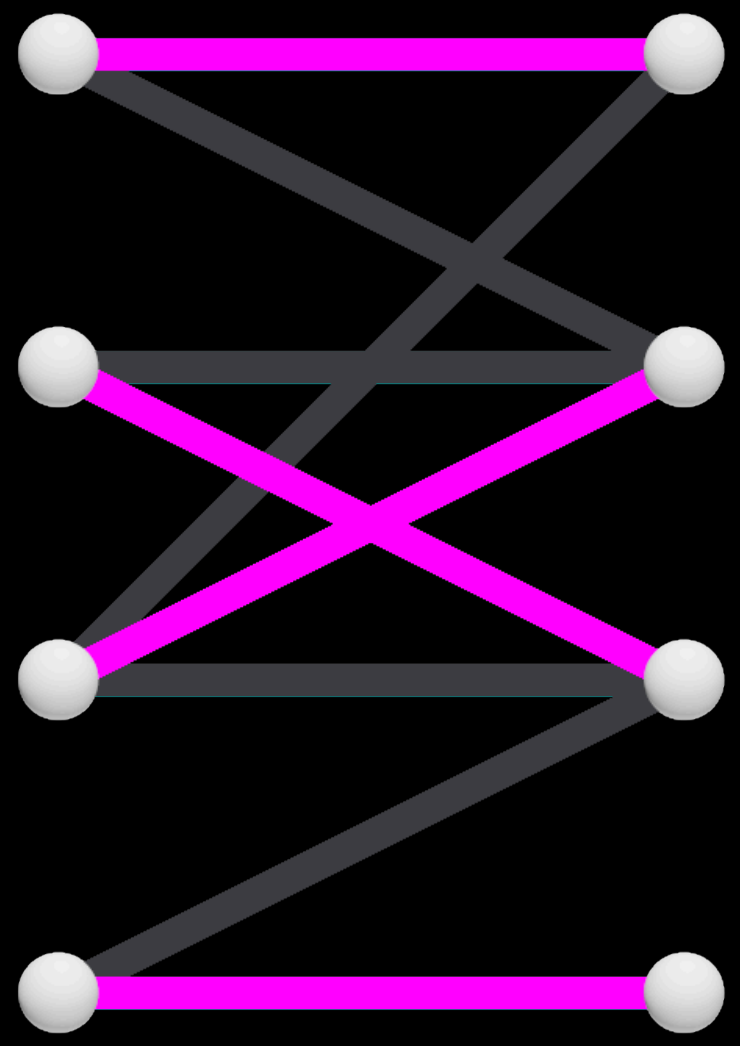
$\mathrm{Adj}(G)$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathrm{per}(\mathrm{Adj}(G)) = 1$$

$$\mathrm{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$$

G



Biadjacency matrix →

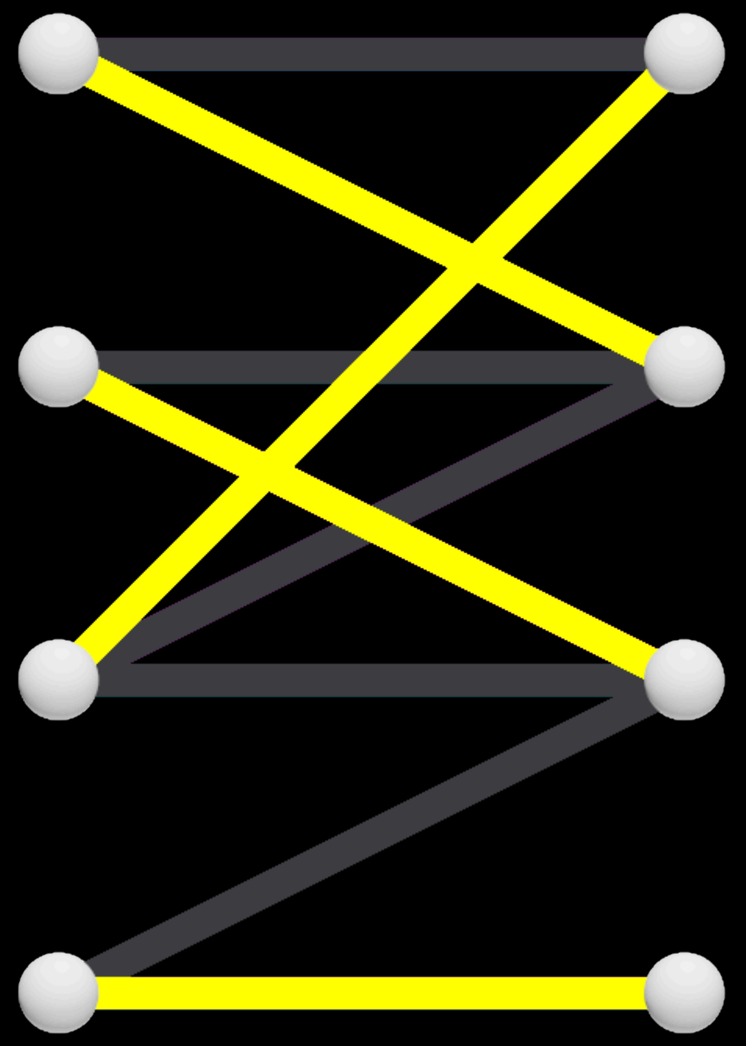
$\mathrm{Adj}(G)$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathrm{per}(\mathrm{Adj}(G)) = 1 + 1$$

$$\mathrm{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$$

G



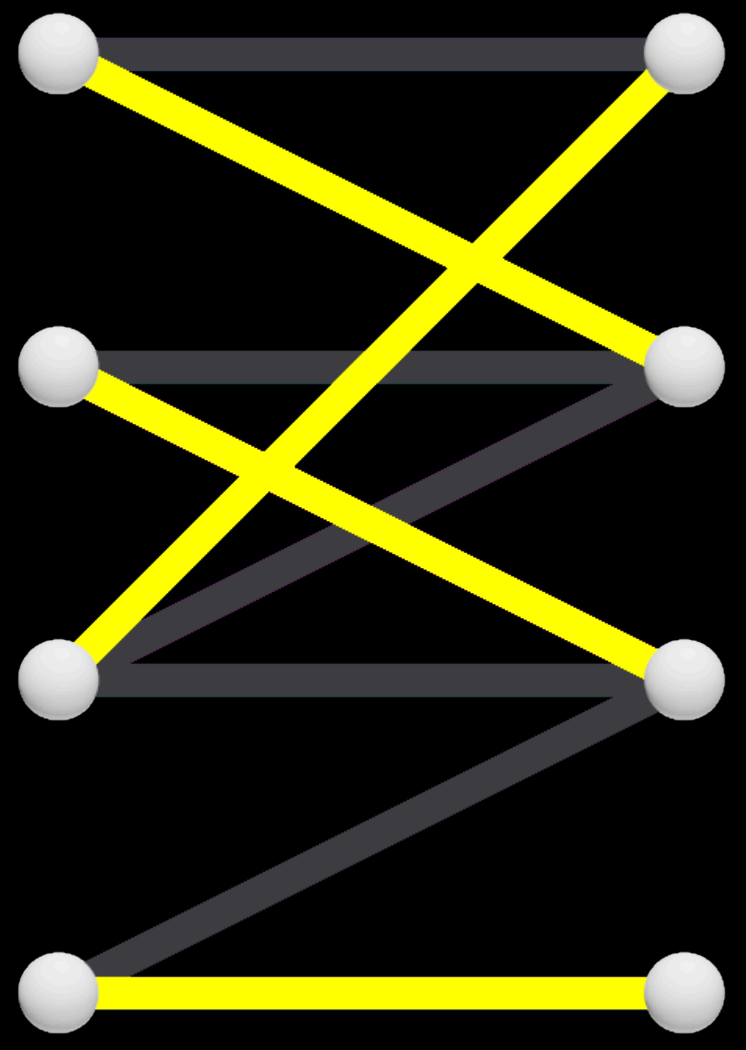
Biadjacency matrix →

$\mathrm{Adj}(G)$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathrm{per}(\mathrm{Adj}(G)) = 1 + 1 + 1$$
$$= 3$$

G



Biadjacency matrix →

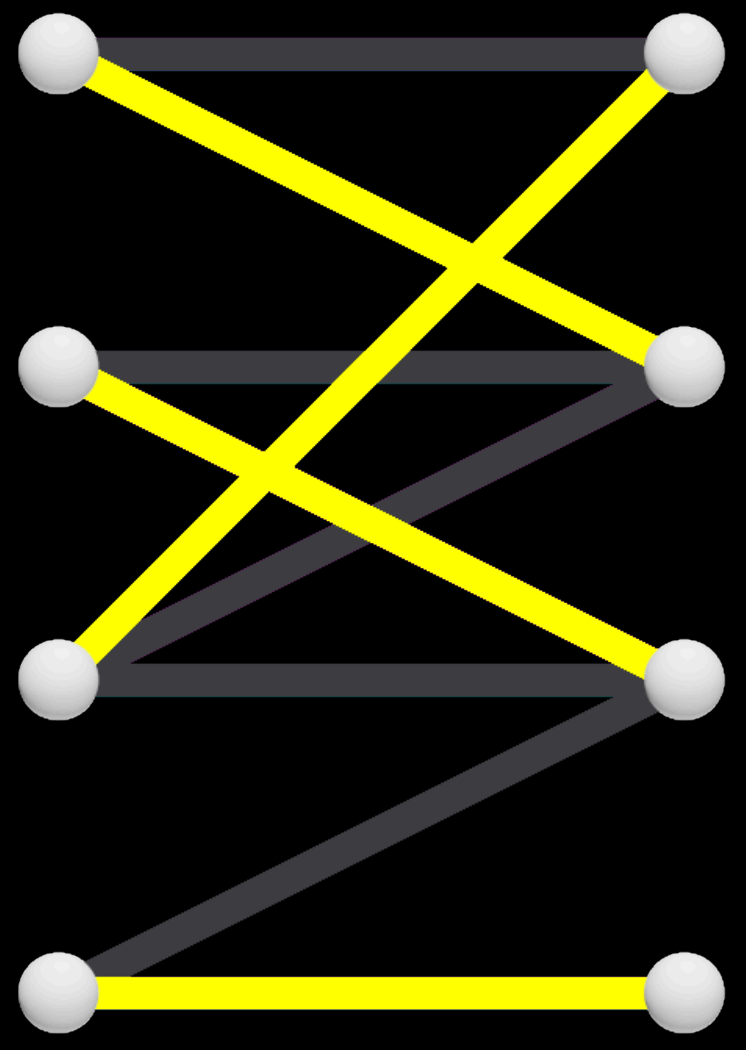
$\mathrm{Adj}(G)$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathrm{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$$

$$\mathrm{per}(\mathrm{Adj}(G)) = 1 + 1 + 1 = 3$$

- **Fact:** $\mathrm{per}(\mathrm{Adj}(G)) =$ #perfect matchings of G.



Biadjacency matrix →

Adj(G)

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathrm{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$$

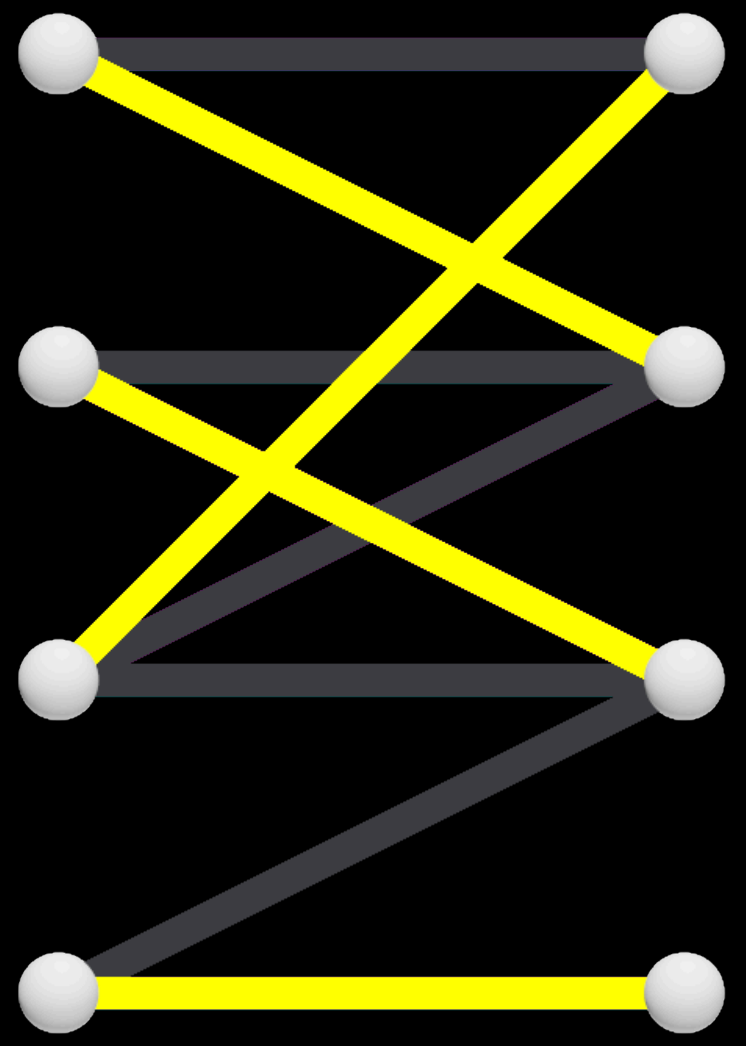
$$\mathrm{per}(\mathrm{Adj}(G)) = 1 + 1 + 1 = 3$$

- **Fact:** $\mathrm{per}(\mathrm{Adj}(G))$ = #perfect matchings of G.

- **Theorem (Valiant '79) :** The complexity of computing the permanent of an $n \times n$ $(0,1)-$matrix is #P-complete.

$$\mathrm{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$$

G



Biadjacency matrix →

$\mathrm{Adj}(G)$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathrm{per}(\mathrm{Adj}(G)) = 1 + 1 + 1 = 3$$

- Fact: $\mathrm{per}(\mathrm{Adj}(G)) =$ #perfect matchings of G.
- Theorem (Valiant '79) : The complexity of computing the permanent of an $n \times n$ $(0,1)$–matrix is #P-complete.
- Even though the decision problem is in P !

- $$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix}$$

$$\mathbf{r}(A) = (\sum_{i=1}^{n} a_{1i}, \dots, \sum_{i=1}^{n} a_{ni})$$: vector of row sums

$$\mathbf{c}(A) = (\sum_{i=1}^{n} a_{i1}, \dots, \sum_{i=1}^{n} a_{in})$$: vector of column sums

- $$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix}$$

$$\mathbf{r}(A) = (\sum_{i=1}^{n} a_{1i}, \dots, \sum_{i=1}^{n} a_{ni})$$: vector of row sums

$$\mathbf{c}(A) = (\sum_{i=1}^{n} a_{i1}, \dots, \sum_{i=1}^{n} a_{in})$$: vector of column sums

- Van der Waerden's conjecture (Egoritsjev '80, Falikman '80):
If $A \in \mathbb{R}^{n \times n}$ is a positive, *doubly stochastic* matrix, then
$$\text{per}(A) \geq \frac{n!}{n^n}$$

$$\frac{1}{n} J_n = \begin{bmatrix} 1/n & 1/n & \dots & 1/n \\ 1/n & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ 1/n & \dots & \dots & 1/n \end{bmatrix}$$

$$\mathbf{r} = \mathbf{1}$$
$$\mathbf{c} = \mathbf{1}$$

- $$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix}$$

$\mathbf{r}(A) = (\sum_{i=1}^{n} a_{1i}, \dots, \sum_{i=1}^{n} a_{ni})$: vector of row sums

$\mathbf{c}(A) = (\sum_{i=1}^{n} a_{i1}, \dots, \sum_{i=1}^{n} a_{in})$: vector of column sums

- Van der Waerden's conjecture (Egoritsjev '80, Falikman '80):
  If $A \in \mathbb{R}^{n \times n}$ is a positive, *doubly stochastic* matrix, then
  $$\text{per}(A) \geq \frac{n!}{n^n}$$

- Assume $A \in \mathbb{R}^{n \times n}$ is a positive $(\mathbf{r}, \mathbf{c})-$matrix, L, R are positive diagonal matrices.
  $\Rightarrow$ LAR is a $(L\mathbf{r}(AR), R\mathbf{c}(LA))-$matrix.

- $A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix}$

$$\mathbf{r}(A) = (\sum_{i=1}^{n} a_{1i}, \dots, \sum_{i=1}^{n} a_{ni}) \quad : \text{ vector of row sums}$$

$$\mathbf{c}(A) = (\sum_{i=1}^{n} a_{i1}, \dots, \sum_{i=1}^{n} a_{in}) \quad : \text{ vector of column sums}$$

- Van der Waerden's conjecture (Egoritsjev '80, Falikman '80):
  If $A \in \mathbb{R}^{n \times n}$ is a positive, *doubly stochastic* matrix, then

$$\text{per}(A) \geq \frac{n!}{n^n}$$

- Assume $A \in \mathbb{R}^{n \times n}$ is a positive $(\mathbf{r}, \mathbf{c})-$matrix, L, R are positive diagonal matrices.
  $\Rightarrow \quad LAR$ is a $(L\mathbf{r}(AR), R\mathbf{c}(LA))-$matrix.

- Idea: If there exist L, R with LAR doubly stochastic then

$$\text{per}(A) = \frac{\text{per}(LAR)}{\text{per}(L)\text{per}(R)} \geq \frac{n!}{\text{per}(L)\text{per}(R)n^n}$$

Matrix Scaling

Matrix Scaling

- Definition: A is called $(\mathbf{r}, \mathbf{c})$-*scalable* if there exist positive diagonal matrices L, R such that
$$(\mathbf{r}(LAR), \mathbf{c}(LAR)) = (\mathbf{r}, \mathbf{c})\,.$$

Matrix Scaling

- **Definition:** A is called $(\mathbf{r}, \mathbf{c})$-*scalable* if there exist positive diagonal matrices L, R such that

$$(\mathbf{r}(LAR), \mathbf{c}(LAR)) = (\mathbf{r}, \mathbf{c})\,.$$

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Matrix Scaling

- Definition: A is called $(\mathbf{r}, \mathbf{c})$-*scalable* if there exist positive diagonal matrices L, R such that
$$(\mathbf{r}(LAR), \mathbf{c}(LAR)) = (\mathbf{r}, \mathbf{c}) \, .$$

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

Not scalable, $\mathrm{per}(A) = 0.$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Scalable, $L = \begin{bmatrix} 1/2 & & \\ & 1/2 & \\ & & 1 \end{bmatrix}$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Almost scalable, $\begin{bmatrix} \varepsilon^{-1} & & \\ & \varepsilon^{-1} & \\ & & \varepsilon^{2} \end{bmatrix} A \begin{bmatrix} \varepsilon & & \\ & \varepsilon & \\ & & \varepsilon^{-2} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ \varepsilon^3 & \varepsilon^3 & 1 \end{bmatrix}$

Matrix Scaling

- Definition: A is called $(\mathbf{r}, \mathbf{c})$-*scalable* if there exist positive diagonal matrices L, R such that
$$(\mathbf{r}(LAR), \mathbf{c}(LAR)) = (\mathbf{r}, \mathbf{c}) .$$

- Definition: A is called *almost* $(\mathbf{r}, \mathbf{c})$-*scalable* if for every $\varepsilon > 0$ there exist positive diagonal matrices L, R such that
$$(\mathbf{r}(LAR), \mathbf{c}(LAR)) = (\mathbf{r}, \mathbf{c}')$$
with $\|c - c'\|_2 < \varepsilon$.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

Not scalable, $\mathrm{per}(A) = 0.$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Scalable, $L = \begin{bmatrix} 1/2 & & \\ & 1/2 & \\ & & 1 \end{bmatrix}$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Almost scalable, $\begin{bmatrix} \varepsilon^{-1} & & \\ & \varepsilon^{-1} & \\ & & \varepsilon^{2} \end{bmatrix} A \begin{bmatrix} \varepsilon & & \\ & \varepsilon & \\ & & \varepsilon^{-2} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ \varepsilon^3 & \varepsilon^3 & 1 \end{bmatrix}$

Matrix Scaling

- **Definition:** A is called $(\mathbf{r}, \mathbf{c})$-*scalable* if there exist positive diagonal matrices L, R such that
$$(\mathbf{r}(LAR), \mathbf{c}(LAR)) = (\mathbf{r}, \mathbf{c}) .$$
- **Definition:** A is called *almost* $(\mathbf{r}, \mathbf{c})$-*scalable* if for every $\varepsilon > 0$ there exist positive diagonal matrices L, R such that
$$(\mathbf{r}(LAR), \mathbf{c}(LAR)) = (\mathbf{r}, \mathbf{c}')$$
with $\|c - c'\|_2 < \varepsilon$.
- **Questions:**
  1) Decide if a given A is scalable/almost scalable.
  2) Find the scaling $LAR = B$.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

Not scalable, $\mathrm{per}(A) = 0$.

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Scalable, $L = \begin{bmatrix} 1/2 & & \\ & 1/2 & \\ & & 1 \end{bmatrix}$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Almost scalable, $\begin{bmatrix} \varepsilon^{-1} & & \\ & \varepsilon^{-1} & \\ & & \varepsilon^{2} \end{bmatrix} A \begin{bmatrix} \varepsilon & & \\ & \varepsilon & \\ & & \varepsilon^{-2} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ \varepsilon^3 & \varepsilon^3 & 1 \end{bmatrix}$

Matrix Scaling

- **Definition:** A is called $(\mathbf{r},\mathbf{c})$-*scalable* if there exist positive diagonal matrices L, R such that

$$(\mathbf{r}(LAR), \mathbf{c}(LAR)) = (\mathbf{r}, \mathbf{c}) .$$

- **Definition:** A is called *almost* $(\mathbf{r},\mathbf{c})$-*scalable* if for every $\varepsilon > 0$ there exist positive diagonal matrices L, R such that

$$(\mathbf{r}(LAR), \mathbf{c}(LAR)) = (\mathbf{r}, \mathbf{c}')$$

with $\|c - c'\|_2 < \varepsilon$.

- **Questions:**
  1) Decide if a given A is scalable/almost scalable.
  2) Find the scaling $LAR = B$.

Max-flow-min-cut formulation

Sinkhorn-Knopp algorithm

[Linial,Samorodnitsky, Wigderson '04] : The first deterministic polynomial time approximation algorithm for the permanent.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

Not scalable, $\mathrm{per}(A) = 0$.

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Scalable, $L = \begin{bmatrix} 1/2 & & \\ & 1/2 & \\ & & 1 \end{bmatrix}$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Almost scalable, $\begin{bmatrix} \varepsilon^{-1} & & \\ & \varepsilon^{-1} & \\ & & \varepsilon^{2} \end{bmatrix} A \begin{bmatrix} \varepsilon & & \\ & \varepsilon & \\ & & \varepsilon^{-2} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ \varepsilon^3 & \varepsilon^3 & 1 \end{bmatrix}$

Scaling Actions

Scaling Actions

- $T = (\mathbb{C}^\times)^d$: $d-$dimensional *algebraic torus*
  $V = \mathbb{C}^n$: $n-$dimensional vector space

Scaling Actions

- $T = (\mathbb{C}^{\times})^d$: $d-$dimensional *algebraic torus*
  $V = \mathbb{C}^n$: $n-$dimensional vector space

- Tori = the family of connected, commutative, reductive groups.

Scaling Actions

- $T = (\mathbb{C}^\times)^d$: $d-$dimensional *algebraic torus*

  $V = \mathbb{C}^n$: $n-$dimensional vector space

- Tori = the family of connected, commutative, reductive groups.

Commutative groups stabilize a flag:

$$0 = V_0 \subset V_1 \subset V_2 \subset \ldots \subset V_n = \mathbb{C}^n.$$

Representations of reductive groups are completely reducible:

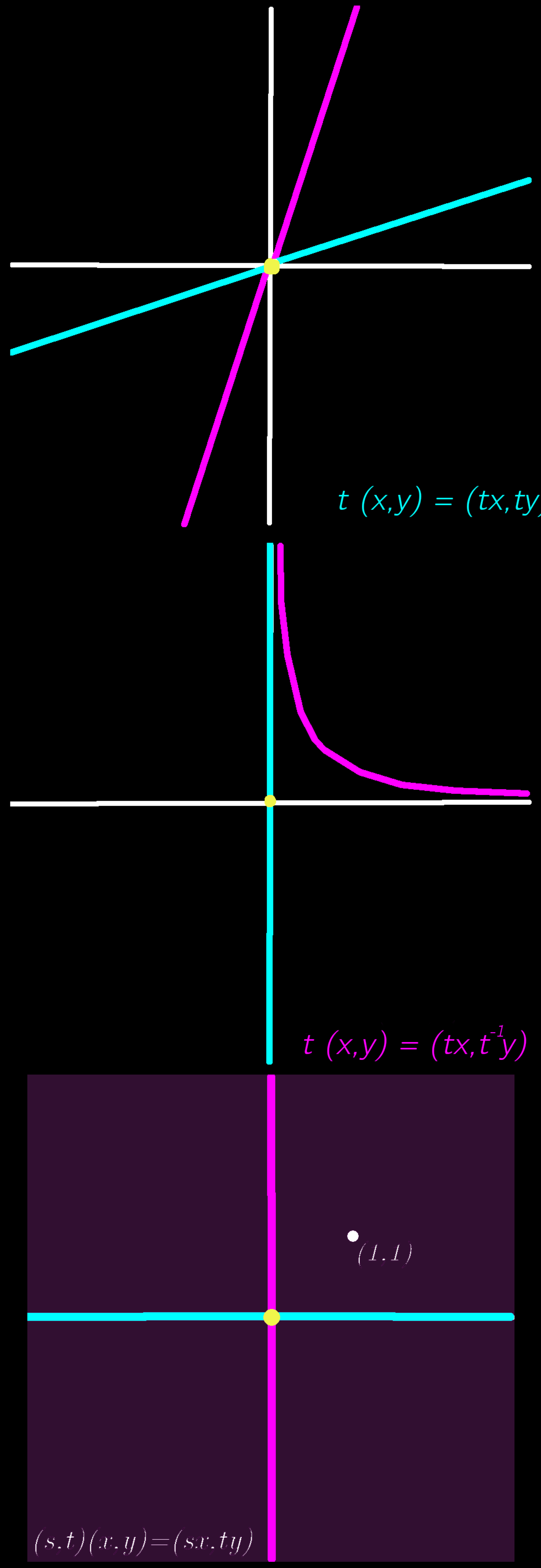
$$V = W_1 \oplus W_2 \oplus \ldots \oplus W_m$$

Scaling Actions

- $T = (\mathbb{C}^{\times})^d$: $d-$dimensional *algebraic torus*
  $V = \mathbb{C}^n$: $n-$dimensional vector space
- Tori = the family of connected, commutative, reductive groups.
- Suppose $\omega_1, \omega_2, \ldots, \omega_n \in \mathbb{Z}^d$:
  $(t_1, t_2, \ldots, t_d) \cdot (v_1, v_2, \ldots, v_n) = (t^{\omega_1} v_1, t^{\omega_2} v_2, \ldots, t^{\omega_n} v_n)$

Scaling Actions

- $T = (\mathbb{C}^{\times})^d$: $d-$dimensional *algebraic torus*
  $V = \mathbb{C}^n$: $n-$dimensional vector space
- Tori = the family of connected, commutative, reductive groups.
- Suppose $\omega_1, \omega_2, \ldots, \omega_n \in \mathbb{Z}^d$:
  $(t_1, t_2, \ldots, t_d) \cdot (v_1, v_2, \ldots, v_n) = (t^{\omega_1} v_1, t^{\omega_2} v_2, \ldots, t^{\omega_n} v_n)$
- Each coordinate v_i is *scaled* according to the *weight* ω_i.



Scaling Actions

- $T = (\mathbb{C}^{\times})^d$: $d-$dimensional *algebraic torus*
  $V = \mathbb{C}^n$: $n-$dimensional vector space

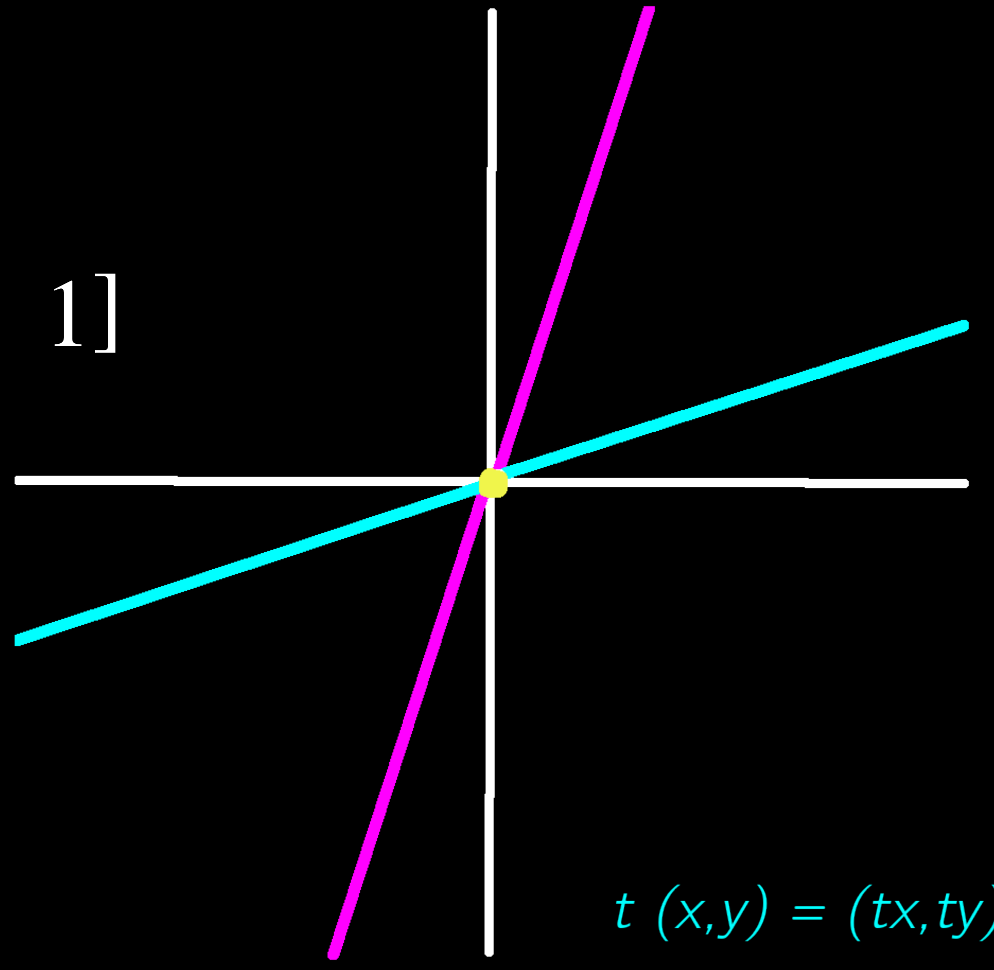
- Tori = the family of connected, commutative, reductive groups.

- Suppose $\omega_1, \omega_2, \ldots, \omega_n \in \mathbb{Z}^d$:
  $(t_1, t_2, \ldots, t_d) \cdot (v_1, v_2, \ldots, v_n) = (t^{\omega_1} v_1, t^{\omega_2} v_2, \ \ldots, t^{\omega_n} v_n)$

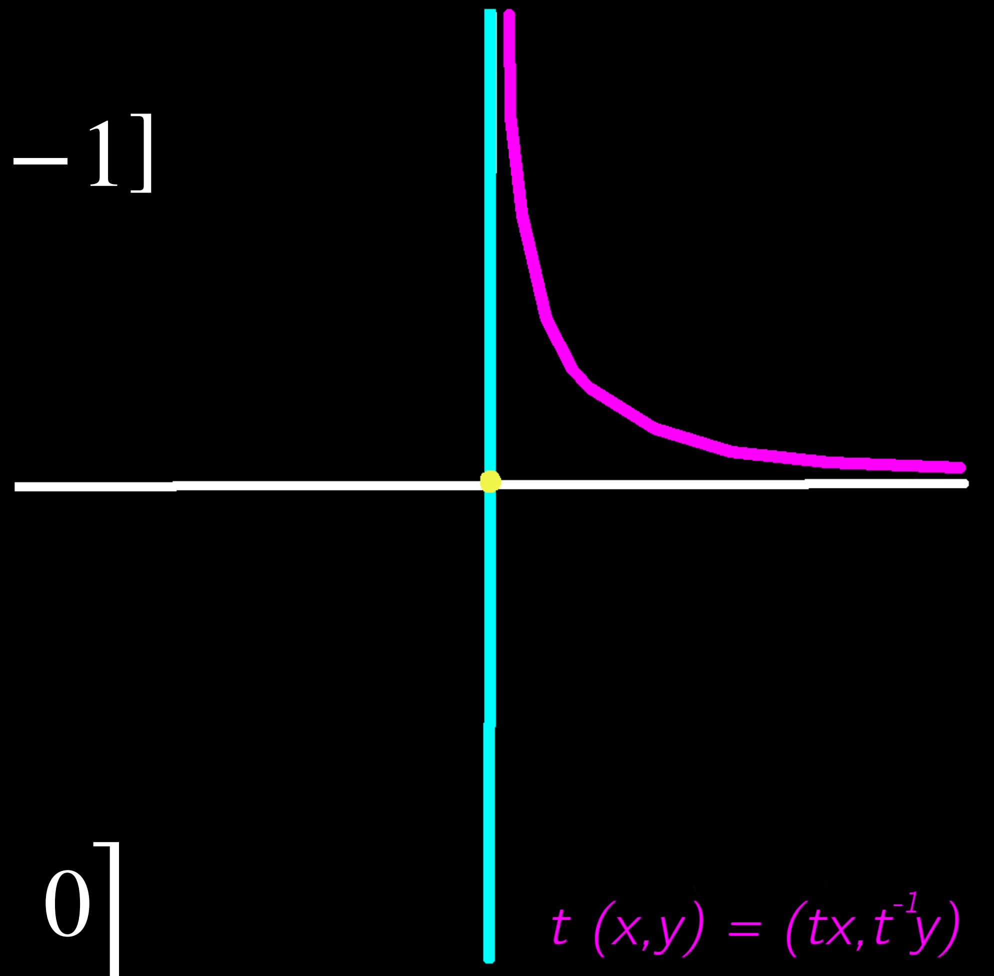
- Each coordinate v_i is *scaled* according to the *weight* ω_i.

- **Definition:** The *weight matrix* of $(\mathbb{C}^{\times})^d \curvearrowright \mathbb{C}^n$ is the integer matrix $M \in \mathbb{Z}^{d \times n}$ having ω_i as its i-th column.

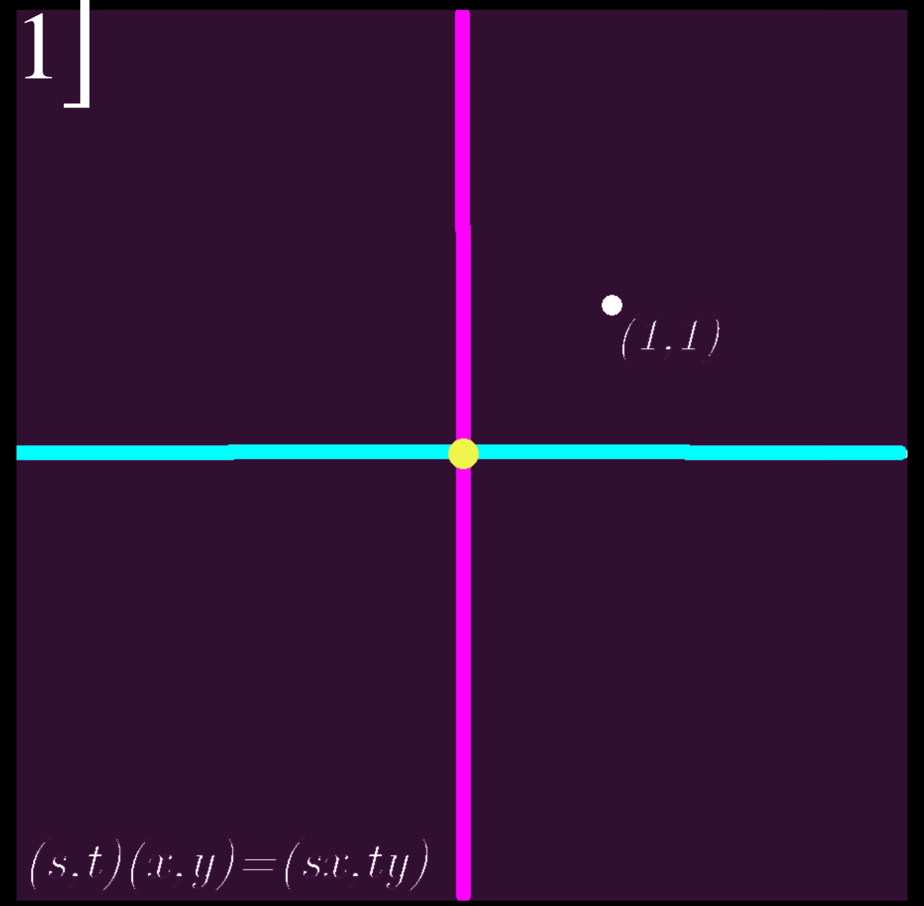
$M = [1 \quad 1]$



$M = [1 \quad -1]$



$M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$



- Assume $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and $v, w \in \mathbb{C}^n$.

- Assume $(\mathbb{C}^{\times})^d \curvearrowright \mathbb{C}^n$ and $v, w \in \mathbb{C}^n$.

- $Tv = \{tv \mid t \in T\}$: the orbit of v
  $\overline{Tv}$: the topological closure of Tv

- Assume $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and $v, w \in \mathbb{C}^n$.
- $Tv = \{tv \mid t \in T\}$: the orbit of v
  $\overline{Tv}$: the topological closure of Tv
- Orbit equality problem:
  Decide whether $Tv = Tw$.

- Assume $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and $v, w \in \mathbb{C}^n$.
- $Tv = \{tv \mid t \in T\}$: the orbit of v
  $\overline{Tv}$: the topological closure of Tv
- Orbit equality problem:
  Decide whether $Tv = Tw$.
- Orbit closure intersection problem:
  Decide whether $\overline{Tv} \cap \overline{Tw} = \emptyset$.

- Assume $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and $v, w \in \mathbb{C}^n$.
- $Tv = \{tv \mid t \in T\}$: the orbit of v
  $\overline{Tv}$: the topological closure of Tv
- Orbit equality problem:
  Decide whether $Tv = Tw$.
- Orbit closure intersection problem:
  Decide whether $\overline{Tv} \cap \overline{Tw} = \emptyset$.
- Orbit closure containment problem:
  Decide whether $w \in \overline{Tv}$.

- Assume $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and $v, w \in \mathbb{C}^n$.
- $Tv = \{tv \mid t \in T\}$: the orbit of v
  $\overline{Tv}$: the topological closure of Tv
- Orbit equality problem:
  Decide whether $Tv = Tw$.
- Orbit closure intersection problem:
  Decide whether $\overline{Tv} \cap \overline{Tw} = \emptyset$.
- Orbit closure containment problem:
  Decide whether $w \in \overline{Tv}$.
- Nullcone membership problem:
  Decide whether $0 \in \overline{Tv}$.

- Assume $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and $v, w \in \mathbb{C}^n$.
- $Tv = \{tv \mid t \in T\}$: the orbit of v
  $\overline{Tv}$: the topological closure of Tv
- Orbit equality problem:
  Decide whether $Tv = Tw$.
- Orbit closure intersection problem:
  Decide whether $\overline{Tv} \cap \overline{Tw} = \emptyset$.
- Orbit closure containment problem:
  Decide whether $w \in \overline{Tv}$.
- Nullcone membership problem:
  Decide whether $0 \in \overline{Tv}$.
- Capacity:
  $$\mathrm{cap}(v) := \inf_{t \in T} \|tv\|_2$$

- Assume $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and $v, w \in \mathbb{C}^n$.
- $Tv = \{tv \mid t \in T\}$: the orbit of v
  $\overline{Tv}$: the topological closure of Tv
- Orbit equality problem:
  Decide whether $Tv = Tw$.
- Orbit closure intersection problem:
  Decide whether $\overline{Tv} \cap \overline{Tw} = \emptyset$.
- Orbit closure containment problem:
  Decide whether $w \in \overline{Tv}$.
- Nullcone membership problem:
  Decide whether $0 \in \overline{Tv}$.

- Capacity:
  $\mathrm{cap}(v) := \inf_{t \in T} \|tv\|_2$
- $$F_v(t) := \|tv\|_2 = \sum_i |v_i|^2 |t|^{2\omega_i} = \sum_i |v_i|^2 \prod_{j=1}^{d} |t_j|^{2\omega_{ij}}$$

- Assume $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and $v, w \in \mathbb{C}^n$.
- $Tv = \{tv \mid t \in T\}$: the orbit of v
  $\overline{Tv}$: the topological closure of Tv
- **Orbit equality problem:**
  Decide whether $Tv = Tw$.
- **Orbit closure intersection problem:**
  Decide whether $\overline{Tv} \cap \overline{Tw} = \emptyset$.
- **Orbit closure containment problem:**
  Decide whether $w \in \overline{Tv}$.
- **Nullcone membership problem:**
  Decide whether $0 \in \overline{Tv}$.

- **Capacity:**
  $\mathrm{cap}(v) := \inf_{t \in T} \|tv\|_2$
- $F_v(t) := \|tv\|_2 = \sum_i |v_i|^2 |t|^{2\omega_i}$

$$= \sum_i |v_i|^2 \prod_{j=1}^{d} |t_j|^{2\omega_{ij}}$$

- Assume $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and $v, w \in \mathbb{C}^n$.
- $Tv = \{tv \mid t \in T\}$: the orbit of v
  $\overline{Tv}$: the topological closure of Tv
- **Orbit equality problem:**
  Decide whether $Tv = Tw$.
- **Orbit closure intersection problem:**
  Decide whether $\overline{Tv} \cap \overline{Tw} = \emptyset$.
- **Orbit closure containment problem:**
  Decide whether $w \in \overline{Tv}$.
- **Nullcone membership problem:**
  Decide whether $0 \in \overline{Tv}$.

- **Capacity:**
  $\mathrm{cap}(v) := \inf_{t \in T} \|tv\|_2$
- $F_v(t) := \|tv\|_2 = \sum_i |v_i|^2 |t|^{2\omega_i}$
  $= \sum_i |v_i|^2 \prod_{j=1}^{d} |t_j|^{2\omega_{ij}}$

Note: In the case of matrix scaling,
Nullcone membership ~ is the matrix scalable?
Orbit closure containment problem ~ is the matrix almost scalable?
Minimizing ∇F_v ~ $(\mathbf{r}, \mathbf{c})-$scaling.

- Unconstrained geometric program:

$$\begin{aligned} &\text{minimize} \;\; f(z) \\ &\text{sbj. to} \quad z \in \mathbb{R}^d, z > 0. \end{aligned}$$

where $f(z) = \sum_{i=1}^{n} q_i z^{\omega_i}$, for some positive $q > 0$ and $\omega_1, \ldots, \omega_n \in \mathbb{R}^d$.

- Unconstrained geometric program:

$$\begin{aligned} &\text{minimize} \quad f(z) \\ &\text{sbj. to} \quad z \in \mathbb{R}^d, z > 0. \end{aligned}$$

where $f(z) = \sum_{i=1}^{n} q_i z^{\omega_i}$, for some positive $q > 0$ and $\omega_1, \ldots, \omega_n \in \mathbb{R}^d$.

- $F(x) = \log f(e^x)$ is convex!

- Unconstrained geometric program:

$$\begin{aligned} &\text{minimize} \quad f(z) \\ &\text{sbj. to} \quad z \in \mathbb{R}^d, z > 0. \end{aligned}$$

where $f(z) = \sum_{i=1}^{n} q_i z^{\omega_i}$, for some positive $q > 0$ and $\omega_1, \ldots, \omega_n \in \mathbb{R}^d$.

- $F(x) = \log f(e^x)$ is convex!

- Compare to Gurvits' capacity optimization: $\mathrm{Cap}_{\mathbf{1}}(p) = \inf_{x>0} \frac{p(x)}{x_1 x_2 \ldots x_n}$

$$\log \mathrm{Cap}_{\mathbf{1}}(p) = \inf_{y \in \mathbb{R}^n} \left(F(y) - \langle y, \mathbf{1} \rangle\right).$$

- Unconstrained geometric program:

$$\begin{aligned} &\text{minimize} \quad f(z) \\ &\text{sbj. to} \quad z \in \mathbb{R}^d, z > 0. \end{aligned}$$

where $f(z) = \sum_{i=1}^{n} q_i z^{\omega_i}$, for some positive $q > 0$ and $\omega_1, \ldots, \omega_n \in \mathbb{R}^d$.

- $F(x) = \log f(e^x)$ is convex!

- Compare to Gurvits' capacity optimization: $\mathrm{Cap}_{\mathbf{1}}(p) = \inf_{x>0} \frac{p(x)}{x_1 x_2 \ldots x_n}$

$$\log \mathrm{Cap}_{\mathbf{1}}(p) = \inf_{y \in \mathbb{R}^n} \left(F(y) - \langle y, \mathbf{1} \rangle \right).$$

- Duality: $\inf_{x \in \mathbb{R}^n} F(x) = \sup\{-D_{KL}(p \,||\, q) \;|\; \sum_i p_i \omega_i = 0, \; \sum_i p_i = 1, p \geq 0\}$

where $D_{KL} = -\sum_i p_i \log \frac{p_i}{q_i}$ is the Kullback-Leibler divergence between distributions p, q.

- Unconstrained geometric program:

$$\begin{aligned}&\text{minimize} \;\; f(z)\\ &\text{sbj . to} \quad z \in \mathbb{R}^d, z > 0.\end{aligned}$$

where $f(z) = \sum_{i=1}^{n} q_i z^{\omega_i}$, for some positive $q > 0$ and $\omega_1, \ldots, \omega_n \in \mathbb{R}^d$.

- $F(x) = \log f(e^x)$ is convex!

- Compare to Gurvits' capacity optimization: $\mathrm{Cap}_1(p) = \inf_{x>0} \frac{p(x)}{x_1 x_2 \ldots x_n}$

$\log \mathrm{Cap}_1(p) = \inf_{y \in \mathbb{R}^n} \left(F(y) - \langle y, \mathbf{1} \rangle\right)$.

- **Duality:** $\inf_{x \in \mathbb{R}^n} F(x) = \sup\{-D_{KL}(p \,||\, q) \;\mid\; \sum_i p_i \omega_i = 0, \; \sum_i p_i = 1, p \geq 0\}$

where $D_{KL} = -\sum_i p_i \log \frac{p_i}{q_i}$ is the Kullback-Leibler divergence between distributions p, q.

~ entropy maximization (D_{KL} is the Shannon entropy when $q = \mathbf{1}$)

- Unconstrained geometric program:

$$\begin{aligned} &\text{minimize} \quad f(z) \\ &\text{sbj. to} \quad z \in \mathbb{R}^d, z > 0. \end{aligned}$$

where $f(z) = \sum_{i=1}^{n} q_i z^{\omega_i}$, for some positive $q > 0$ and $\omega_1, \ldots, \omega_n \in \mathbb{R}^d$.

- $F(x) = \log f(e^x)$ is convex!

- Compare to Gurvits' capacity optimization: $\mathrm{Cap}_{\mathbf{1}}(p) = \inf_{x>0} \frac{p(x)}{x_1 x_2 \ldots x_n}$

$\log \mathrm{Cap}_{\mathbf{1}}(p) = \inf_{y \in \mathbb{R}^n} (F(y) - \langle y, \mathbf{1} \rangle)$.

- Duality: $\inf_{x \in \mathbb{R}^n} F(x) = \sup\{-D_{KL}(p \,||\, q) \;\mid\; \sum_i p_i \omega_i = 0, \; \sum_i p_i = 1, p \geq 0\}$

where $D_{KL} = -\sum_i p_i \log \frac{p_i}{q_i}$ is the Kullback-Leibler divergence between distributions p, q .

- Both the primal and the dual problem can be solved by the means of convex optimization (ellipsoid method / interior-point method).

[Gurvits '04], [Kortanek, Xu, Ye '97], [Andersen, Ye '98], [Singh, Vishnoi '14], [Straszak,Vishnoi '19], [Bürgisser, Li, Nieuwboer, Walter '20]

- Unconstrained geometric program:

$$\begin{aligned} &\text{minimize} \quad f(z) \\ &\text{sbj. to} \quad z \in \mathbb{R}^d, z > 0. \end{aligned}$$

where $f(z) = \sum_{i=1}^{n} q_i z^{\omega_i}$, for some positive $q > 0$ and $\omega_1, \ldots, \omega_n \in \mathbb{R}^d$.

- $F(x) = \log f(e^x)$ is convex!

- Compare to Gurvits' capacity optimization: $\mathrm{Cap}_{\mathbf{1}}(p) = \inf_{x>0} \frac{p(x)}{x_1 x_2 \ldots x_n}$

$\log \mathrm{Cap}_{\mathbf{1}}(p) = \inf_{y \in \mathbb{R}^n} \left(F(y) - \langle y, \mathbf{1} \rangle\right).$

- Duality: $\inf_{x \in \mathbb{R}^n} F(x) = \sup\{-D_{KL}(p \,||\, q) \mid \sum_i p_i \omega_i = 0, \ \sum_i p_i = 1, p \geq 0\}$

where $D_{KL} = -\sum_i p_i \log \frac{p_i}{q_i}$ is the Kullback-Leibler divergence between distributions p, q.

- Both the primal and the dual problem can be solved by the means of convex optimization (ellipsoid method / interior-point method).
[Gurvits '04], [Kortanek, Xu, Ye '97], [Andersen, Ye '98], [Singh, Vishnoi '14], [Straszak,Vishnoi '19], [Bürgisser, Li, Nieuwboer, Walter '20]

- Ongoing research for non-commutative groups: [Bürgisser, Franks, Garg, Oliveira, Walter, Wigderson '19]

- Theorem (Bürgisser, D., Makam, Walter, Wigderson):
  Let $M \in \mathbb{Z}^{d \times n}$ be the weight matrix of $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and suppose $v, w \in \mathbb{Q}(i)^n$. Let b denote the maximum of the bit-lengths of the entries of v, w and M. Then, in poly(d, n, b) time we can decide orbit equality, orbit closure intersection and orbit closure containment.

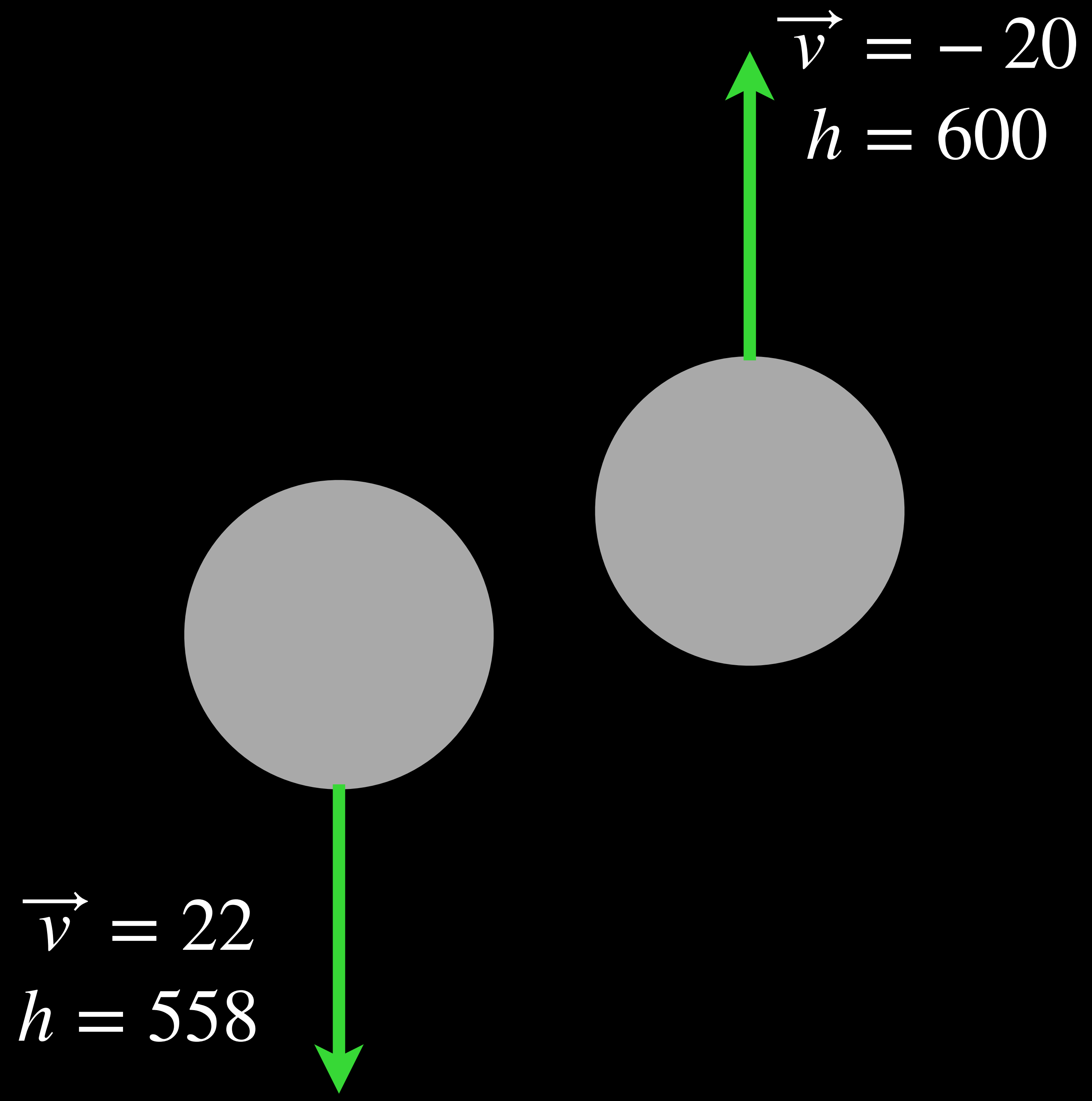
- Theorem (Bürgisser, D., Makam, Walter, Wigderson):
  Let $M \in \mathbb{Z}^{d \times n}$ be the weight matrix of $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and suppose $v, w \in \mathbb{Q}(i)^n$. Let b denote the maximum of the bit-lengths of the entries of v, w and M. Then, in poly(d, n, b) time we can decide orbit equality, orbit closure intersection and orbit closure containment.

- For non-commutative group actions:
  Orbit equality : as hard as graph isomorphism problem
  Orbit closure intersection : conjectured to be in P. Geometric complexity theory program [Mulmuley, Sohoni '01]
  Orbit closure containment : NP-hard / as hard as tensor rank [Bläser, Ikenmeyer, Lysikov, Pandey, Schreyer '21]

- Theorem (Bürgisser, D., Makam, Walter, Wigderson):
  Let $M \in \mathbb{Z}^{d \times n}$ be the weight matrix of $(\mathbb{C}^\times)^d \curvearrowright \mathbb{C}^n$ and suppose $v, w \in \mathbb{Q}(i)^n$. Let b denote the maximum of the bit-lengths of the entries of v, w and M. Then, in poly(d, n, b) time we can decide orbit equality, orbit closure intersection and orbit closure containment.

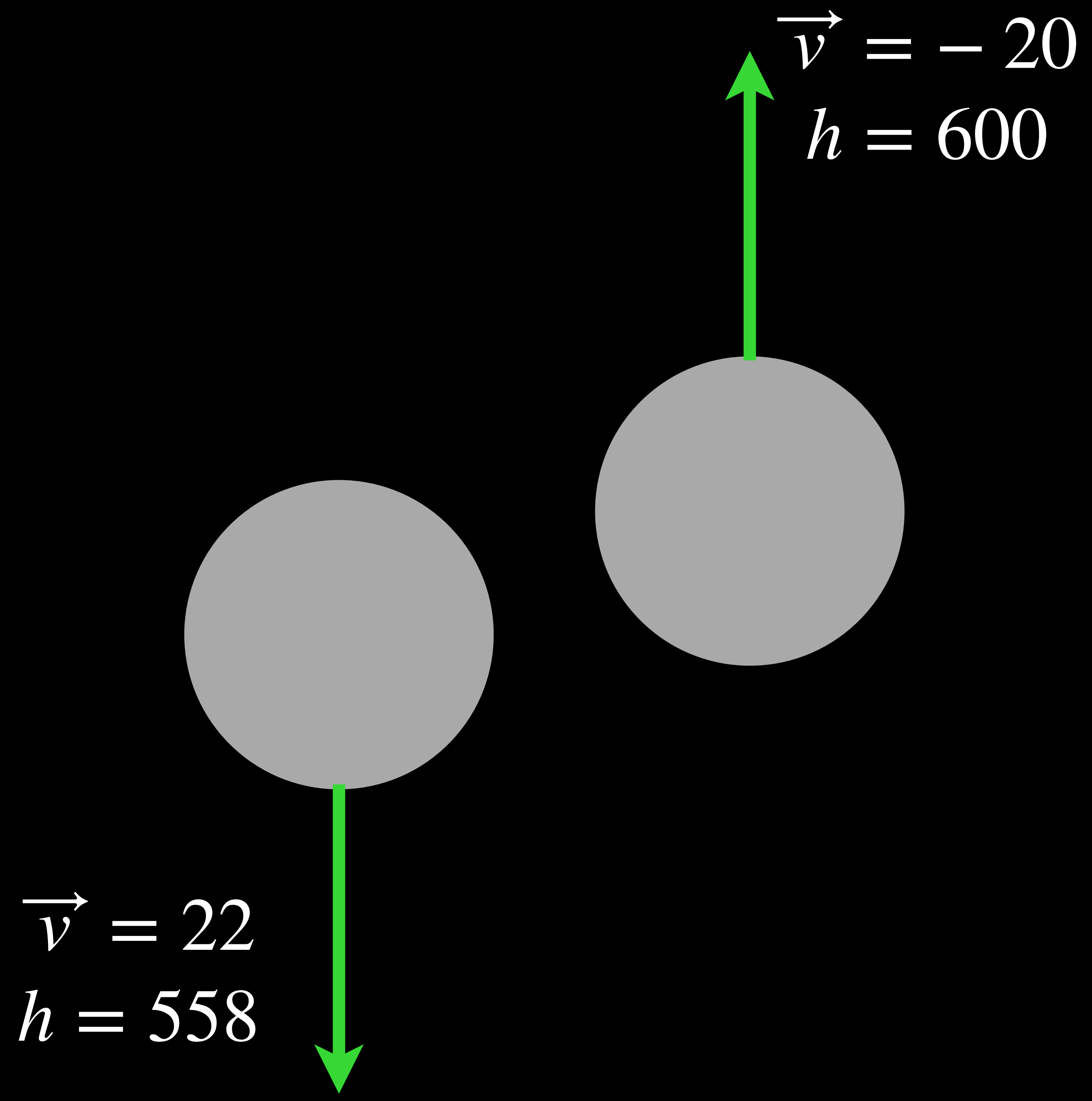
- For non-commutative group actions:
  Orbit equality : as hard as graph isomorphism problem
  Orbit closure intersection : conjectured to be in P. Geometric complexity theory program [Mulmuley, Sohoni '01]
  Orbit closure containment : NP-hard / as hard as tensor rank [Bläser, Ikenmeyer, Lysikov, Pandey, Schreyer '21]

- Orbit closure intersection problem admits polynomial time algorithm for:
  Left-right action, [Derksen,Makam '18], [Allen-Zhu, Garg, Li, Oliveira, Wigderson '18], [Ivanyos, Qiao, Subrahmanyam '18]
  simultaneous conjugation, [Derksen, Makam '18]
  quiver representations,
  actions of groups of bounded dimension [Mulmuley '12]
  …



$\vec{v} = -20$
$h = 600$
$\vec{v} = 22$
$h = 558$

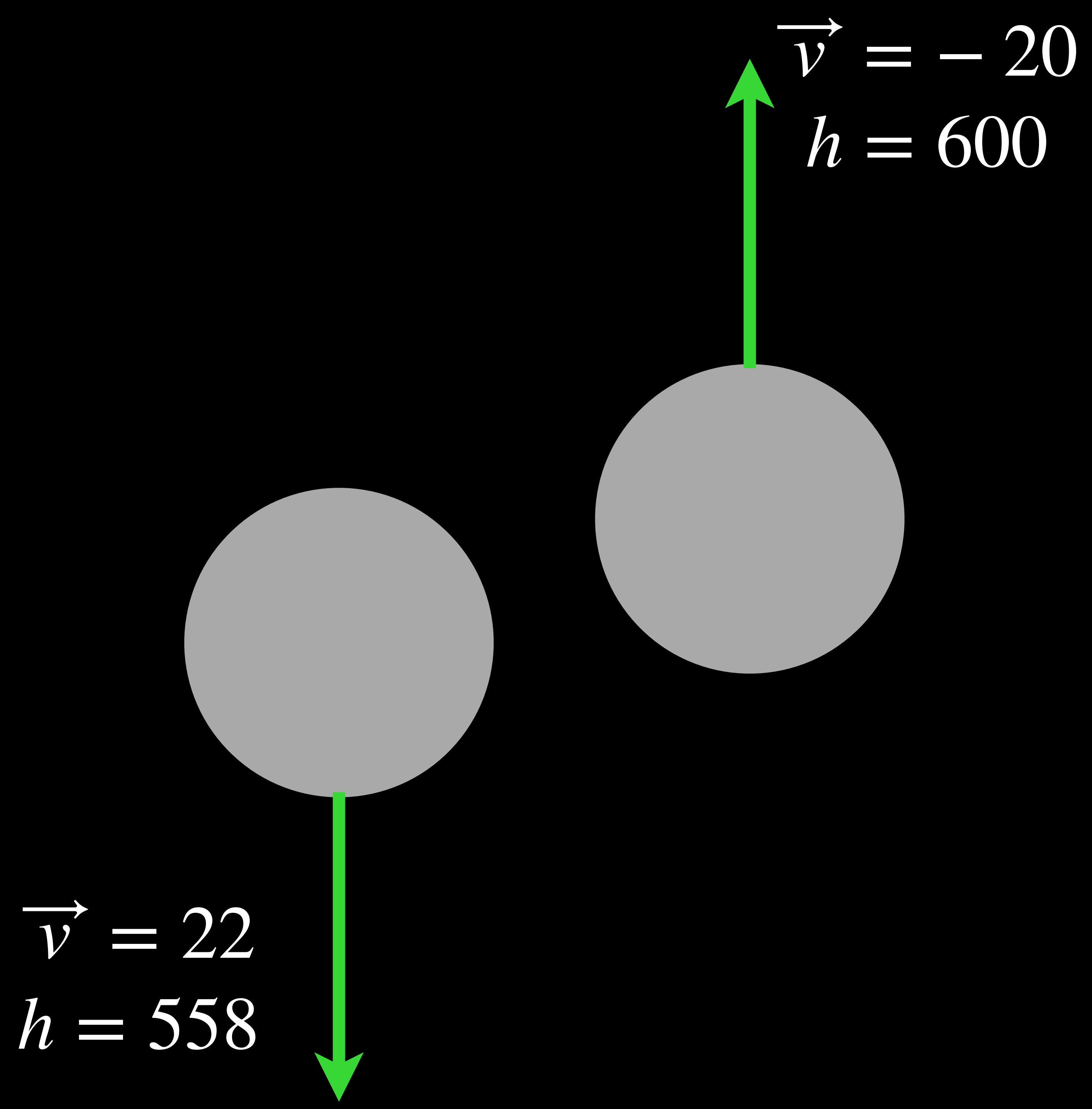
- How to tell if two shapshots belong to the same ball?



- How to tell if two shapshots belong to the same ball?

- The total energy of the ball is an invariant:

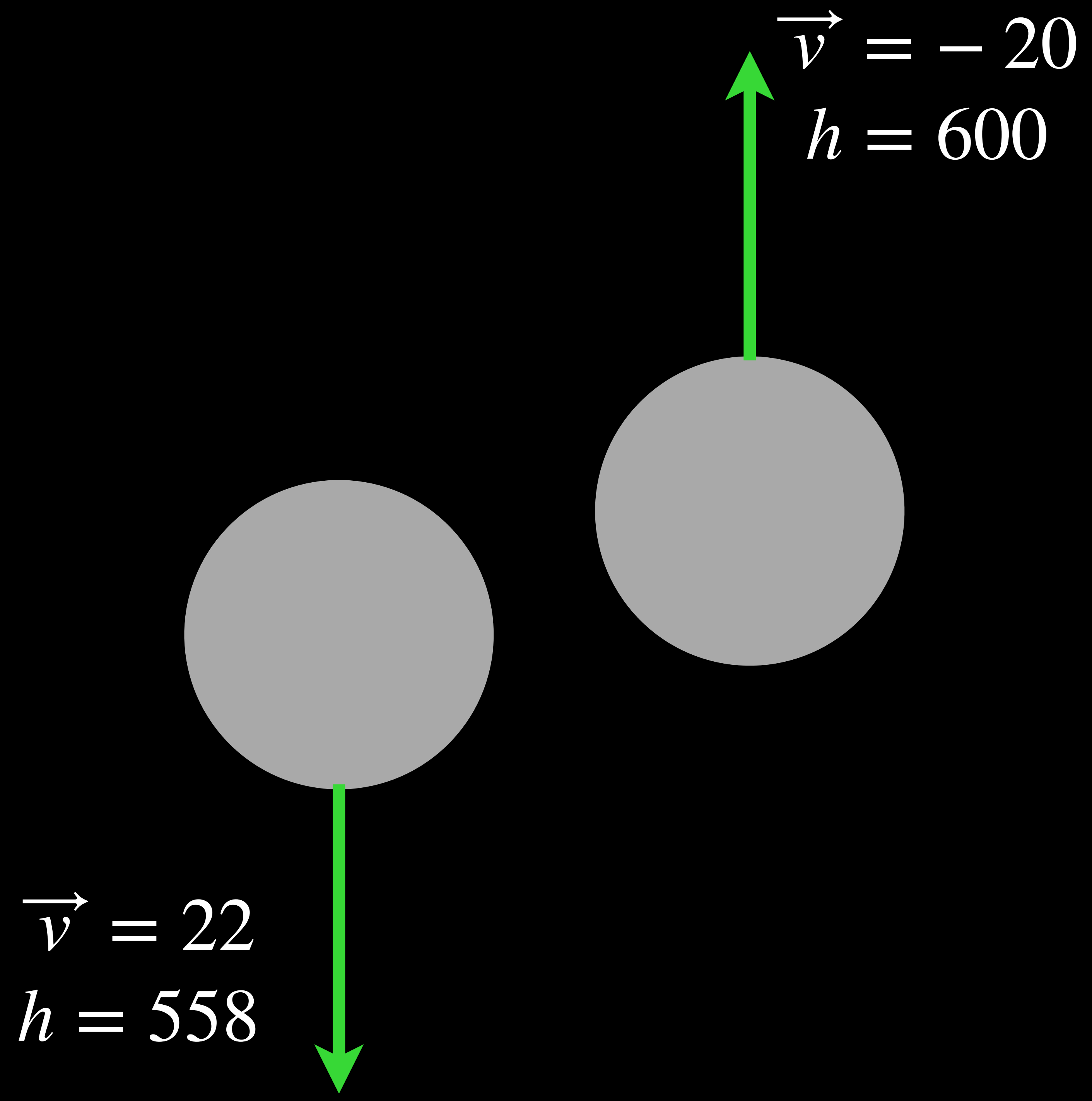
$$E = \frac{1}{2}mv^2 + mgh$$



- How to tell if two shapshots belong to the same ball?
- The total energy of the ball is an invariant:

$$E = \frac{1}{2}mv^2 + mgh$$

- An *invariant* is a polynomial $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$ such that
  $\forall t \in T, v \in V, \qquad f(t \cdot v) = f(v)$



- How to tell if two shapshots belong to the same ball?

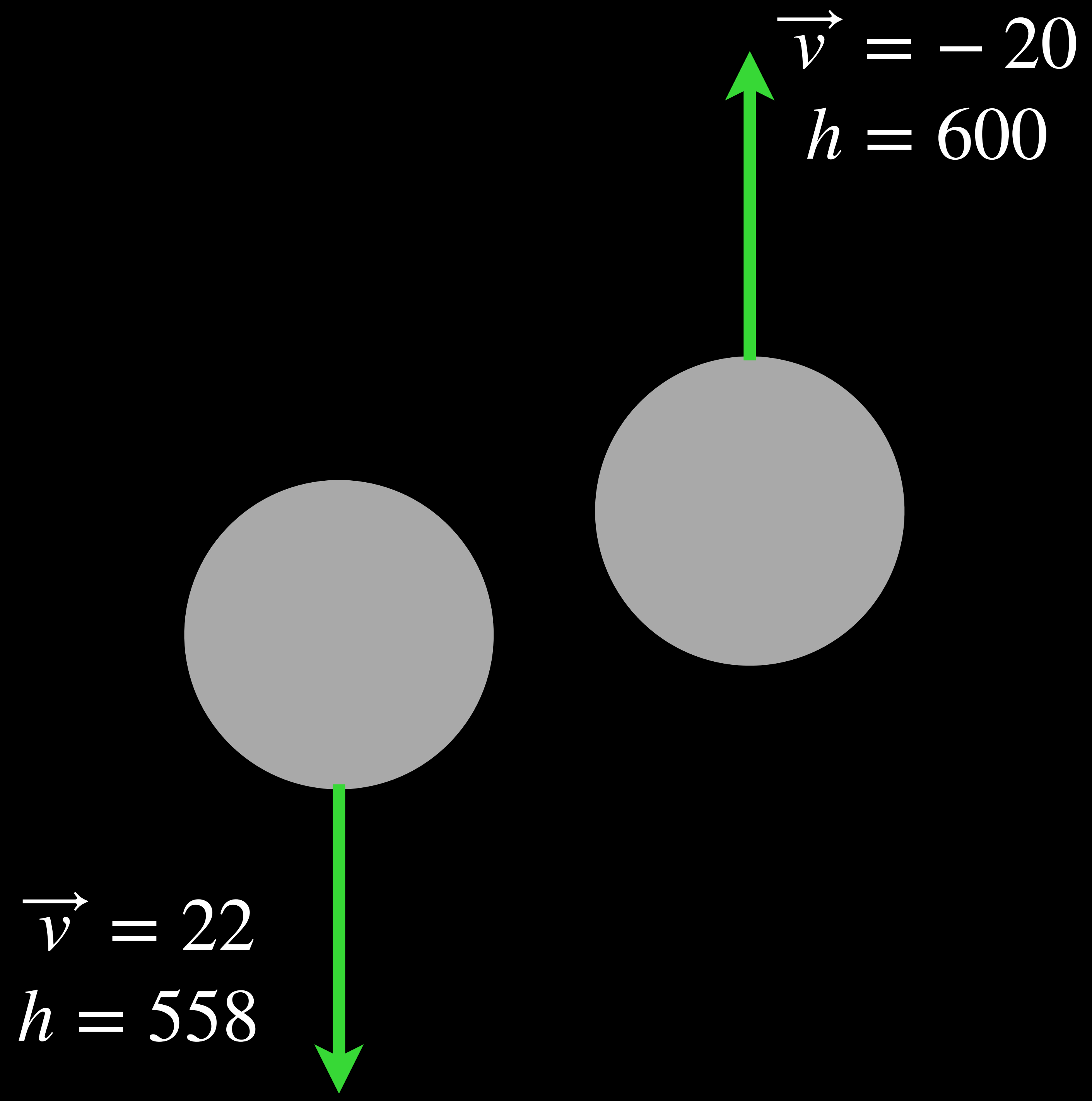
- The total energy of the ball is an invariant:

$$E = \frac{1}{2}mv^2 + mgh$$

- An *invariant* is a polynomial $f \in \mathbb{C}[x_1, x_2, \dots, x_n]$ such that

$$\forall t \in T, v \in V, \qquad f(t \cdot v) = f(v)$$

- $\mathbb{C}[x_1, x_2, \dots, x_n]^T = \{f \mid f$ is an invariant$\}$ is a finitely generated subalgebra of $\mathbb{C}[x_1, \dots, x_n]$, called the ring of invariants.



- How to tell if two shapshots belong to the same ball?

- The total energy of the ball is an invariant:

$$E = \frac{1}{2}mv^2 + mgh$$

- An *invariant* is a polynomial $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$ such that $\forall t \in T, v \in V, \qquad f(t \cdot v) = f(v)$

- $\mathbb{C}[x_1, x_2, \ldots, x_n]^T = \{f \mid f \text{ is an invariant}\}$ is a finitely generated subalgebra of $\mathbb{C}[x_1, \ldots, x_n]$, called the ring of invariants.

- **Mumford's Theorem:** $\overline{Tv} \cap \overline{Tw} \neq \emptyset \iff \forall f \in \mathbb{C}[x_1, x_2, \ldots, x_n]^T, \quad f(v) = f(w)\,.$

- How to tell if two shapshots belong to the same ball?

- The total energy of the ball is an invariant:

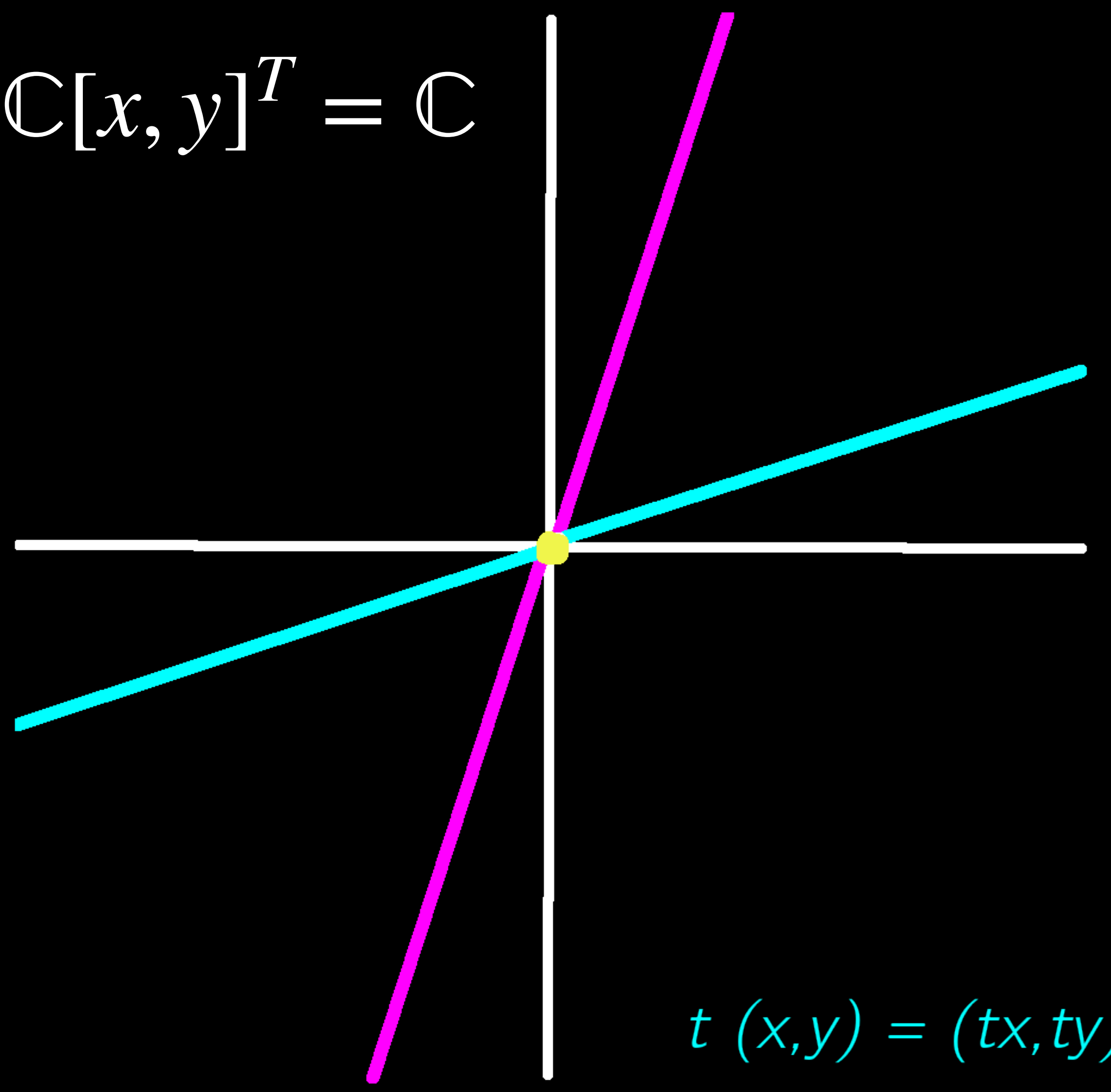
$$E = \frac{1}{2}mv^2 + mgh$$

- An *invariant* is a polynomial $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$ such that
$\forall t \in T, v \in V, \qquad f(t \cdot v) = f(v)$

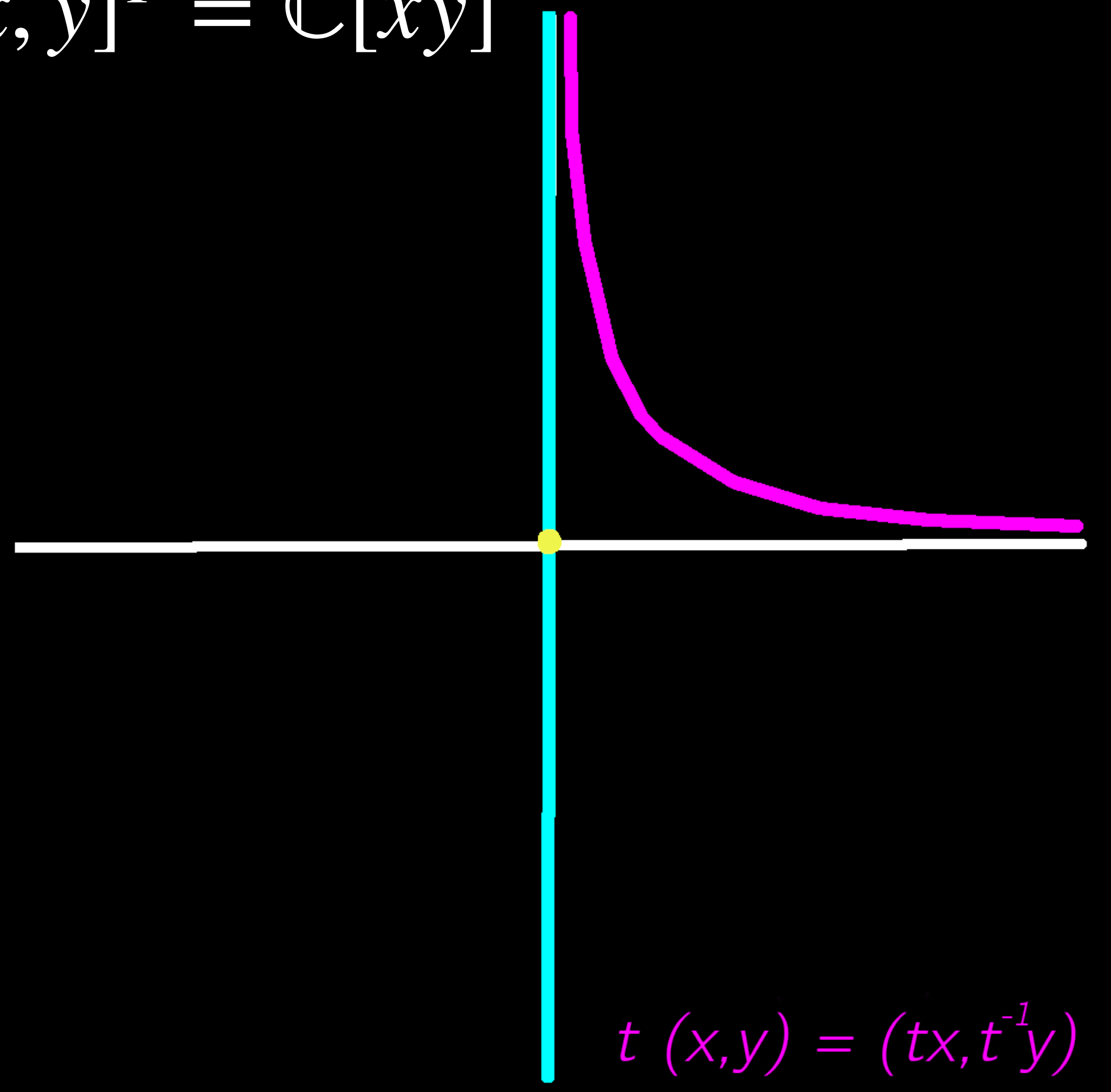
- $\mathbb{C}[x_1, x_2, \ldots, x_n]^T = \{f \mid f \text{ is an invariant}\}$ is a finitely generated subalgebra of $\mathbb{C}[x_1, \ldots, x_n]$, called the ring of invariants.

- **Mumford's Theorem:**
$\overline{Tv} \cap \overline{Tw} \neq \emptyset \quad \Longleftrightarrow \quad \forall f \in \mathbb{C}[x_1, x_2, \ldots, x_n]^T, \quad f(v) = f(w)\,.$

$\mathbb{C}[x, y]^T = \mathbb{C}$



$\mathbb{C}[x, y]^T = \mathbb{C}[xy]$



- How to tell if two shapshots belong to the same ball?

- The total energy of the ball is an invariant:

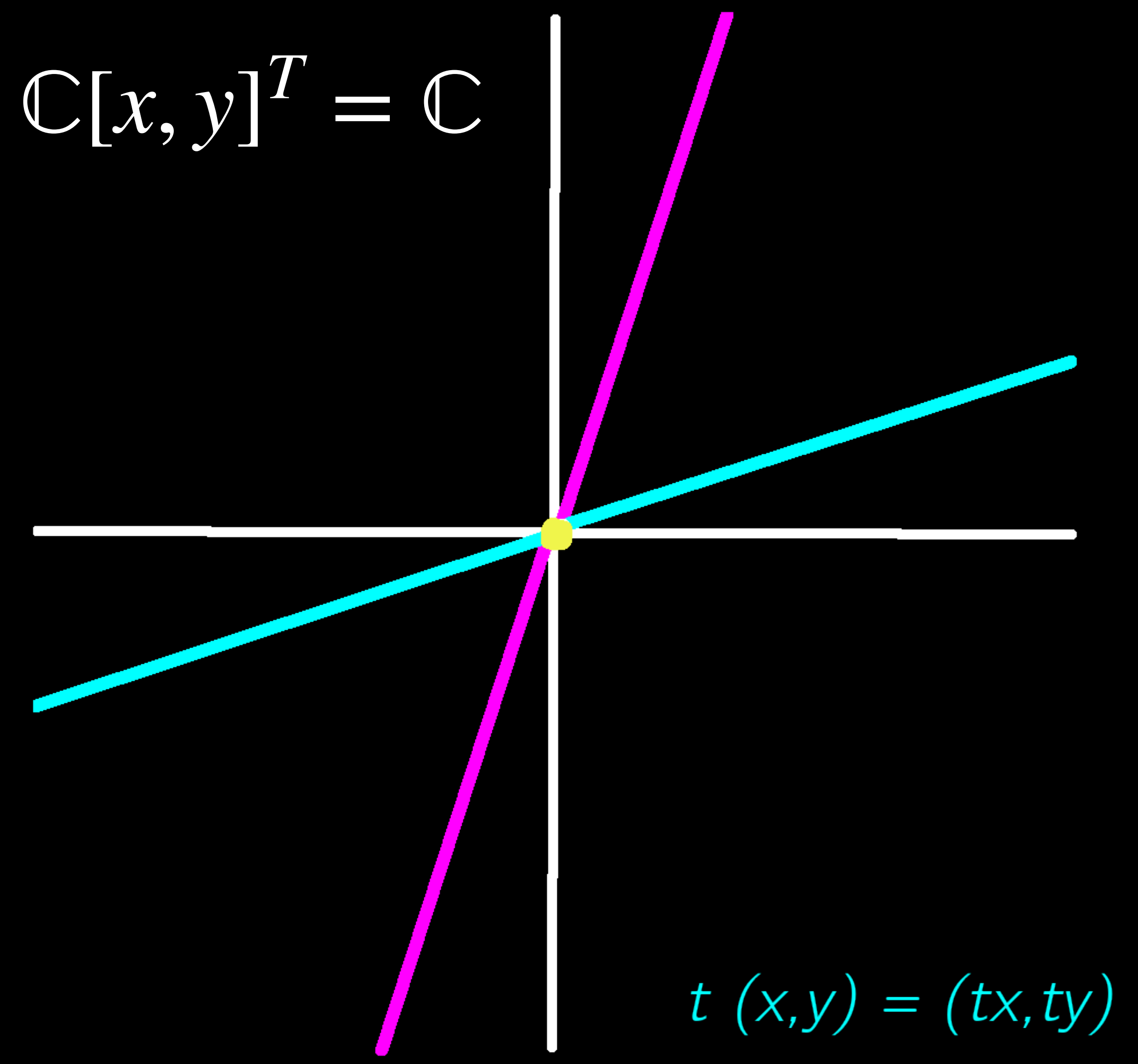
$$E = \frac{1}{2}mv^2 + mgh$$

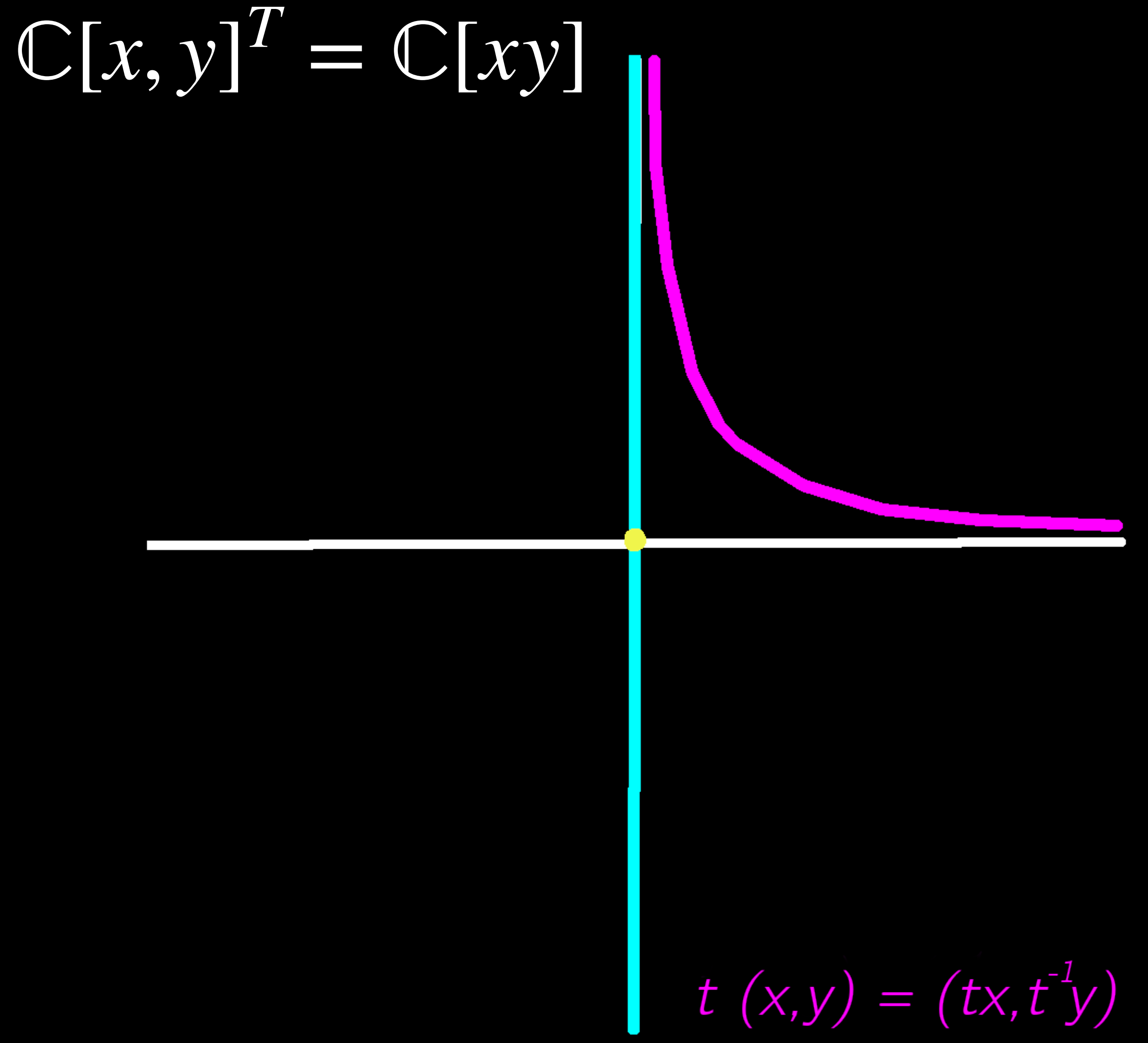
- An *invariant* is a polynomial $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$ such that
$\forall t \in T, v \in V, \qquad f(t \cdot v) = f(v)$

- $\mathbb{C}[x_1, x_2, \ldots, x_n]^T = \{f \mid f \text{ is an invariant}\}$ is a finitely generated subalgebra of $\mathbb{C}[x_1, \ldots, x_n]$, called the ring of invariants.

- Mumford's Theorem:
$\overline{Tv} \cap \overline{Tw} \neq \emptyset \iff \forall f \in \mathbb{C}[x_1, x_2, \ldots, x_n]^T, \quad f(v) = f(w)$.

- If $\mathbb{C}[x_1, x_2, \ldots, x_n]^T$ is generated by the invariants $f_1, f_2, \ldots, f_k$ then
$\overline{Tv} \cap \overline{Tw} \neq \emptyset \iff \forall i \in [k], \quad f_i(v) = f_i(w)$.





Invariants of Scaling Actions

Invariants of Scaling Actions

- Polynomial invariant: $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$,
  $f(tv) = f(v)$ for every $t \in T, v \in V$.

Invariants of Scaling Actions

- Polynomial invariant: $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$,
  $f(tv) = f(v)$ for every $t \in T, v \in V$.

- The invariants are spanned by invariant monomials:
  $\{x^\alpha \mid \alpha \in \mathbb{N}^n, M\alpha = 0\}$

Invariants of Scaling Actions

- Polynomial invariant: $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$, $f(tv) = f(v)$ for every $t \in T, v \in V$.

- The invariants are spanned by invariant monomials: $\{x^\alpha \mid \alpha \in \mathbb{N}^n, M\alpha = 0\}$

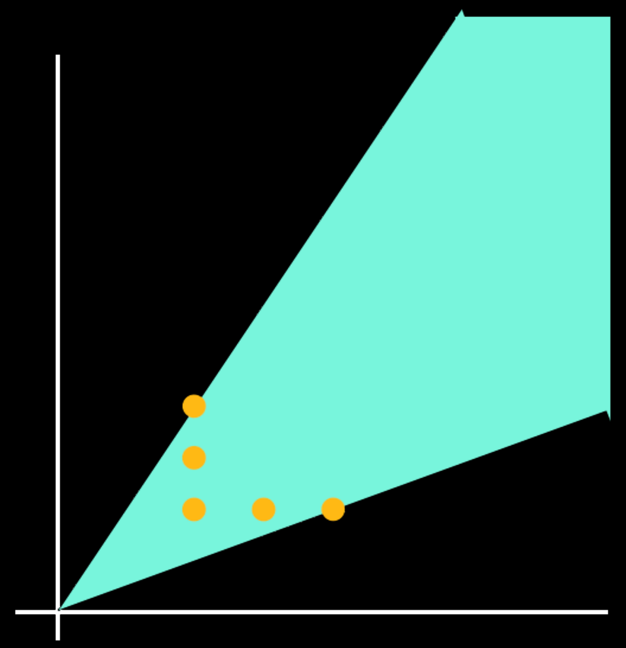
- Exponent vectors form a *semigroup*: $S = \{\alpha \in \mathbb{N}^n \mid M\alpha = 0\}$, which admits a *Hilbert basis*, $\mathscr{H}$.

Invariants of Scaling Actions

- Polynomial invariant: $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$, $f(tv) = f(v)$ for every $t \in T, v \in V$.
- The invariants are spanned by invariant monomials: $\{x^\alpha \mid \alpha \in \mathbb{N}^n, M\alpha = 0\}$
- Exponent vectors form a *semigroup*: $S = \{\alpha \in \mathbb{N}^n \mid M\alpha = 0\}$, which admits a *Hilbert basis*, $\mathscr{H}$.
- $\mathbb{C}[x_1, \ldots, x_n]^T$ is generated by $\{x^\alpha \mid \alpha \in \mathscr{H}\}$.

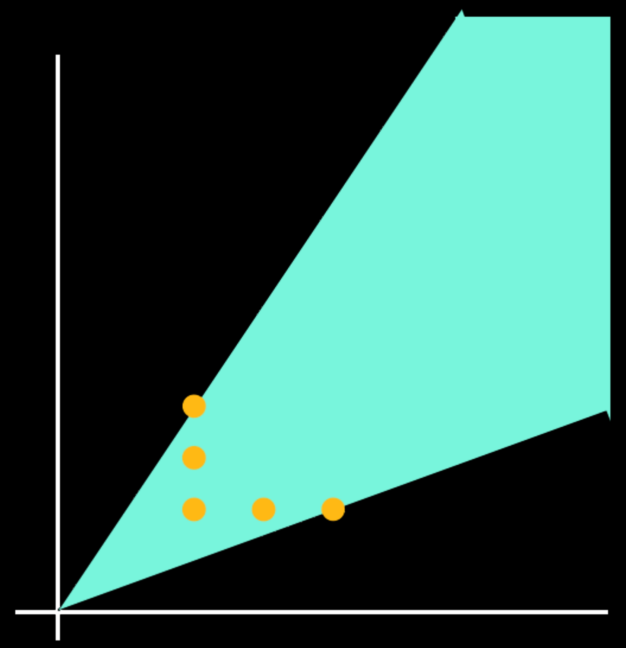
Invariants of Scaling Actions

- Polynomial invariant: $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$, $f(tv) = f(v)$ for every $t \in T, v \in V$.
- The invariants are spanned by invariant monomials: $\{x^\alpha \mid \alpha \in \mathbb{N}^n, M\alpha = 0\}$
- Exponent vectors form a *semigroup*: $S = \{\alpha \in \mathbb{N}^n \mid M\alpha = 0\}$, which admits a *Hilbert basis*, $\mathscr{H}$.
- $\mathbb{C}[x_1, \ldots, x_n]^T$ is generated by $\{x^\alpha \mid \alpha \in \mathscr{H}\}$.
- $\mathscr{H}$ might have exponentially large cardinality!



Invariants of Scaling Actions

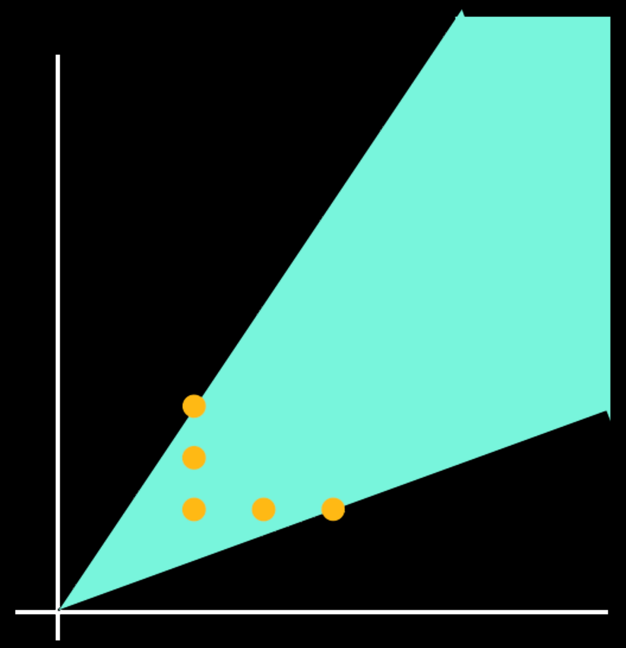
- Polynomial invariant: $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$, $f(tv) = f(v)$ for every $t \in T, v \in V$.
- The invariants are spanned by invariant monomials: $\{x^\alpha \mid \alpha \in \mathbb{N}^n, M\alpha = 0\}$
- Exponent vectors form a *semigroup*: $S = \{\alpha \in \mathbb{N}^n \mid M\alpha = 0\}$, which admits a *Hilbert basis*, $\mathcal{H}$.
- $\mathbb{C}[x_1, \ldots, x_n]^T$ is generated by $\{x^\alpha \mid \alpha \in \mathcal{H}\}$.
- $\mathcal{H}$ might have exponentially large cardinality!



- Rational invariants: $\varphi = \dfrac{f}{g} \in \mathbb{C}(x_1, x_2, \ldots, x_n)$, $\varphi(tv) = \varphi(v)$ for every $t \in T, v \in V$.

Invariants of Scaling Actions

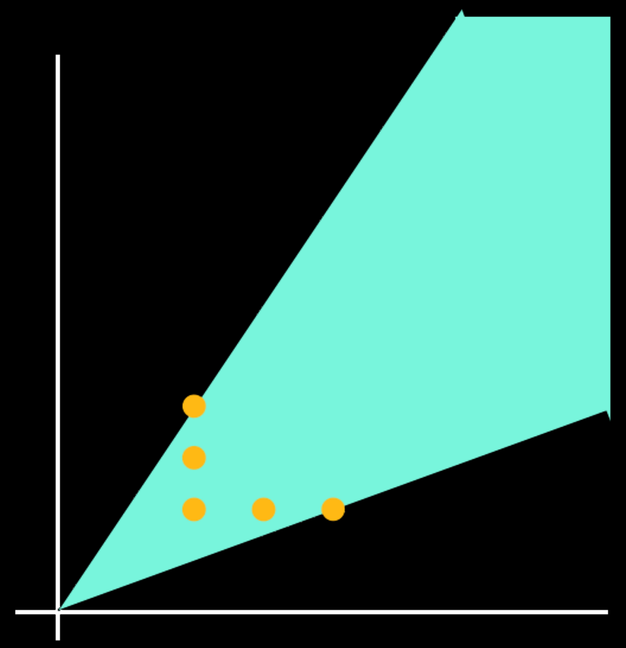
- Polynomial invariant: $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$, $f(tv) = f(v)$ for every $t \in T, v \in V$.
- The invariants are spanned by invariant monomials: $\{x^\alpha \mid \alpha \in \mathbb{N}^n, M\alpha = 0\}$
- Exponent vectors form a *semigroup*: $S = \{\alpha \in \mathbb{N}^n \mid M\alpha = 0\}$, which admits a *Hilbert* basis, $\mathscr{H}$.
- $\mathbb{C}[x_1, \ldots, x_n]^T$ is generated by $\{x^\alpha \mid \alpha \in \mathscr{H}\}$.
- $\mathscr{H}$ might have exponentially large cardinality!



- Rational invariants: $\varphi = \dfrac{f}{g} \in \mathbb{C}(x_1, x_2, \ldots, x_n)$, $\varphi(tv) = \varphi(v)$ for every $t \in T, v \in V$.
- Spanned by invariant *Laurent monomials:* $\{x^\alpha \mid \alpha \in \mathbb{Z}^n, M\alpha = 0\}$

Invariants of Scaling Actions

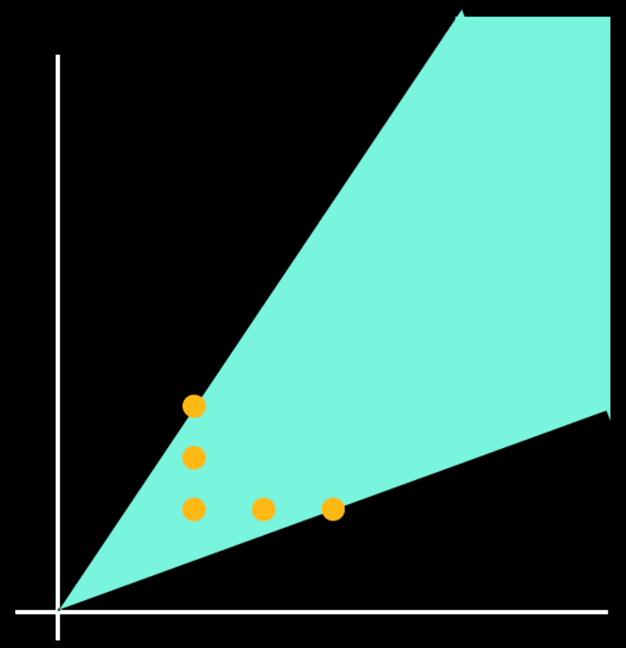
- Polynomial invariant: $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$, $f(tv) = f(v)$ for every $t \in T, v \in V$.
- The invariants are spanned by invariant monomials: $\{x^\alpha \mid \alpha \in \mathbb{N}^n, M\alpha = 0\}$
- Exponent vectors form a *semigroup*: $S = \{\alpha \in \mathbb{N}^n \mid M\alpha = 0\}$, which admits a *Hilbert basis*, $\mathscr{H}$.
- $\mathbb{C}[x_1, \ldots, x_n]^T$ is generated by $\{x^\alpha \mid \alpha \in \mathscr{H}\}$.
- $\mathscr{H}$ might have exponentially large cardinality!



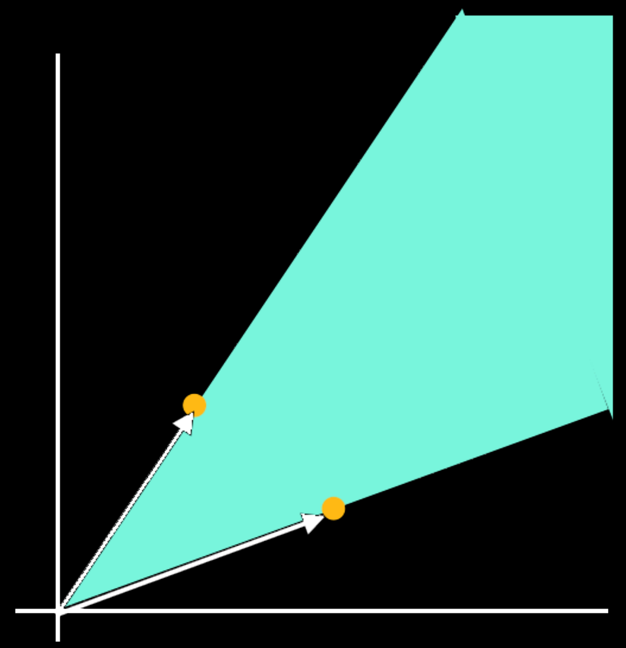
- Rational invariants: $\varphi = \dfrac{f}{g} \in \mathbb{C}(x_1, x_2, \ldots, x_n)$, $\varphi(tv) = \varphi(v)$ for every $t \in T, v \in V$.
- Spanned by invariant *Laurent monomials*: $\{x^\alpha \mid \alpha \in \mathbb{Z}^n, M\alpha = 0\}$
- Exponent vectors form a *lattice*: $L = \{\alpha \in \mathbb{Z}^n \mid M\alpha = 0\}$, which admits a *lattice basis*, $\mathscr{B}$.

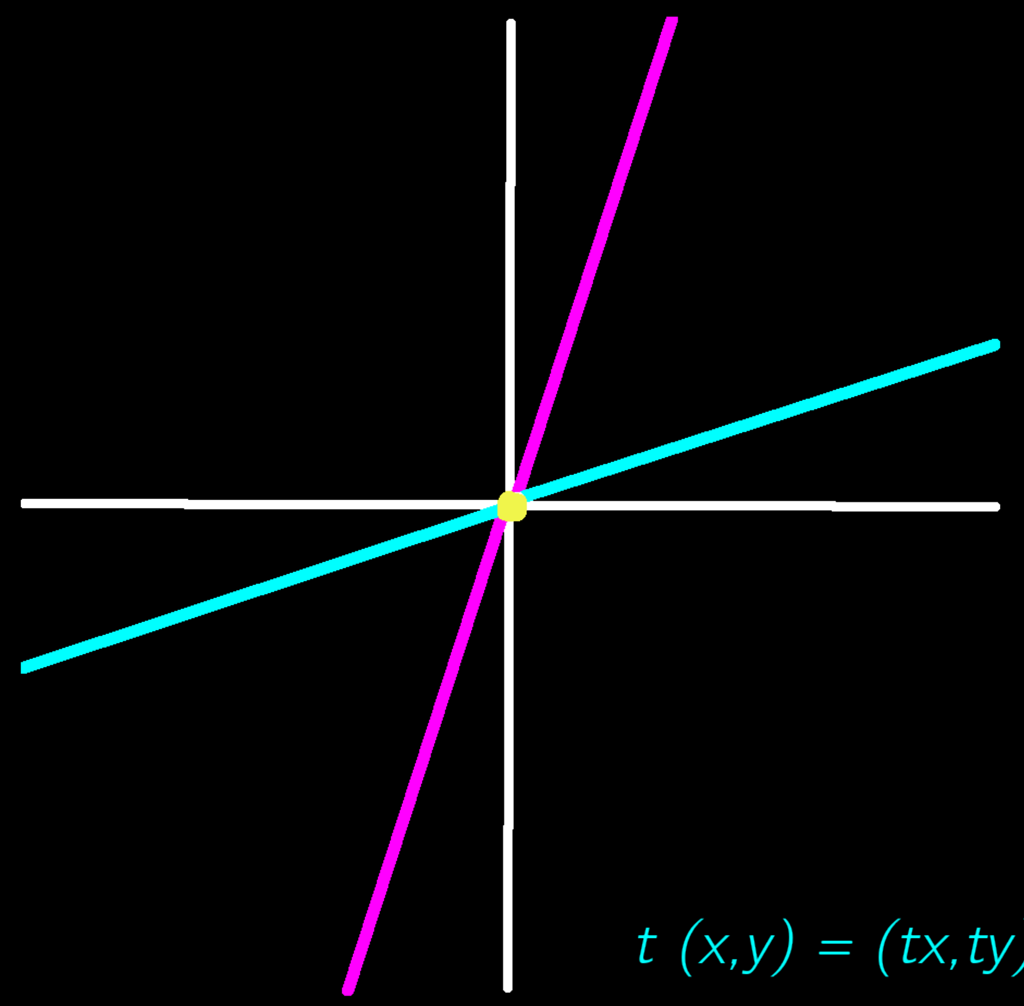
Invariants of Scaling Actions

- Polynomial invariant: $f \in \mathbb{C}[x_1, x_2, \ldots, x_n]$, $f(tv) = f(v)$ for every $t \in T, v \in V$.
- The invariants are spanned by invariant monomials: $\{x^\alpha \mid \alpha \in \mathbb{N}^n, M\alpha = 0\}$
- Exponent vectors form a *semigroup*: $S = \{\alpha \in \mathbb{N}^n \mid M\alpha = 0\}$, which admits a *Hilbert basis*, $\mathscr{H}$.
- $\mathbb{C}[x_1, \ldots, x_n]^T$ is generated by $\{x^\alpha \mid \alpha \in \mathscr{H}\}$.
- $\mathscr{H}$ might have exponentially large cardinality!

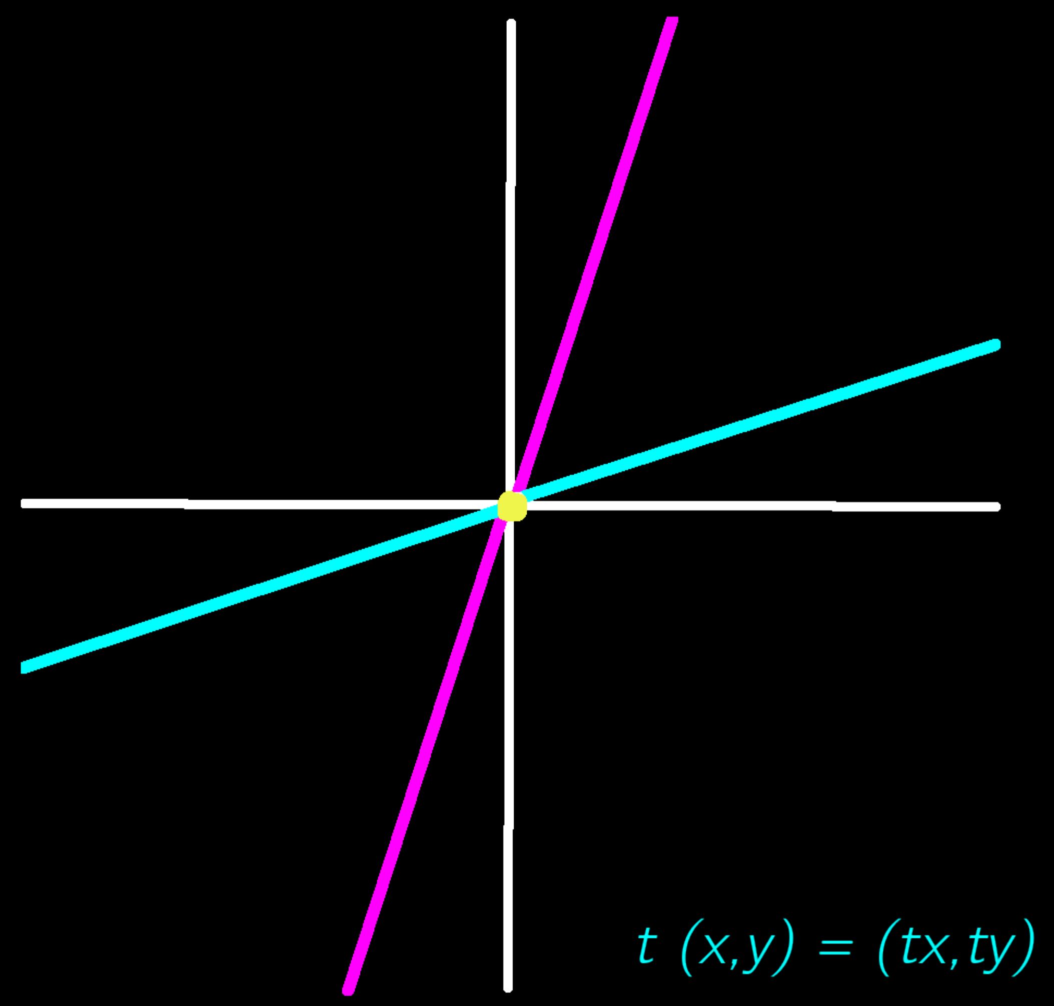


- Rational invariants: $\varphi = \frac{f}{g} \in \mathbb{C}(x_1, x_2, \ldots, x_n)$, $\varphi(tv) = \varphi(v)$ for every $t \in T, v \in V$.
- Spanned by invariant *Laurent monomials*: $\{x^\alpha \mid \alpha \in \mathbb{Z}^n, M\alpha = 0\}$
- Exponent vectors form a *lattice*: $L = \{\alpha \in \mathbb{Z}^n \mid M\alpha = 0\}$, which admits a *lattice basis*, $\mathscr{B}$.
- $\mathscr{B}$ has cardinality at most n.

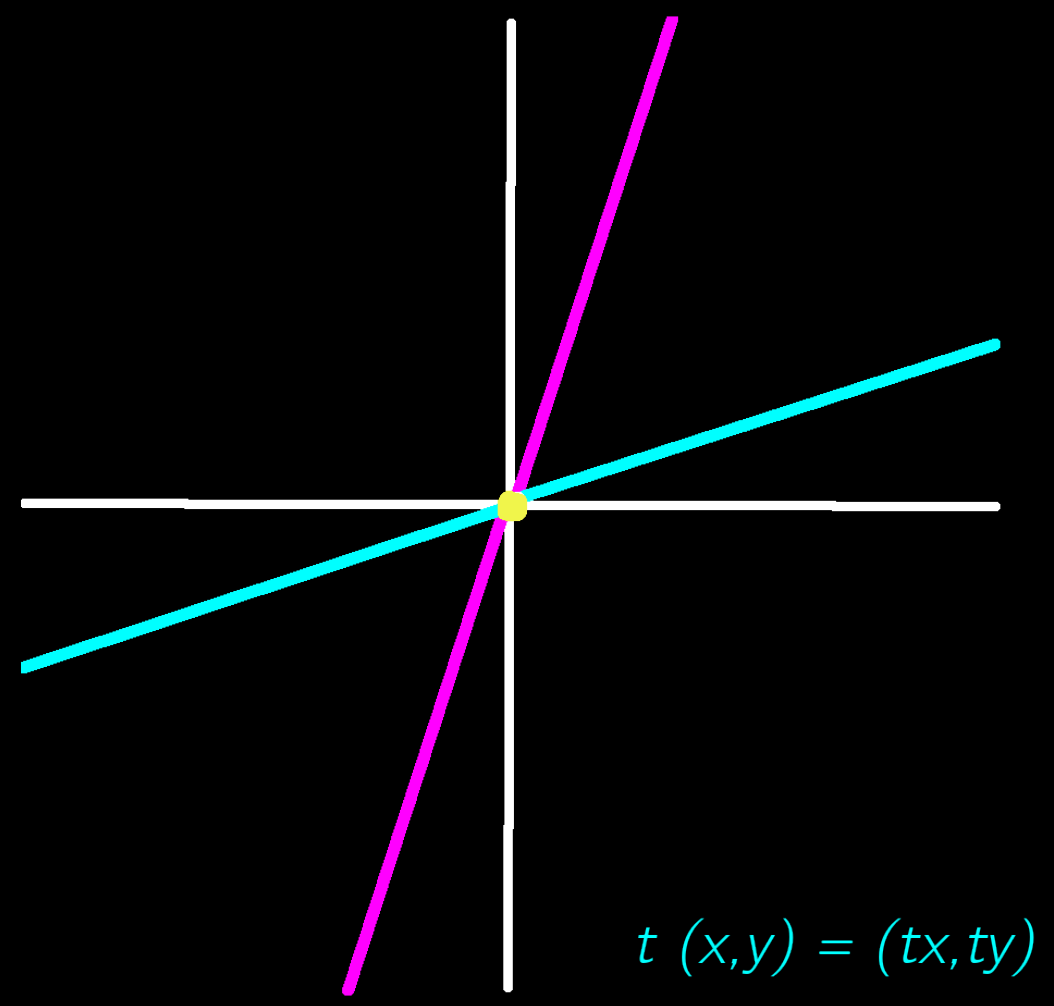




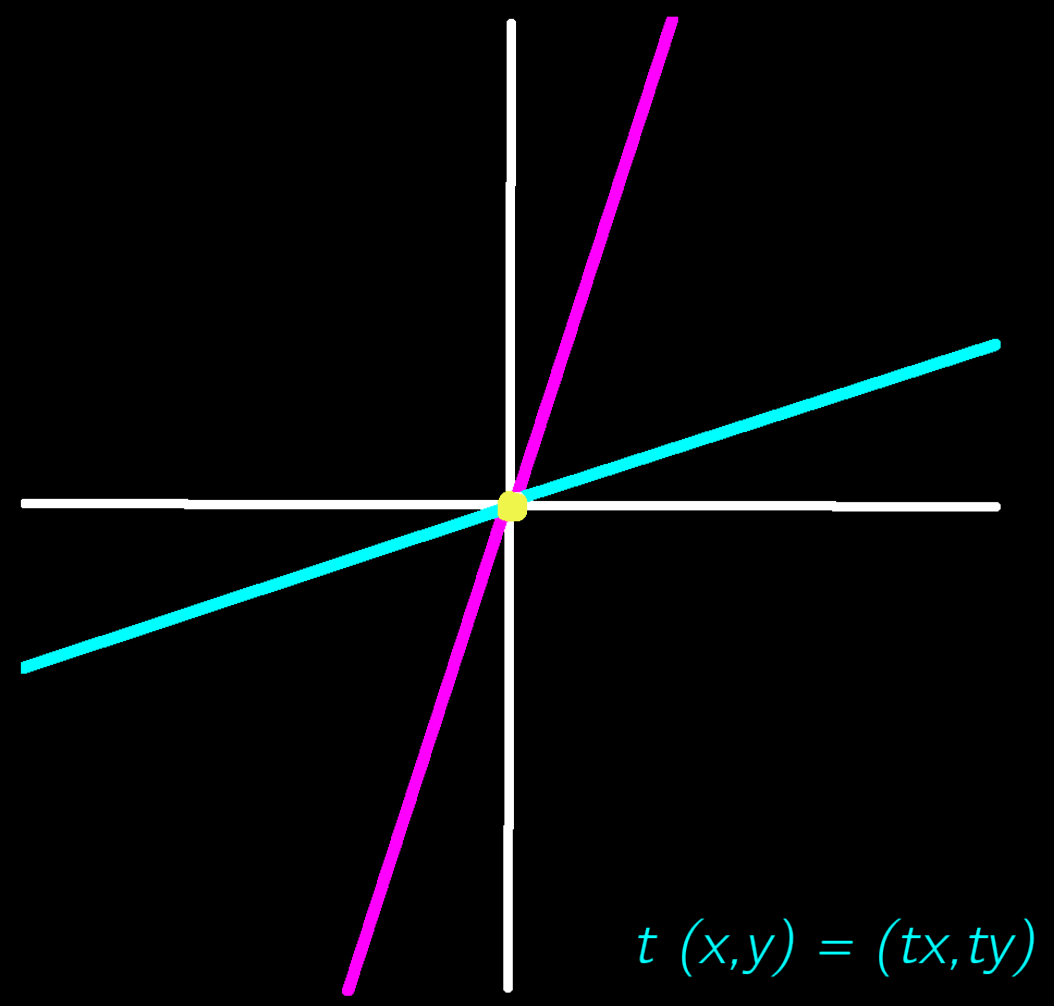
$t\ (x,y) = (tx,ty)$



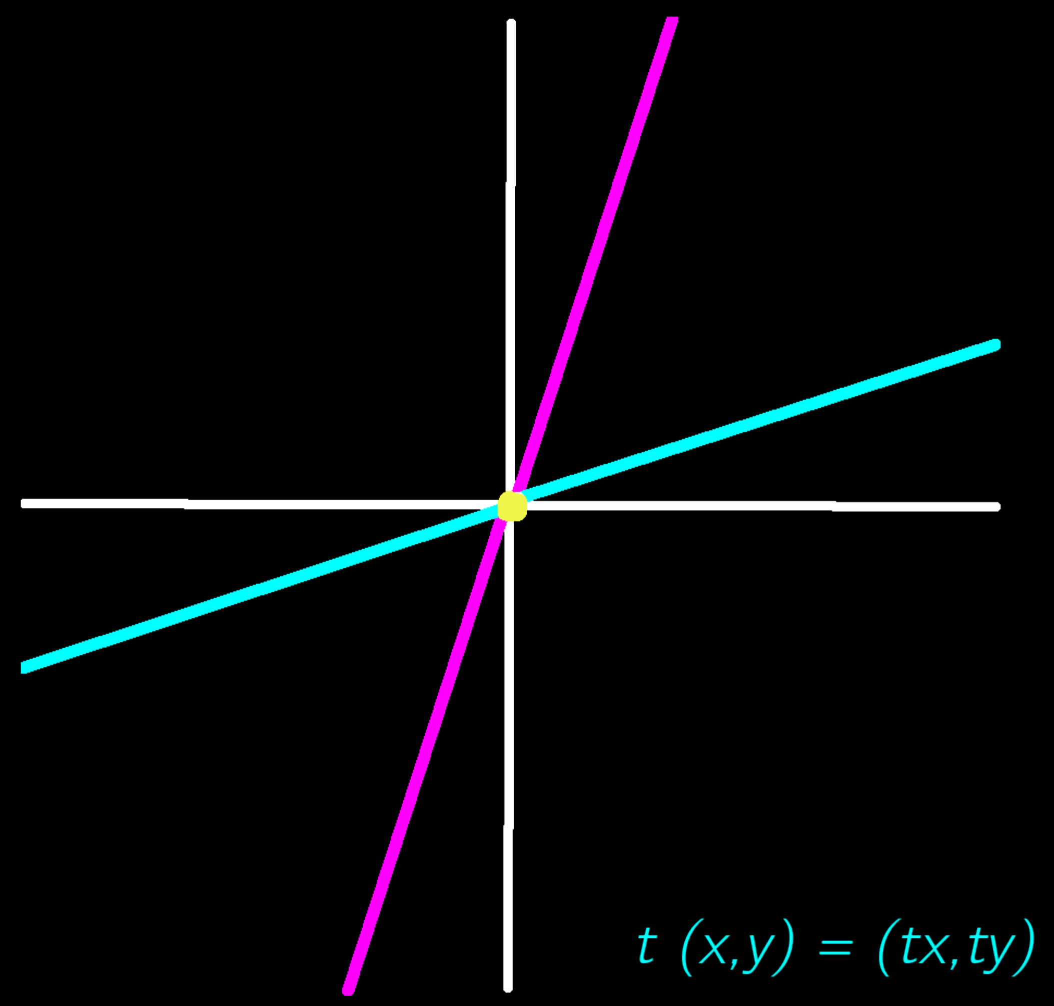
- **The ring of invariants is trivial :** $\mathbb{C}[x,y]^T = \mathbb{C}$
  Orbit closures intersect at 0.



- The ring of invariants is trivial : $\mathbb{C}[x,y]^T = \mathbb{C}$
  Orbit closures intersect at 0.
- The field of invariants contains y/x : $\mathbb{C}(x,y)^T = \mathbb{C}(y/x)$.
  y/x separates the lines with finite slope. It does not separate the y-axis and the origin.



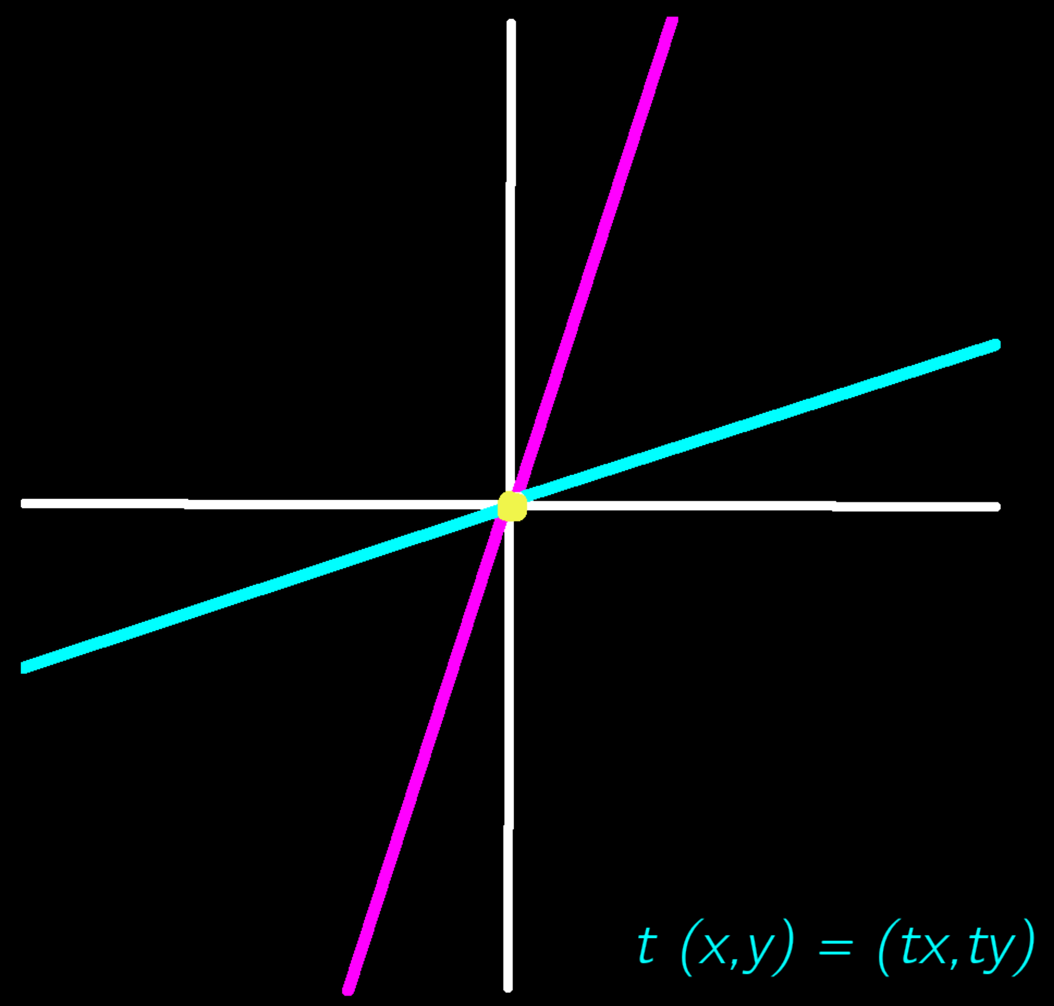
- The ring of invariants is trivial : $\mathbb{C}[x,y]^T = \mathbb{C}$
  Orbit closures intersect at 0.
- The field of invariants contains y/x : $\mathbb{C}(x,y)^T = \mathbb{C}(y/x)$.
  y/x separates the lines with finite slope. It does not separate the y-axis and the origin.
- **Fact:** Rational invariants separate orbits of vectors with full-coordinates.



- The ring of invariants is trivial : $\mathbb{C}[x, y]^T = \mathbb{C}$
  Orbit closures intersect at 0.
- The field of invariants contains y/x : $\mathbb{C}(x, y)^T = \mathbb{C}(y/x)$.
  y/x separates the lines with finite slope. It does not separate the y-axis and the origin.
- **Fact:** Rational invariants separate orbits of vectors with full-coordinates.

Algorithm: Orbit Equality

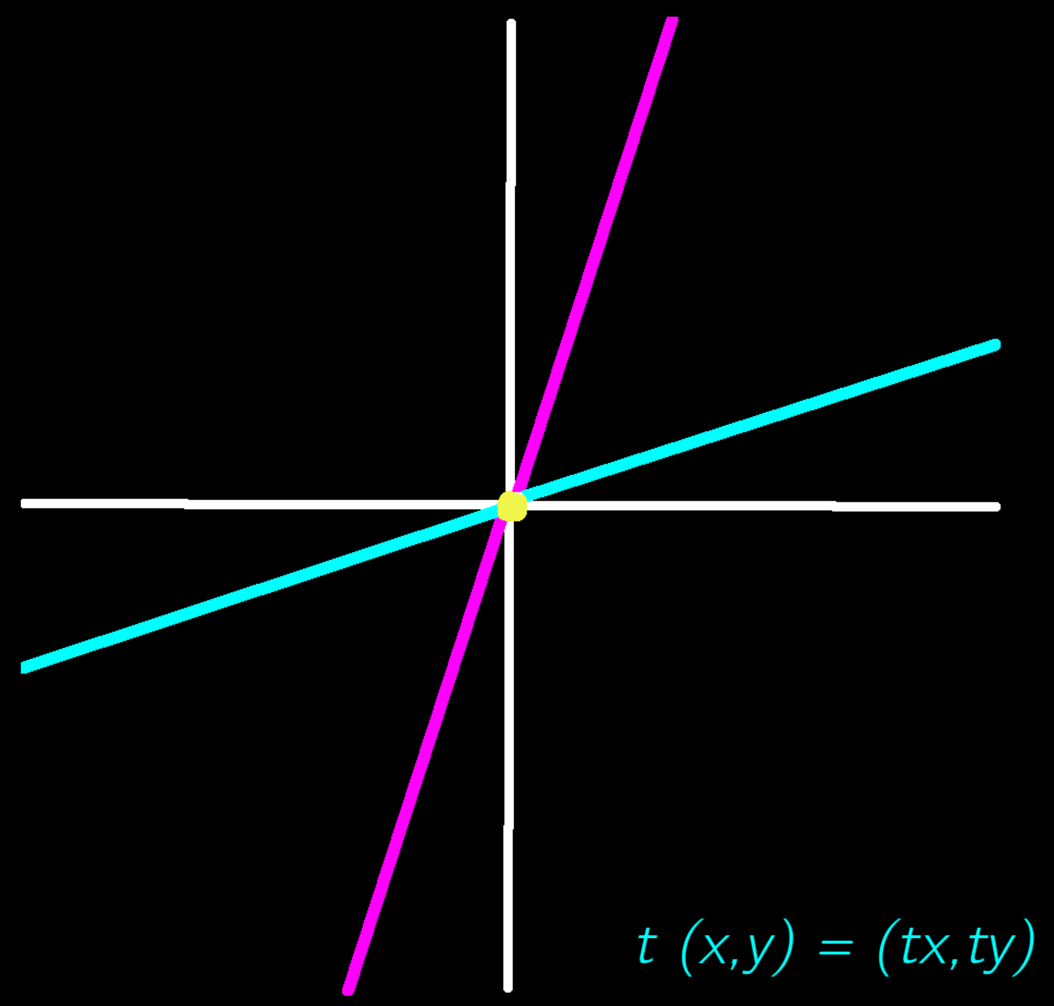
- Input: $v, w \in \mathbb{Q}(i)^n, M \in \mathbb{Z}^{d \times n}$.



- The ring of invariants is trivial : $\mathbb{C}[x, y]^T = \mathbb{C}$
  Orbit closures intersect at 0.
- The field of invariants contains y/x : $\mathbb{C}(x, y)^T = \mathbb{C}(y/x)$.
  y/x separates the lines with finite slope. It does not separate the y-axis and the origin.
- **Fact:** Rational invariants separate orbits of vectors with full-coordinates.

Algorithm: Orbit Equality

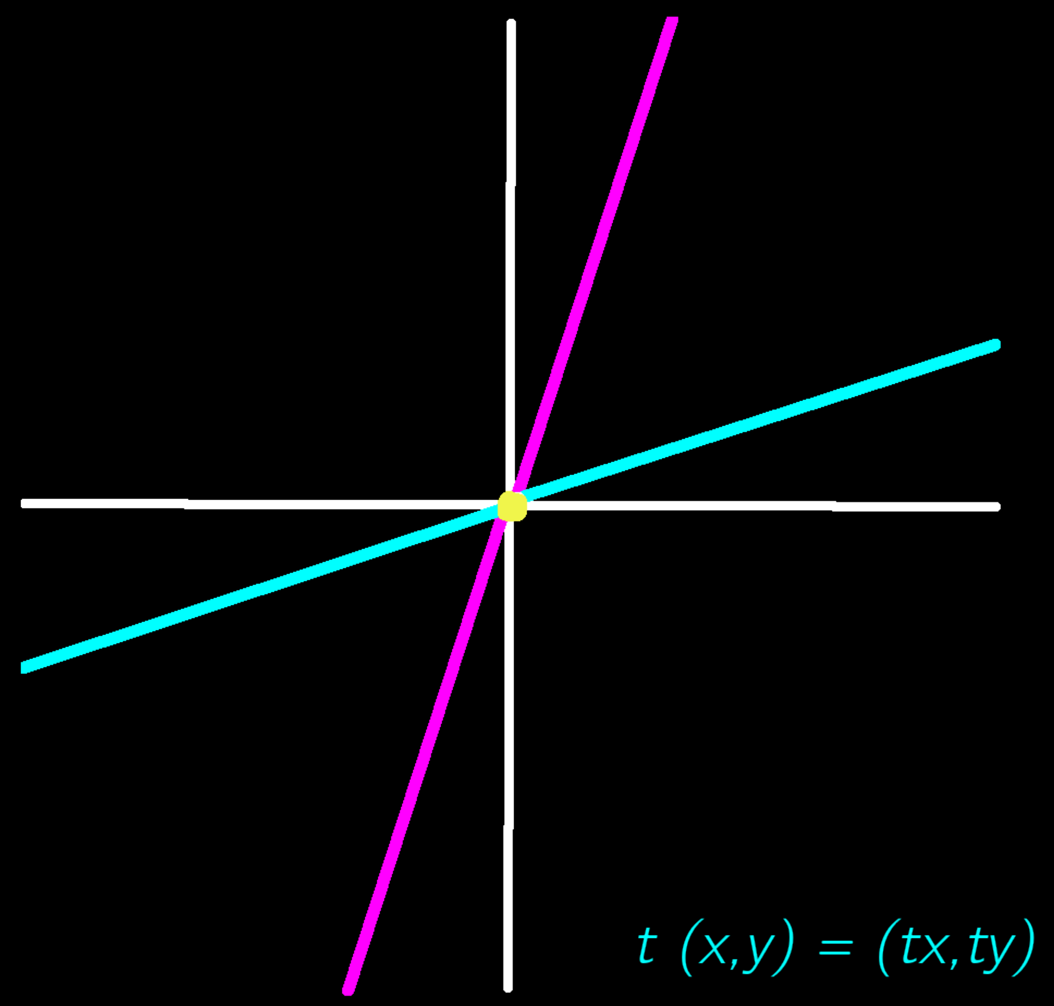
- Input: $v, w \in \mathbb{Q}(i)^n, M \in \mathbb{Z}^{d \times n}$.
- Test if $\{i \mid v_i = 0\} = \{i \mid w_i = 0\}$. If not equal, $Tv \neq Tw$.



- The ring of invariants is trivial : $\mathbb{C}[x,y]^T = \mathbb{C}$
  Orbit closures intersect at 0.
- The field of invariants contains y/x : $\mathbb{C}(x,y)^T = \mathbb{C}(y/x)$.
  y/x separates the lines with finite slope. It does not separate the y-axis and the origin.
- **Fact:** Rational invariants separate orbits of vectors with full-coordinates.

Algorithm: Orbit Equality

- Input: $v, w \in \mathbb{Q}(i)^n, M \in \mathbb{Z}^{d\times n}$.
- Test if $\{i \mid v_i = 0\} = \{i \mid w_i = 0\}$. If not equal, $Tv \neq Tw$.
- Delete the columns ω_i of M with $v_i = 0$.

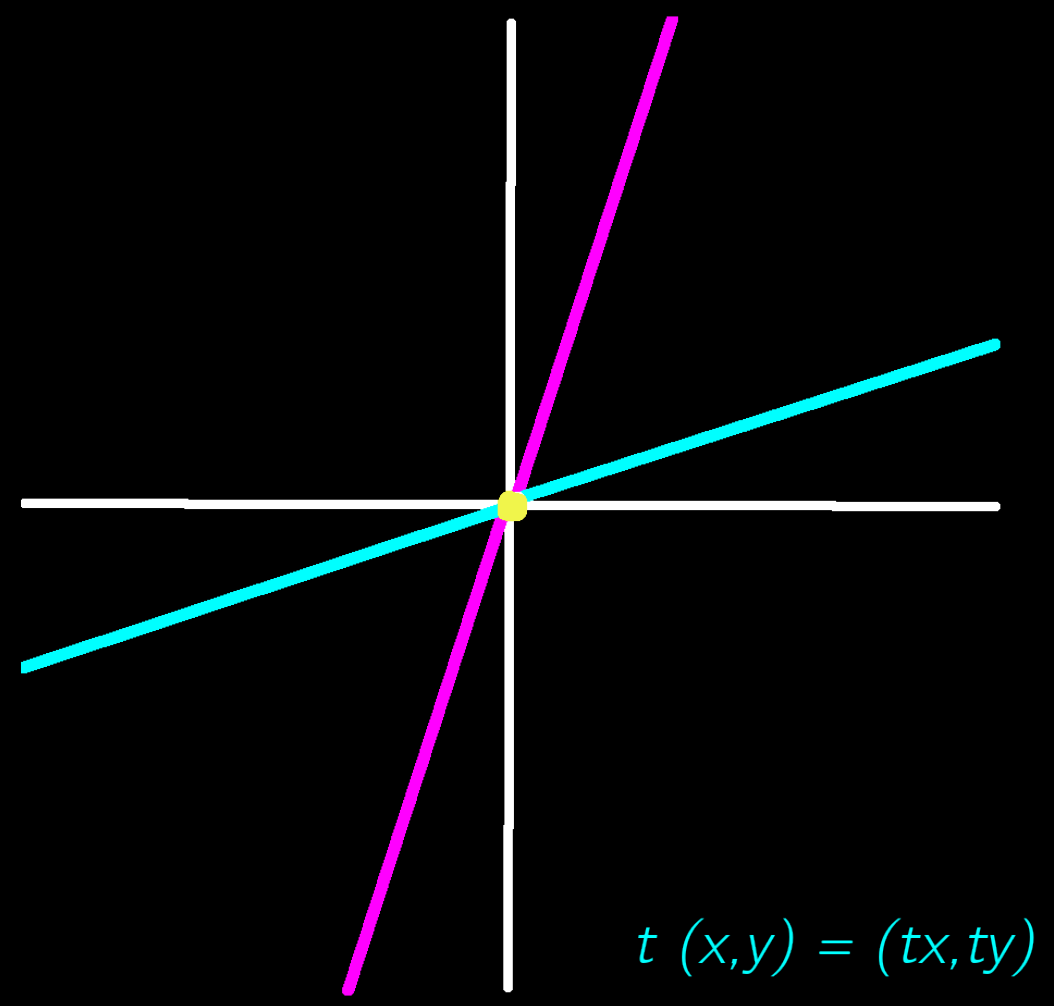


- The ring of invariants is trivial : $\mathbb{C}[x,y]^T = \mathbb{C}$
  Orbit closures intersect at 0.
- The field of invariants contains y/x : $\mathbb{C}(x,y)^T = \mathbb{C}(y/x)$.
  y/x separates the lines with finite slope. It does not separate the y-axis and the origin.
- **Fact:** Rational invariants separate orbits of vectors with full-coordinates.

Algorithm: Orbit Equality

- Input: $v, w \in \mathbb{Q}(i)^n, M \in \mathbb{Z}^{d \times n}$.
- Test if $\{i \mid v_i = 0\} \;=\; \{i \mid w_i = 0\}$. If not equal, $Tv \neq Tw$.
- Delete the columns ω_i of M with $v_i = 0$.
- Compute a lattice basis for

$$L = \{\alpha \in \mathbb{Z}^n \mid M\alpha = 0\}.$$



- The ring of invariants is trivial : $\mathbb{C}[x, y]^T = \mathbb{C}$
  Orbit closures intersect at 0.
- The field of invariants contains y/x : $\mathbb{C}(x, y)^T = \mathbb{C}(y/x)$.
  y/x separates the lines with finite slope. It does not separate the y-axis and the origin.
- **Fact:** Rational invariants separate orbits of vectors with full-coordinates.

Algorithm: Orbit Equality

- Input: $v, w \in \mathbb{Q}(i)^n, M \in \mathbb{Z}^{d\times n}$.
- Test if $\{i \mid v_i = 0\} = \{i \mid w_i = 0\}$. If not equal, $Tv \neq Tw$.
- Delete the columns ω_i of M with $v_i = 0$.
- Compute a lattice basis for

$$L = \{\alpha \in \mathbb{Z}^n \mid M\alpha = 0\}.$$

- For $\alpha \in L$, check $x^\alpha(v) = x^\alpha(w)$. If all equal, $Tv = Tw$. If not, $Tv \neq Tw$.

Computing a lattice basis

Computing a lattice basis

- **Smith Normal Form:** Let $M \in \mathbb{Z}^{d \times n}$. There exist unimodular matrices $U \in \mathbb{Z}^{d \times d}$, $W \in \mathbb{Z}^{n \times n}$ such that UMW is diagonal.

Computing a lattice basis

- Smith Normal Form: Let $M \in \mathbb{Z}^{d \times n}$. There exist unimodular matrices $U \in \mathbb{Z}^{d \times d}$, $W \in \mathbb{Z}^{n \times n}$ such that UMW is diagonal.

- Kannan & Bachem ('79): The diagonal matrix and the multiplier matrices U, W can be computed in polynomial time.

Computing a lattice basis

- Smith Normal Form: Let $M \in \mathbb{Z}^{d \times n}$. There exist unimodular matrices $U \in \mathbb{Z}^{d \times d}$, $W \in \mathbb{Z}^{n \times n}$ such that UMW is diagonal.

- Kannan & Bachem ('79): The diagonal matrix and the multiplier matrices U, W can be computed in polynomial time.

- W gives an isomorphism between the lattices
  $\{\alpha \in \mathbb{Z}^n \mid (UMW)\alpha = 0\} \;\to\; \{\alpha \in \mathbb{Z}^n \mid M\alpha = 0\}$.

Computing a lattice basis

- Smith Normal Form: Let $M \in \mathbb{Z}^{d \times n}$. There exist unimodular matrices $U \in \mathbb{Z}^{d \times d}$, $W \in \mathbb{Z}^{n \times n}$ such that UMW is diagonal.

- Kannan & Bachem ('79): The diagonal matrix and the multiplier matrices U, W can be computed in polynomial time.

- W gives an isomorphism between the lattices $\{\alpha \in \mathbb{Z}^n \mid (UMW)\alpha = 0\} \;\to\; \{\alpha \in \mathbb{Z}^n \mid M\alpha = 0\}$.

- As UMW is diagonal, a basis for the latter lattice is given by the columns $W^{(r+1)}, \ldots, W^{(n)}$ of W, where r is the rank of M.

Laurent monomial equivalence

Laurent monomial equivalence

- Input: $\alpha \in \mathbb{Z}^n, v, w \in \mathbb{Q}(i)^n$ with bit-lengths bounded by b
  Decide $x^\alpha(v) = x^\alpha(w)$.

Laurent monomial equivalence

- Input: $\alpha \in \mathbb{Z}^n, v, w \in \mathbb{Q}(i)^n$ with bit-lengths bounded by b
  Decide $x^\alpha(v) = x^\alpha(w)$.

- $\log x^\alpha(v) = O(b2^b)$. Hence it is not efficient to actually compute $x^\alpha(v)$.

Laurent monomial equivalence

- Input: $\alpha \in \mathbb{Z}^n, v, w \in \mathbb{Q}(i)^n$ with bit-lengths bounded by b
  Decide $x^\alpha(v) = x^\alpha(w)$.

- $\log x^\alpha(v) = O(b2^b)$. Hence it is not efficient to actually compute $x^\alpha(v)$.

- Factor refinement: Pick two entries, one from left and one from right. Quotient out the gcd.
  $v_1^{\alpha_1} v_2^{\alpha_2} \ldots v_n^{\alpha_n} \stackrel{?}{=} w_1^{\alpha_1} w_2^{\alpha_2} \ldots w_n^{\alpha_n} \;\rightarrow\; d = \gcd(v_i, w_j)$
  $\rightarrow\; d^{\alpha_i - \alpha_j} v_1^{\alpha_1} \ldots (v_i/d)^{\alpha_i} \ldots v_n^{\alpha_n} \stackrel{?}{=} w_1^{\alpha_1} \ldots (w_j/d)^{\alpha_j} \ldots w_n^{\alpha_n}$

Laurent monomial equivalence

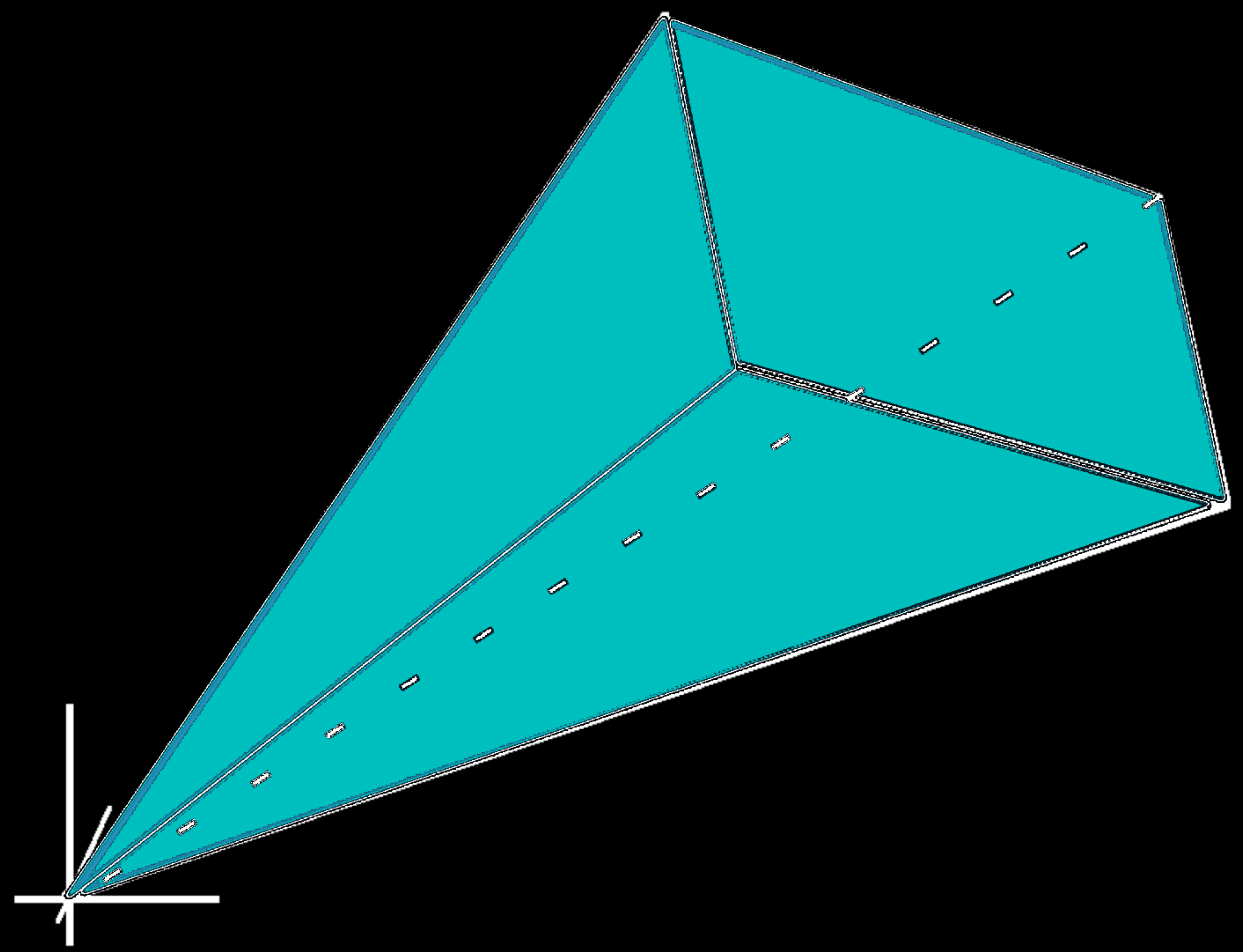
- Input: $\alpha \in \mathbb{Z}^n, v, w \in \mathbb{Q}(i)^n$ with bit-lengths bounded by b
  Decide $x^\alpha(v) = x^\alpha(w)$.

- $\log x^\alpha(v) = O(b2^b)$. Hence it is not efficient to actually compute $x^\alpha(v)$.

- Factor refinement: Pick two entries, one from left and one from right. Quotient out the gcd.
  $v_1^{\alpha_1} v_2^{\alpha_2} \dots v_n^{\alpha_n} \stackrel{?}{=} w_1^{\alpha_1} w_2^{\alpha_2} \dots w_n^{\alpha_n} \;\rightarrow\; d = \gcd(v_i, w_j)$
  $\rightarrow\; d^{\alpha_i - \alpha_j} v_1^{\alpha_1} \dots (v_i/d)^{\alpha_i} \dots v_n^{\alpha_n} \stackrel{?}{=} w_1^{\alpha_1} \dots (w_j/d)^{\alpha_j} \dots w_n^{\alpha_n}$

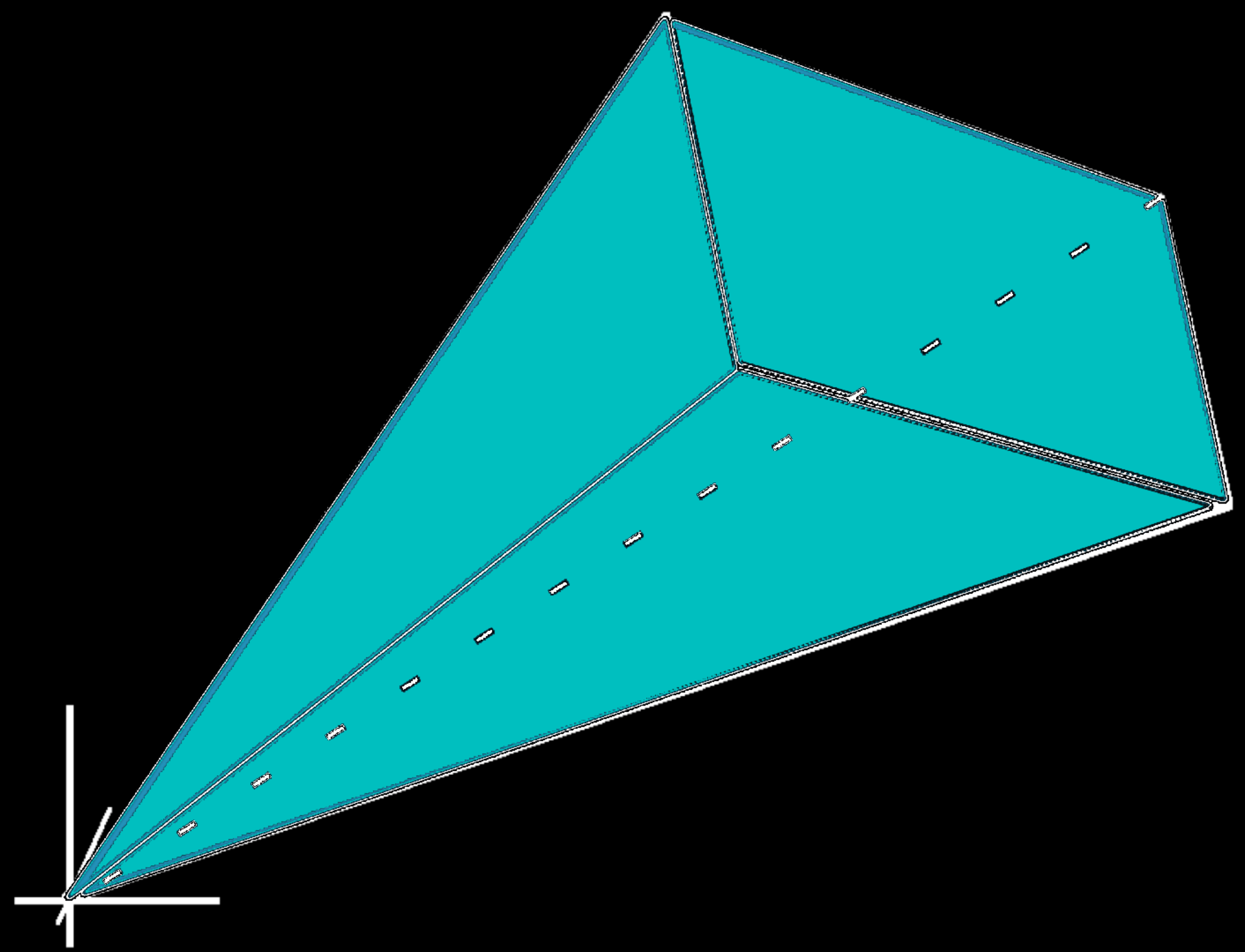
- Theorem (Ge '93): Laurent monomial equivalence over a number field K can be tested in polynomial time.

- The *Newton cone* of v is the convex cone generated by the weights of v.
$C(v) = \{ \sum_{v_i \neq 0} \lambda_i \omega_i \mid \lambda_i \geq 0 \} \subset \mathbb{R}^d.$



- The *Newton cone* of v is the convex cone generated by the weights of v.
$C(v) = \{ \sum_{v_i \neq 0} \lambda_i \omega_i \mid \lambda_i \geq 0 \} \subset \mathbb{R}^d$.

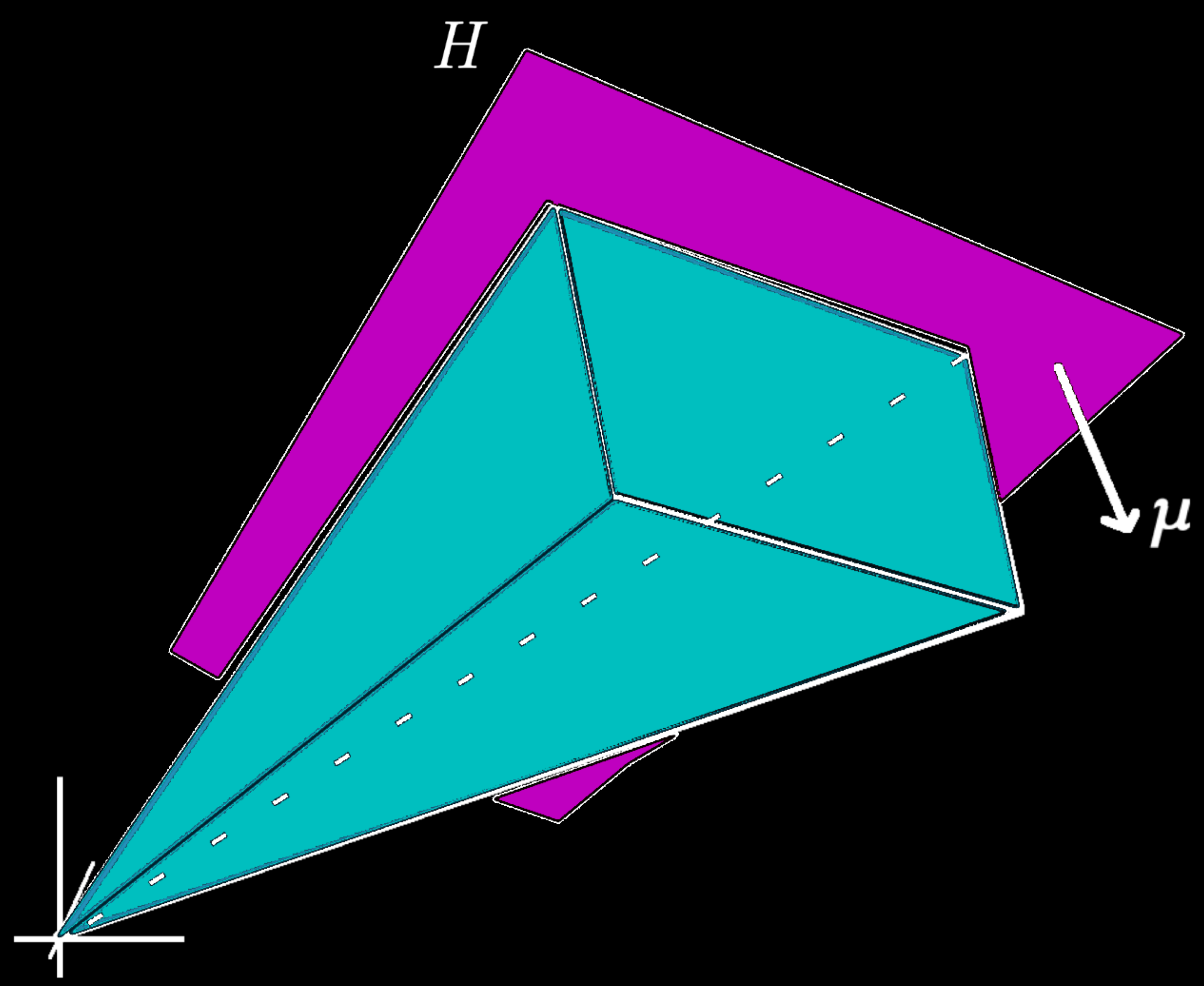
- **Proposition:** There is a bijection
$\{\text{Faces of } C(v)\} \leftrightarrow \{\text{Orbits in } \overline{Tv}\}$.



- The *Newton cone* of v is the convex cone generated by the weights of v.
$C(v) = \{\sum_{v_i \neq 0} \lambda_i \omega_i \mid \lambda_i \geq 0\} \subset \mathbb{R}^d$.

- Proposition: There is a bijection
$\{\text{Faces of } C(v)\} \leftrightarrow \{\text{Orbits in } \overline{Tv}\}$.

- $F \leftarrow$ a face of $C(v)$
$H \leftarrow$ supporting hyperplane for F
$\mu \in \mathbb{Z}^d \leftarrow$ a normal vector for H
$v_F := \lim_{\epsilon \to 0} (\epsilon^{\mu_1}, \epsilon^{\mu_2}, \ldots, \epsilon^{\mu_d}) \cdot v = \lim_{\epsilon \to 0} (\epsilon^{\mu \cdot \omega_1} v_1, \ldots, \epsilon^{\mu \cdot \omega_n} v_n)$.

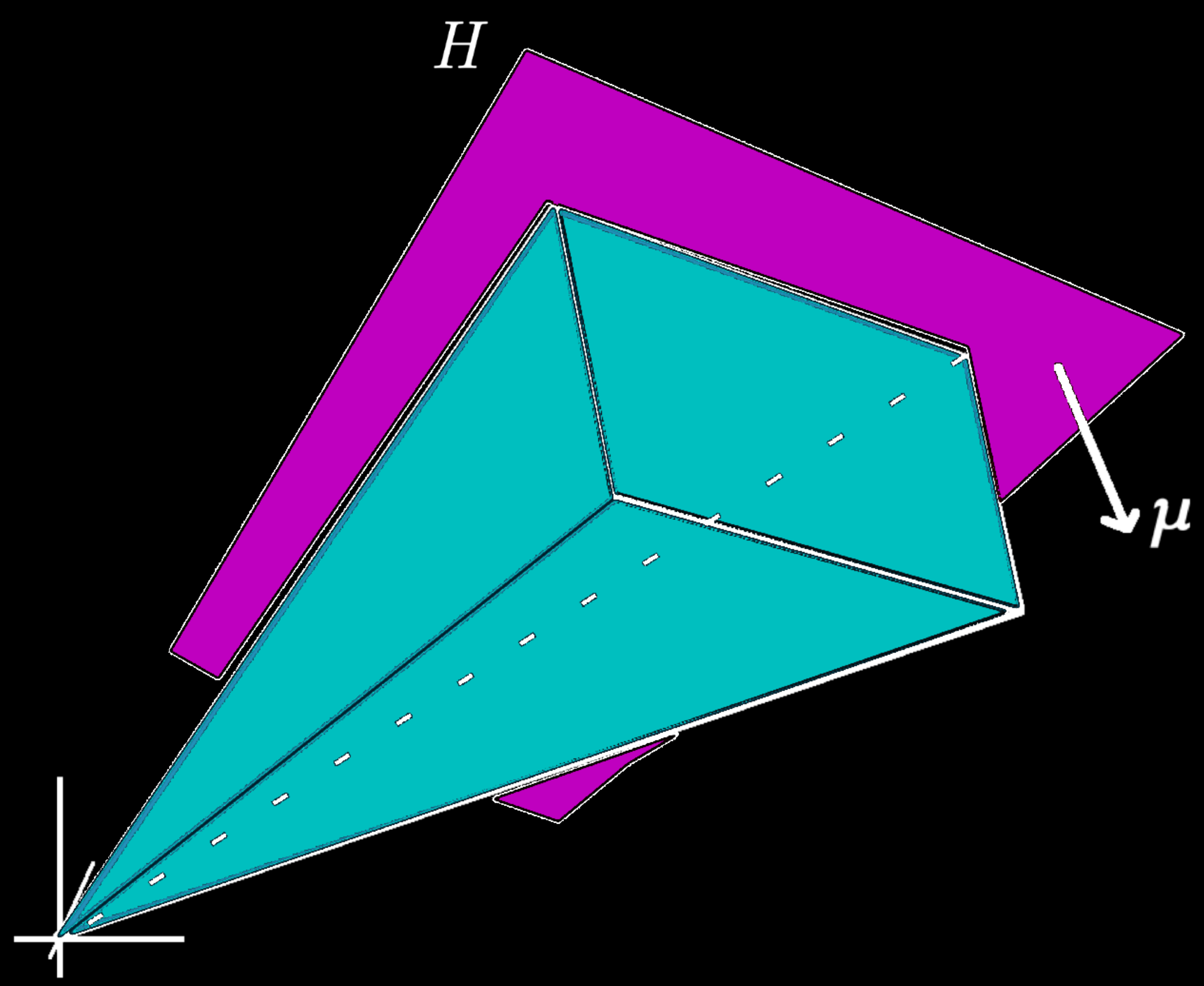


- The *Newton cone* of v is the convex cone generated by the weights of v.
  $C(v) = \{ \sum_{v_i \neq 0} \lambda_i \omega_i \mid \lambda_i \geq 0 \} \subset \mathbb{R}^d$.

- Proposition: There is a bijection
  $\{\text{Faces of } C(v)\} \leftrightarrow \{\text{Orbits in } \overline{Tv}\}$.

- $F \leftarrow$ a face of $C(v)$
  $H \leftarrow$ supporting hyperplane for F
  $\mu \in \mathbb{Z}^d \leftarrow$ a normal vector for H
  $v_F := \lim_{\epsilon \to 0} (\epsilon^{\mu_1}, \epsilon^{\mu_2}, \ldots, \epsilon^{\mu_d}) \cdot v = \lim_{\epsilon \to 0} (\epsilon^{\mu \cdot \omega_1} v_1, \ldots, \epsilon^{\mu \cdot \omega_n} v_n)$.

- Hilbert-Mumford Criterion: If $Tw \subset \overline{Tv}$, then there exists $\mu \in \mathbb{Z}^d$ with
  $$\lim_{\epsilon \to 0} (\epsilon^{\mu_1}, \epsilon^{\mu_2}, \ldots, \epsilon^{\mu_d}) \cdot v \in Tw .$$

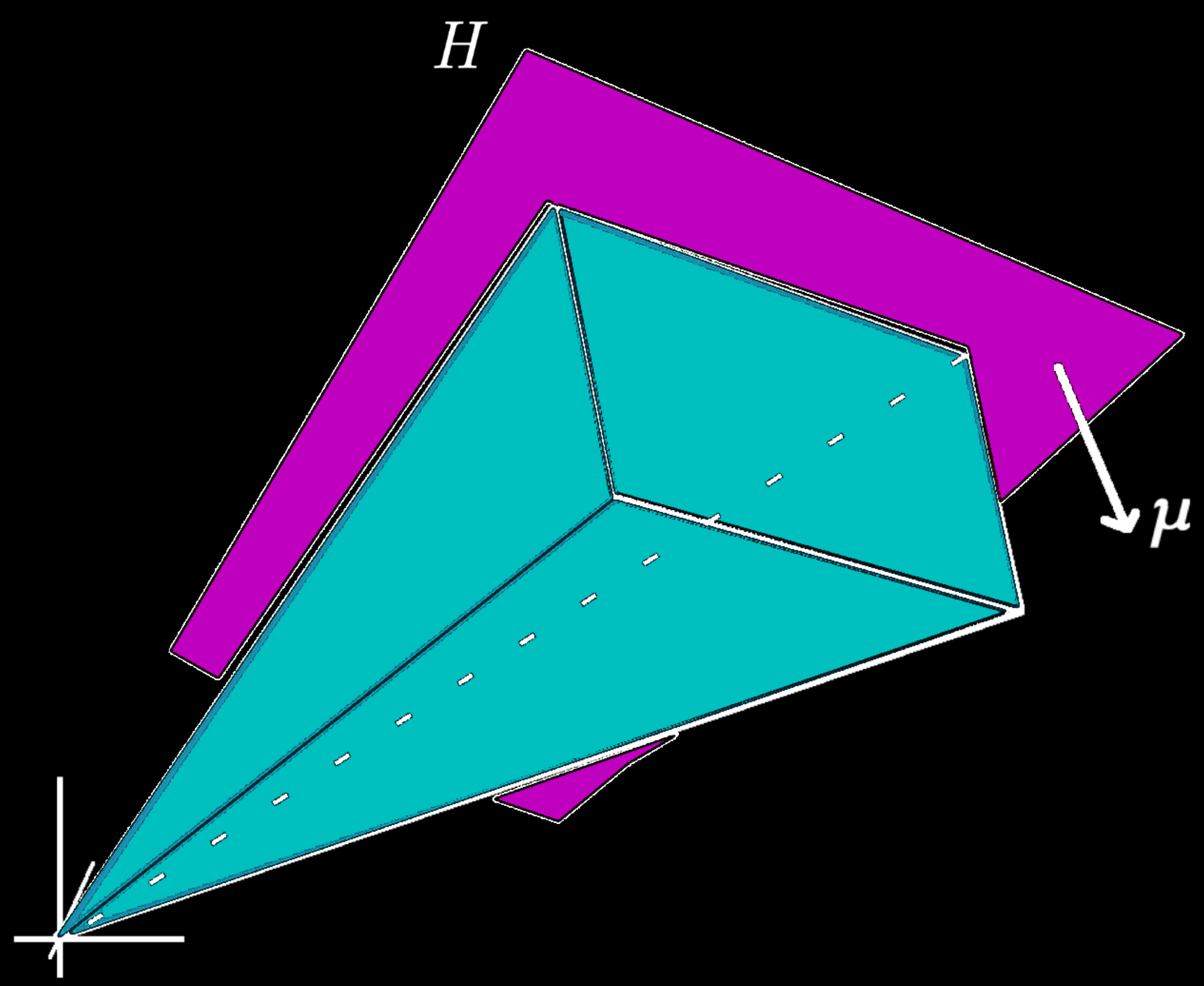


- The *Newton cone* of v is the convex cone generated by the weights of v.
  $C(v) = \{ \sum_{v_i \neq 0} \lambda_i \omega_i \mid \lambda_i \geq 0\} \subset \mathbb{R}^d$.

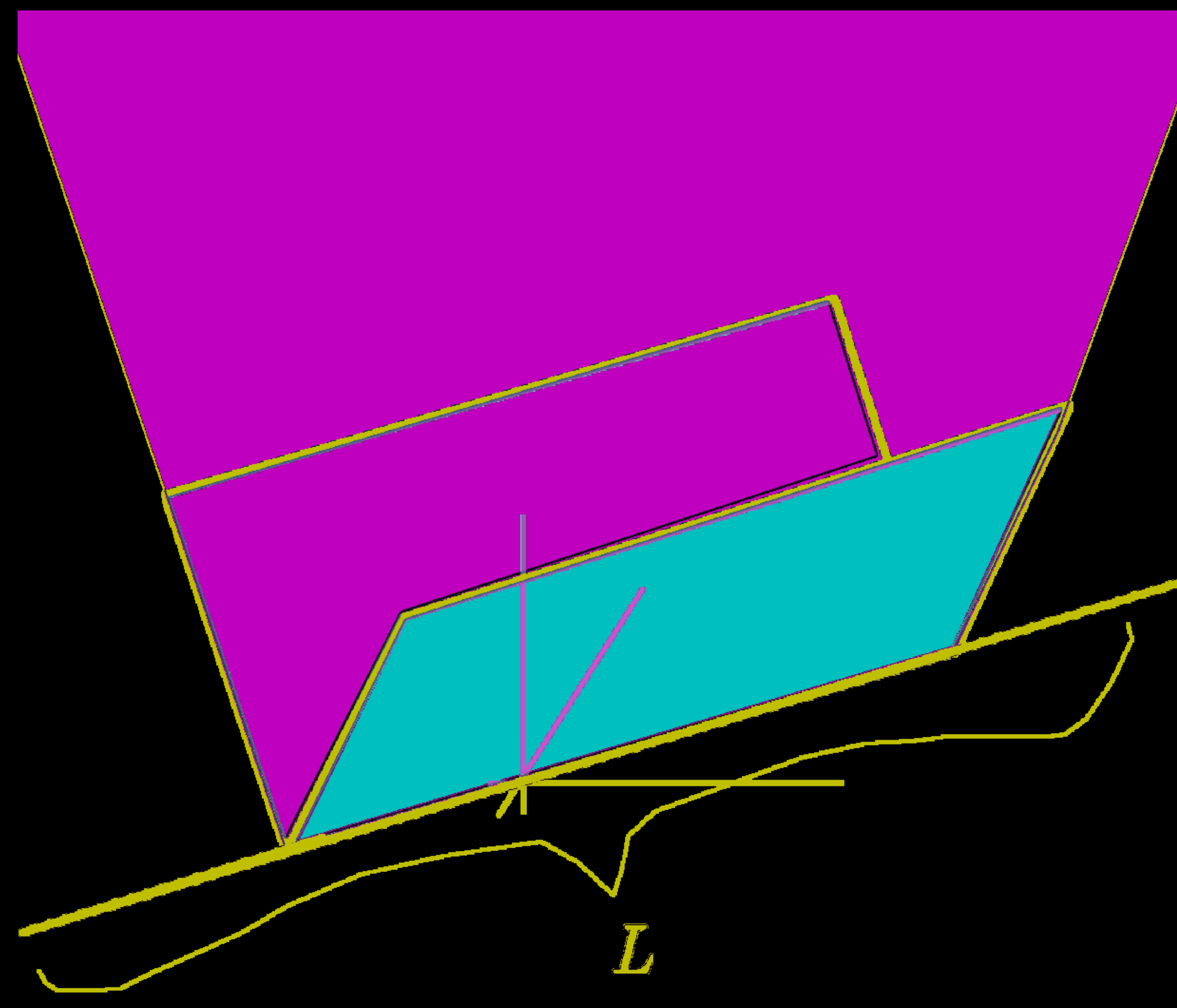
- Proposition: There is a bijection
  $\{\text{Faces of } C(v)\} \leftrightarrow \{\text{Orbits in } \overline{Tv}\}$.

- $F \leftarrow$ a face of $C(v)$
  $H \leftarrow$ supporting hyperplane for F
  $\mu \in \mathbb{Z}^d \leftarrow$ a normal vector for H
  $v_F := \lim_{\epsilon \to 0} (\epsilon^{\mu_1}, \epsilon^{\mu_2}, \ldots, \epsilon^{\mu_d}) \cdot v = \lim_{\epsilon \to 0} (\epsilon^{\mu \cdot \omega_1} v_1, \ldots, \epsilon^{\mu \cdot \omega_n} v_n)$.

- Hilbert-Mumford Criterion: If $Tw \subset \overline{Tv}$, then there exists $\mu \in \mathbb{Z}^d$ with
  $$\lim_{\epsilon \to 0} (\epsilon^{\mu_1}, \epsilon^{\mu_2}, \ldots, \epsilon^{\mu_d}) \cdot v \in Tw .$$

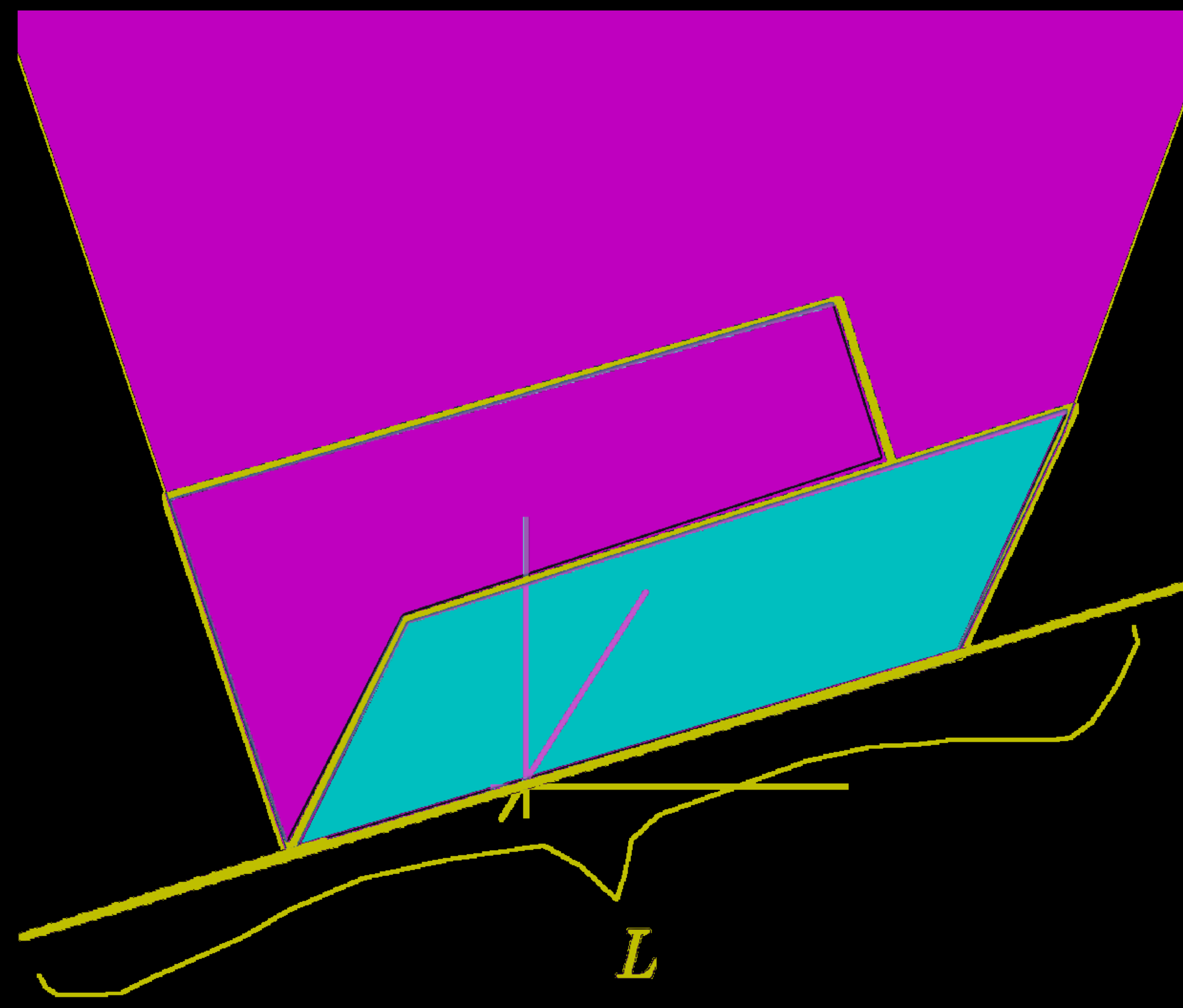


- Corollary:
  $$\begin{aligned} \{\text{Faces of } C(v)\} &\rightarrow \{\text{Orbits in } \overline{Tv}\} \\ F &\mapsto Tv_F \end{aligned}$$
  is a bijection.


L

- **Theorem:** $\overline{Tv}$ contains a *unique closed orbit* $T\tilde{v}$. Moreover, we have

$$\overline{Tv} \cap \overline{Tw} \neq \emptyset \quad \Longleftrightarrow \quad T\tilde{v} = T\tilde{w}\,.$$

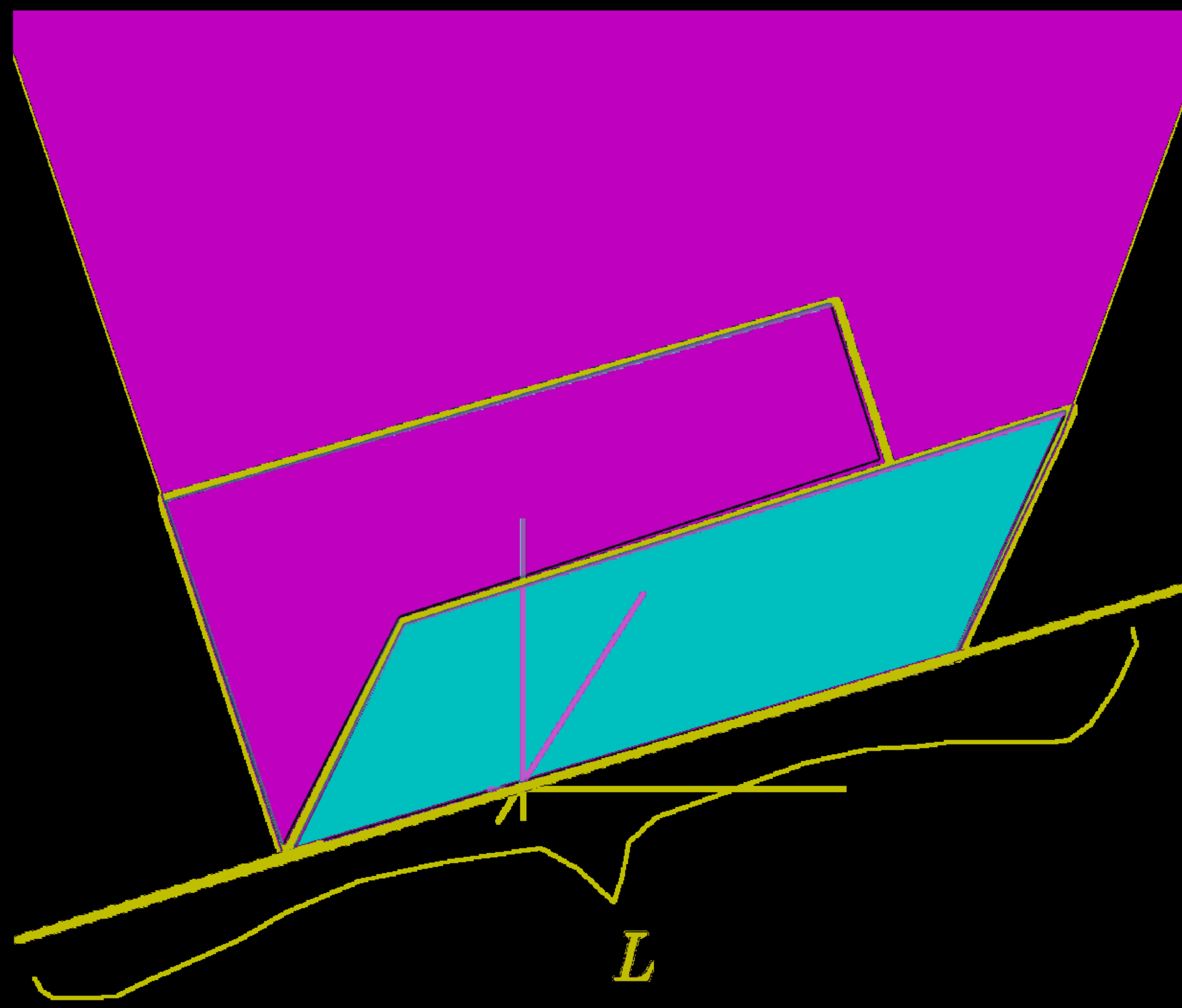


- Theorem: $\overline{Tv}$ contains a *unique closed orbit* $T\tilde{v}$. Moreover, we have

$$\overline{Tv} \cap \overline{Tw} \neq \emptyset \iff T\tilde{v} = T\tilde{w}.$$

- Define the *lineality space* of a cone C as the the largest subspace contained in C.

$$L = C \cap (-C).$$



- Theorem: $\overline{Tv}$ contains a *unique closed orbit* $T\tilde{v}$. Moreover, we have

$$\overline{Tv} \cap \overline{Tw} \neq \emptyset \iff T\tilde{v} = T\tilde{w}\,.$$

- Define the *lineality space* of a cone C as the the largest subspace contained in C.

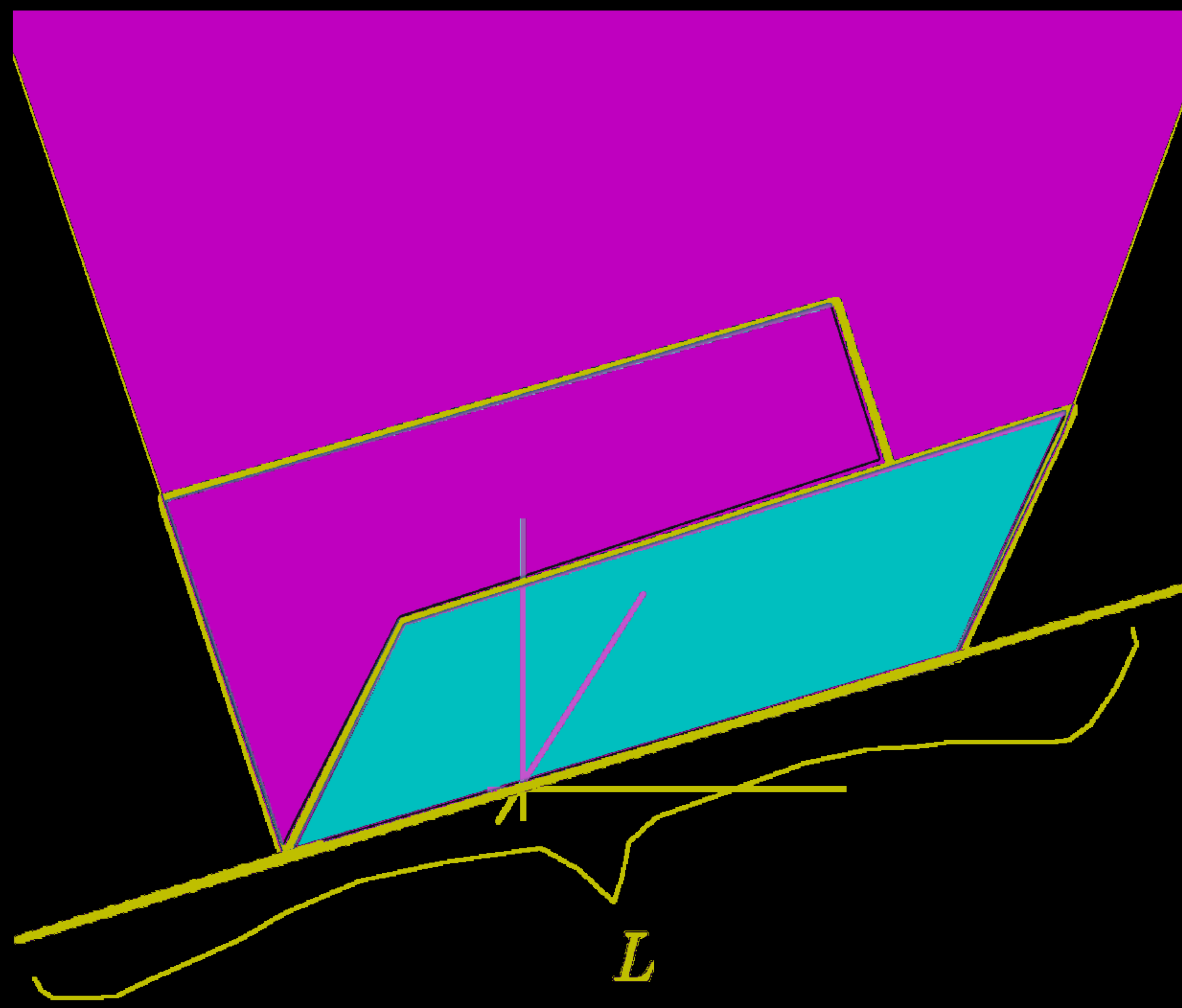
$$L = C \cap (-C)\,.$$

- Proposition: Let $L(v)$ denote the lineality space of $C(v)$. Then

$$Tv_{L(v)} = \text{the unique closed orbit in } \overline{Tv}\,.$$

Moreover, $v_{L(v)}$ can be constructed in polynomial time using *linear programming*.

$$(v_{L(v)})_i = \begin{cases} v_i & \text{if } -\omega_i \in C(v) \\ 0 & \text{otherwise.} \end{cases}$$



- **Theorem:** $\overline{Tv}$ contains a *unique closed orbit* $T\tilde{v}$. Moreover, we have

$$\overline{Tv} \cap \overline{Tw} \neq \emptyset \iff T\tilde{v} = T\tilde{w}.$$

- Define the *lineality space* of a cone C as the the largest subspace contained in C.

$$L = C \cap (-C).$$

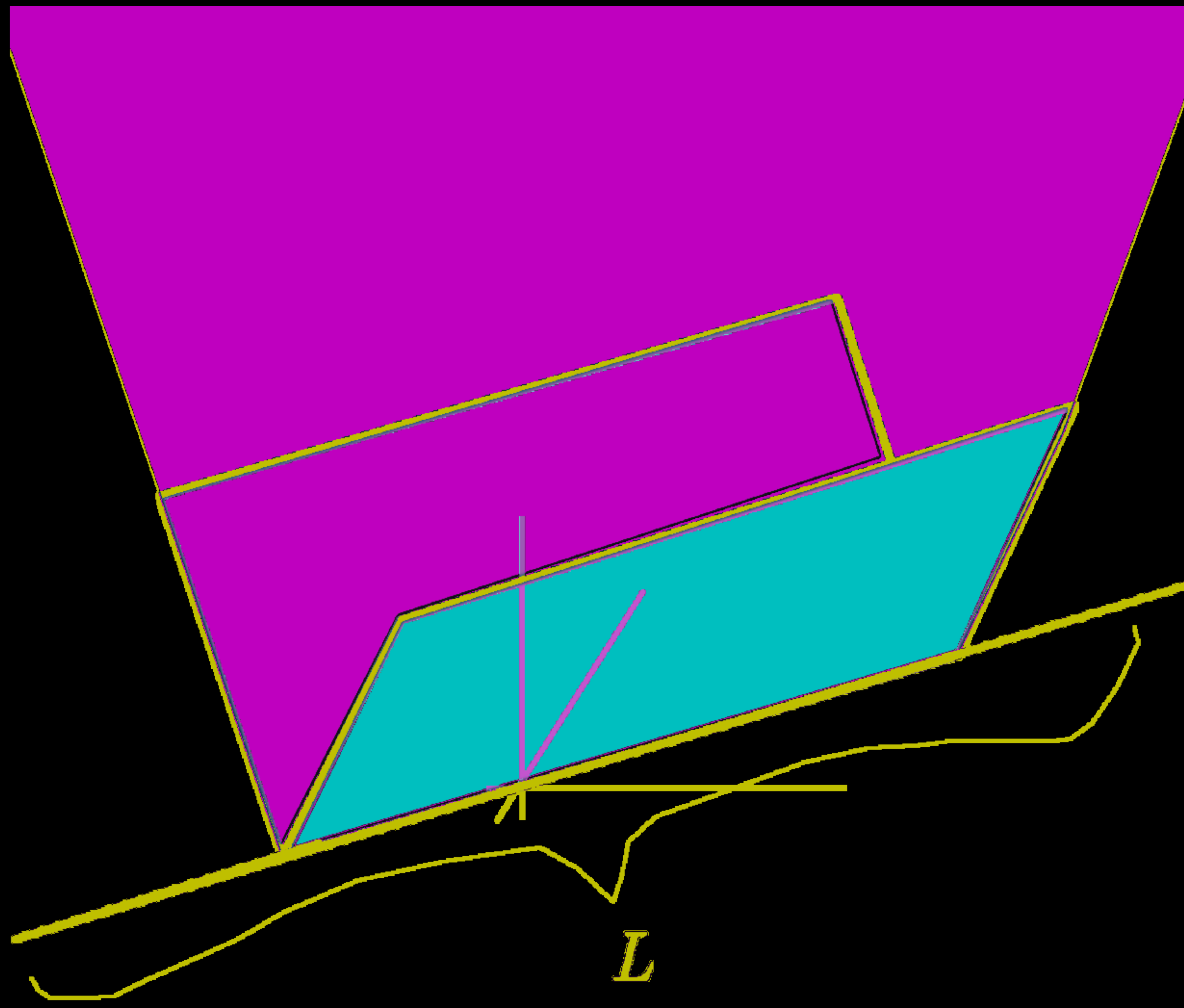
- **Proposition:** Let $L(v)$ denote the lineality space of $C(v)$. Then

$$Tv_{L(v)} = \text{the unique closed orbit in } \overline{Tv}.$$

Moreover, $v_{L(v)}$ can be constructed in polynomial time using *linear programming*.

$$(v_{L(v)})_i = \begin{cases} v_i & \text{if } -\omega_i \in C(v) \\ 0 & \text{otherwise.} \end{cases}$$

- **Corollary:** There is a poly-time reduction from orbit closure intersection problem to orbit equality problem.



Algorithm: Orbit Closure Intersection

Algorithm: Orbit Closure Intersection

- Input: $v, w \in \mathbb{Q}(i)^n, M \in \mathbb{Z}^{d \times n}$.

Algorithm: Orbit Closure Intersection

- Input: $v, w \in \mathbb{Q}(i)^n, M \in \mathbb{Z}^{d \times n}$.
- Using linear programming, construct $v_{L(v)}, w_{L(w)}$.

Algorithm: Orbit Closure Intersection

- Input: $v, w \in \mathbb{Q}(i)^n, M \in \mathbb{Z}^{d \times n}$.
- Using linear programming, construct $v_{L(v)}, w_{L(w)}$.
- Use the algorithm for orbit equality to test

$$Tv_{L(v)} = Tw_{L(w)}.$$

Algorithm: Orbit Closure Intersection

- Input: $v, w \in \mathbb{Q}(i)^n, M \in \mathbb{Z}^{d \times n}$.
- Using linear programming, construct $v_{L(v)}, w_{L(w)}$.
- Use the algorithm for orbit equality to test

$$Tv_{L(v)} = Tw_{L(w)}.$$

- If equal, then the orbit closures intersect. If not, the intersection is empty.

Thanks!