Mittagsseminar 12.10.2022

Introduction to Cluster Algebras

Based on ch. 1 - 3 of

(Fomin, Williams & Zelevinsky, 2021)

Talk by Sandro Roch

Mittagsseminar 12.10.2022

Introduction to Cluster Algebras

Based  - 3 of

(Fomin, Willia evinsky, 2021)

For pedestrians!

Talk by Sandro Roch

$(2 \times m)$-matrices with positive maximal minors

Problem I: For $A \in \mathbb{R}^{2\times m}$, check efficiently whether all (2×2)-minors are positive.

$(2 \times m)$-matrices with positive maximal minors

Problem I: For $A \in \mathbb{R}^{2\times m}$, check efficiently whether all (2×2)-minors are positive.

Algebraic background:

- *Grassmanian manifold* $G_{2,m}$ consists of 2-dimensional subspaces of $\mathbb{R}^m$

$(2 \times m)$-matrices with positive maximal minors

Problem I: For $A \in \mathbb{R}^{2\times m}$, check efficiently whether all (2×2)-minors are positive.

Algebraic background:

- *Grassmanian manifold* $G_{2,m}$ consists of 2-dimensional subspaces of $\mathbb{R}^m$
- *Plücker coordinates* $P_{i,j} : A \mapsto \det(A_i, A_j)$ polynomials in matrix entries

$(2 \times m)$-matrices with positive maximal minors

Problem I: For $A \in \mathbb{R}^{2\times m}$, check efficiently whether all (2×2)-minors are positive.

Algebraic background:

- *Grassmanian manifold* $G_{2,m}$ consists of 2-dimensional subspaces of $\mathbb{R}^m$
- *Plücker coordinates* $P_{i,j} : A \mapsto \det(A_i, A_j)$ polynomials in matrix entries
- *Totally positive Grassmanian*

$$G^+_{2,m} := \{\mathrm{rowsp}(A) : A \in \mathbb{R}^{2\times m}, P_{i,j}(A) > 0 \text{ f.a. } i,j\}$$

Appears in Statstical Physics, Integrable Systems, ...

$(2 \times m)$-matrices with positive maximal minors

- Use *Grassmann-Plücker relations:*

$$P_{i,k}P_{j,l} = P_{i,j}P_{k,l} + P_{i,l}P_{j,k}$$

$(2 \times m)$-matrices with positive maximal minors

- Use *Grassmann-Plücker relations:*

$$P_{i,k}P_{j,l} = P_{i,j}P_{k,l} + P_{i,l}P_{j,k}$$

- If $P_{i,k}, P_{i,j}, P_{k,l}, P_{i,l}, P_{j,k} > 0$, then also

$$P_{j,l} = \frac{P_{i,j}P_{k,l} + P_{i,l}P_{j,k}}{P_{i,k}} > 0$$

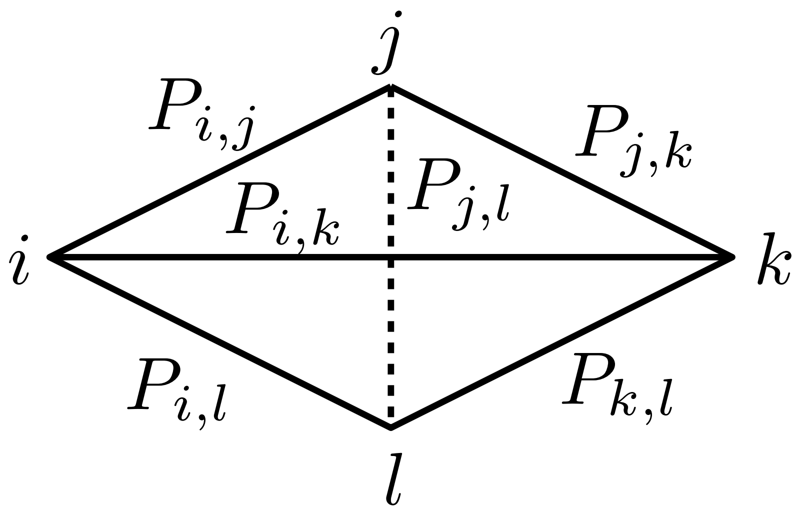
$(2 \times m)$-matrices with positive maximal minors

- Use *Grassmann-Plücker relations:*

$$P_{i,k}P_{j,l} = P_{i,j}P_{k,l} + P_{i,l}P_{j,k}$$

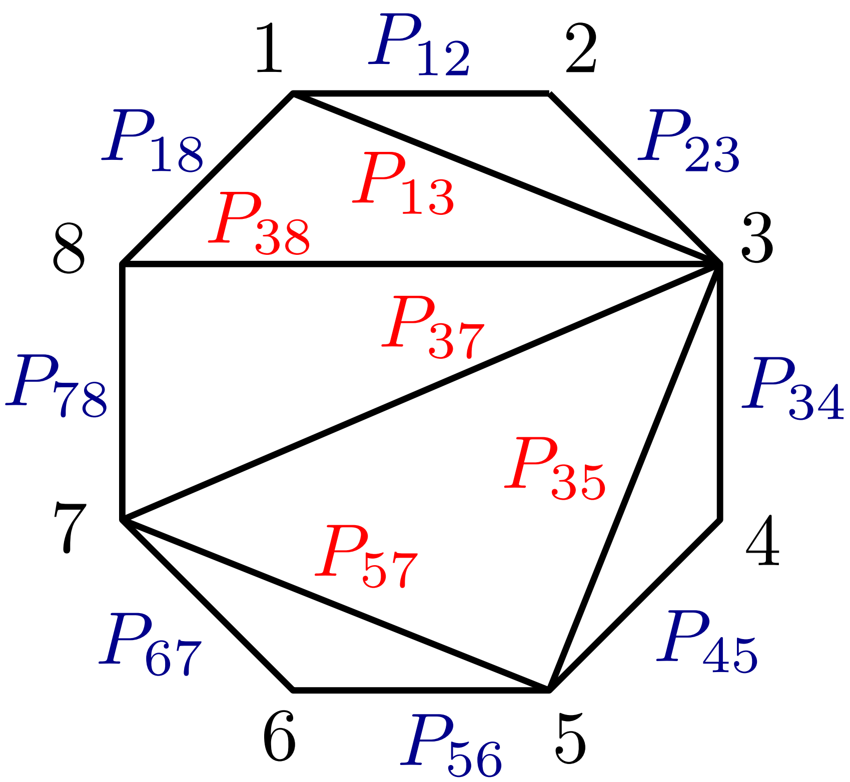
- If $P_{i,k}, P_{i,j}, P_{k,l}, P_{i,l}, P_{j,k} > 0$, then also

$$P_{j,l} = \frac{P_{i,j}P_{k,l} + P_{i,l}P_{j,k}}{P_{i,k}} > 0$$



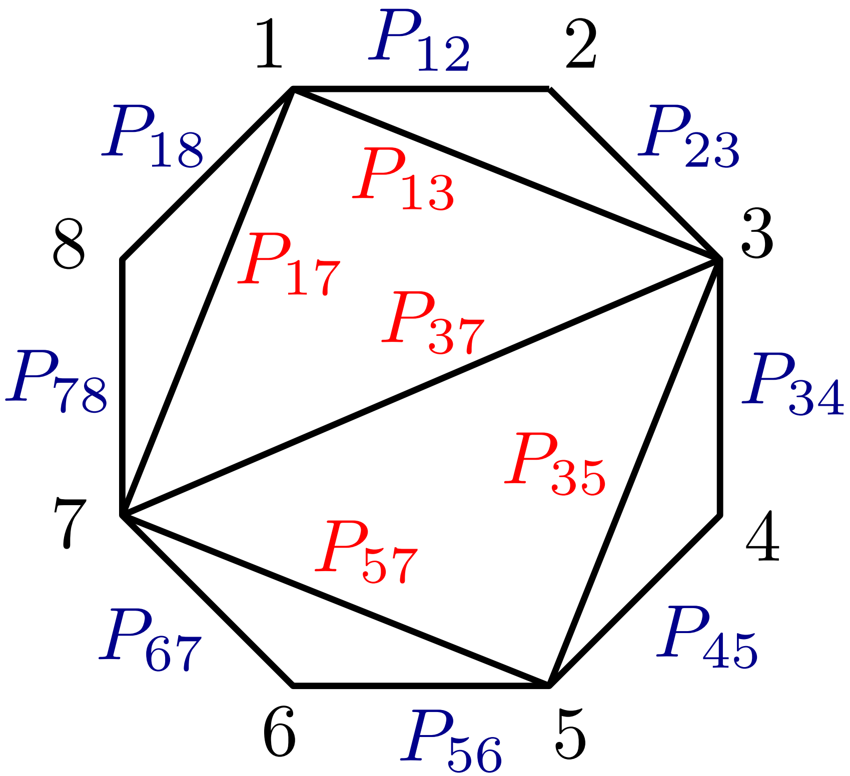
$(2 \times m)$-matrices with positive maximal minors

- **Claim:** All (2×2)-minors are positive if and only if all $2m - 3$ (2×2)-minors appearing as edges of a triangulation of an m-gon are positive.



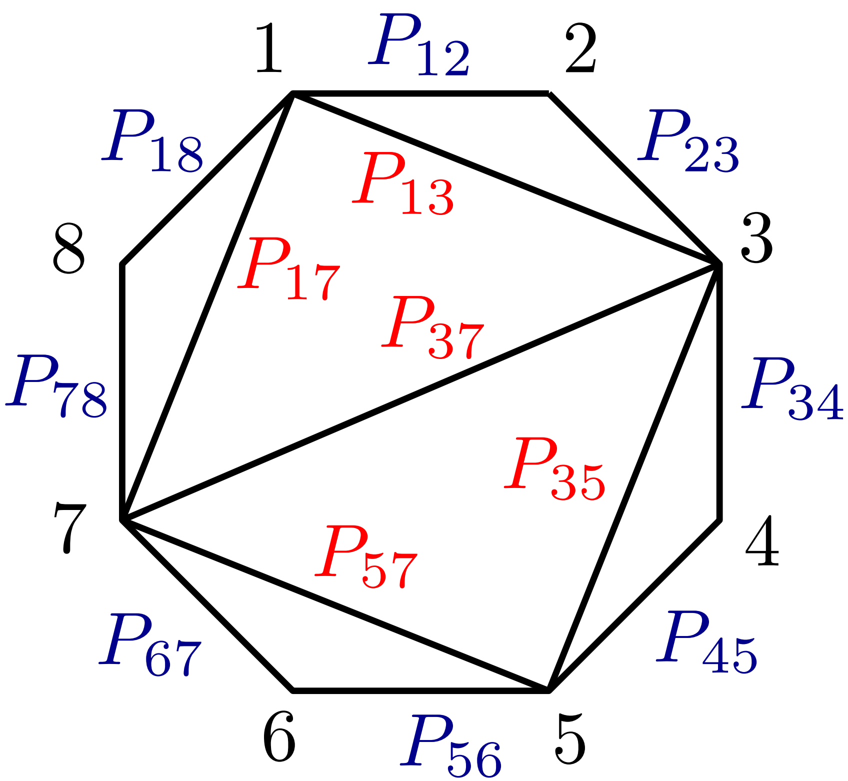
$(2 \times m)$-matrices with positive maximal minors

- **Claim:** All (2×2)-minors are positive if and only if all $2m - 3$ (2×2)-minors appearing as edges of a triangulation of an m-gon are positive.



$(2 \times m)$-matrices with positive maximal minors

- **Claim:** All (2×2)-minors are positive if and only if all $2m-3$ (2×2)-minors appearing as edges of a triangulation of an m-gon are positive.

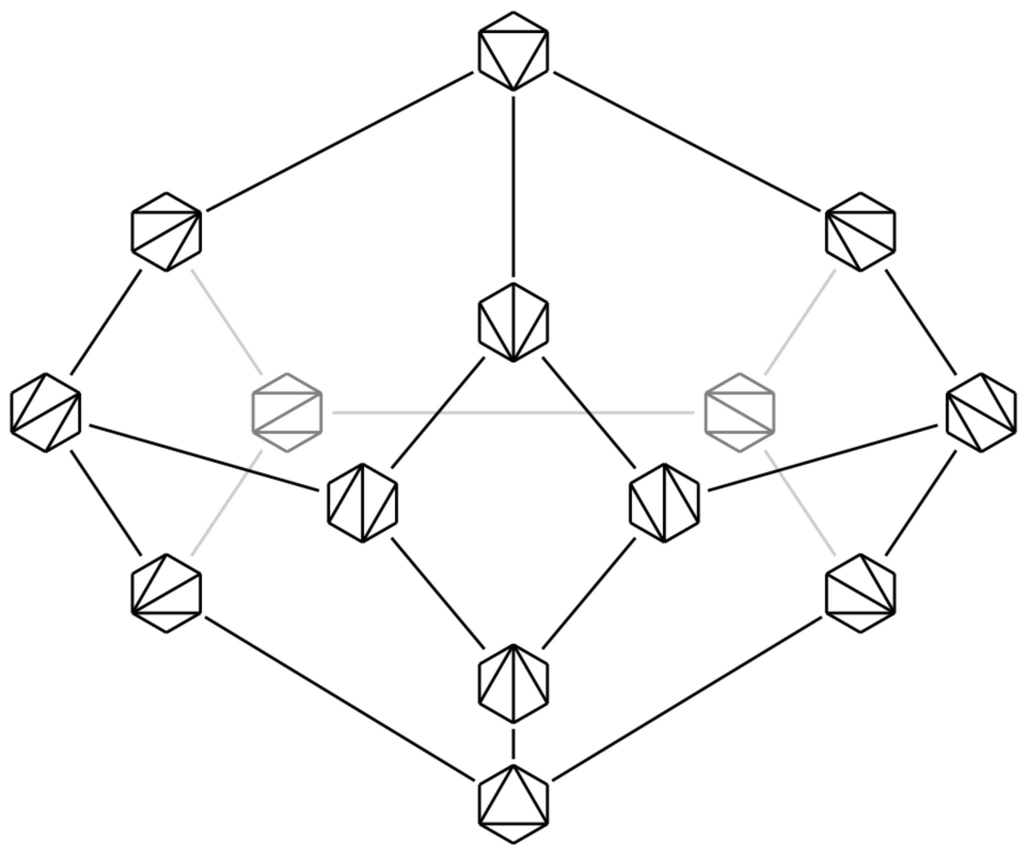


Frozen variables

Cluster variables

$(2 \times m)$-matrices with positive maximal minors

Flip graph of *positivity tests* on $\mathbb{R}^{2\times 6}$
or triangulations of a regular 6-gon:



Flag totally positive matrices

Definition: Let $A \in \mathsf{SL}_n$.

- For $J \subsetneq [n]$, $J \neq \emptyset$ define *flag minor* P_J as

$$P_J : [a_{i,j}] \mapsto \det(a_{i,j} : i \in J, j \leq |J|)$$

- Matrix A *totally flag positive* if $P_J(A) > 0$ for all flag minors P_J.

Flag totally positive matrices

Definition: Let $A \in \mathsf{SL}_n$.

- For $J \subsetneq [n]$, $J \neq \emptyset$ define *flag minor* P_J as

$$P_J : [a_{i,j}] \mapsto \det(a_{i,j} : i \in J, j \leq |J|)$$

- Matrix A *totally flag positive* if $P_J(A) > 0$ for all flag minors P_J.

Algebraic background:

- Subgroup $U \subset \mathsf{SL}_n$ of unipotent lower-triangular matrices acting on SL_n by multiplication.

Flag totally positive matrices

Definition: Let $A \in \mathsf{SL}_n$.

- For $J \subsetneq [n]$, $J \neq \emptyset$ define *flag minor* P_J as

$$P_J : [a_{i,j}] \mapsto \det(a_{i,j} : i \in J, j \leq |J|)$$

- Matrix A *totally flag positive* if $P_J(A) > 0$ for all flag minors P_J.

Algebraic background:

- Subgroup $U \subset \mathsf{SL}_n$ of unipotent lower-triangular matrices acting on SL_n by multiplication.
- Flag minors (P_J) generate ring $\mathbb{C}[\mathsf{SL}_n]^U$ of polynomials $p \in \mathbb{C}[x_{i,j}]$ invariant under this action: $\forall A \in \mathsf{SL}_n \; \forall M \in U : p(M \cdot A) = p(A)$

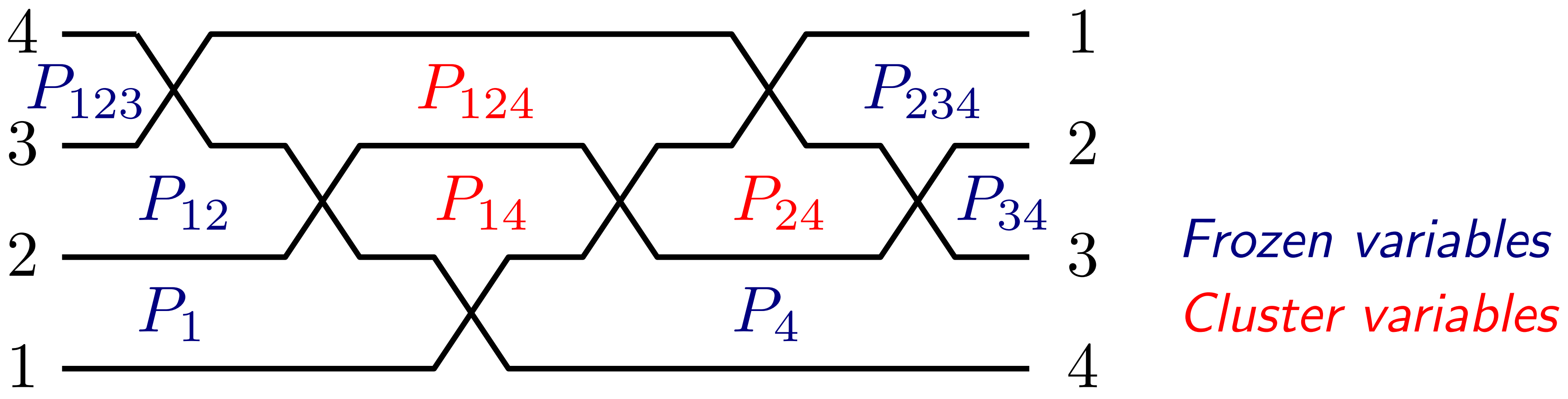
Flag totally positive matrices

Problem II: For $A \in \mathsf{SL}_n$, check efficiently whether A is totally flag positive.

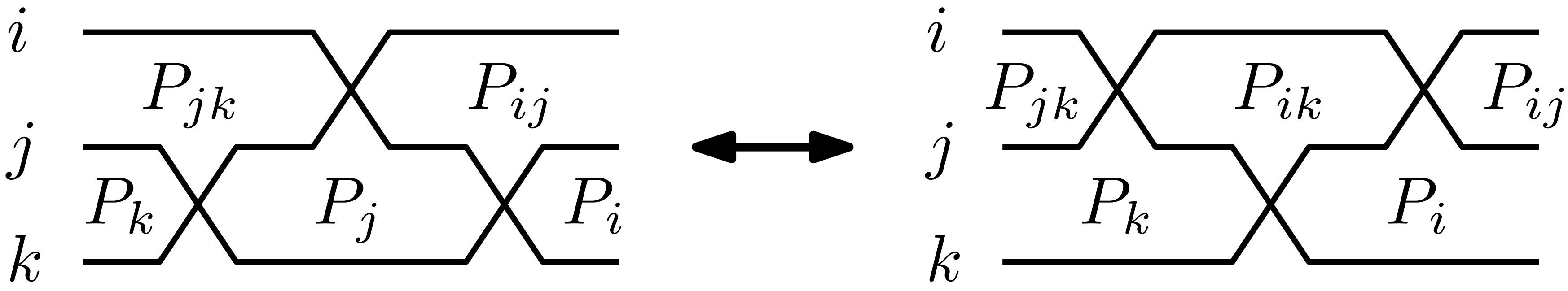
Flag totally positive matrices

Problem II: For $A \in \mathsf{SL}_n$, check efficiently whether A is totally flag positive.

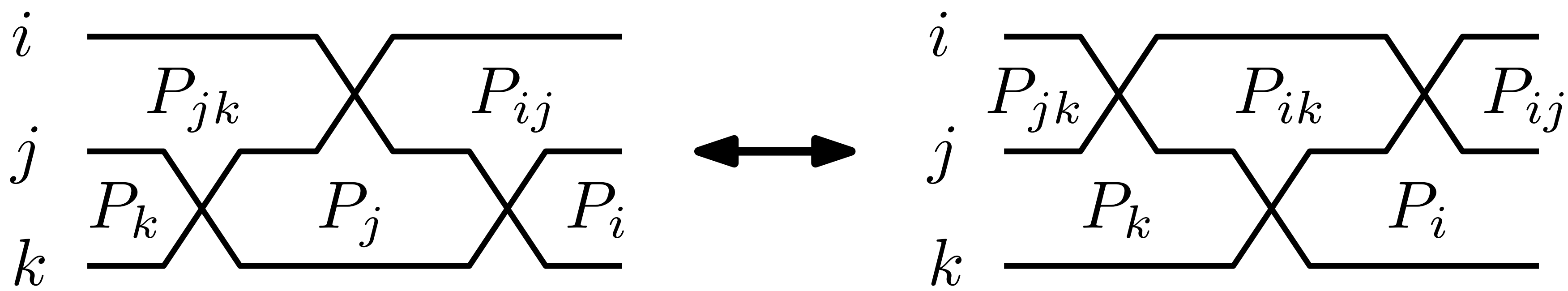
- **Claim:** All flag minors are positive if and only if all $\frac{(n-1)(n+2)}{2}$ flag minors appearing as chambers of a wiring diagram of size n are positive.



Flag totally positive matrices



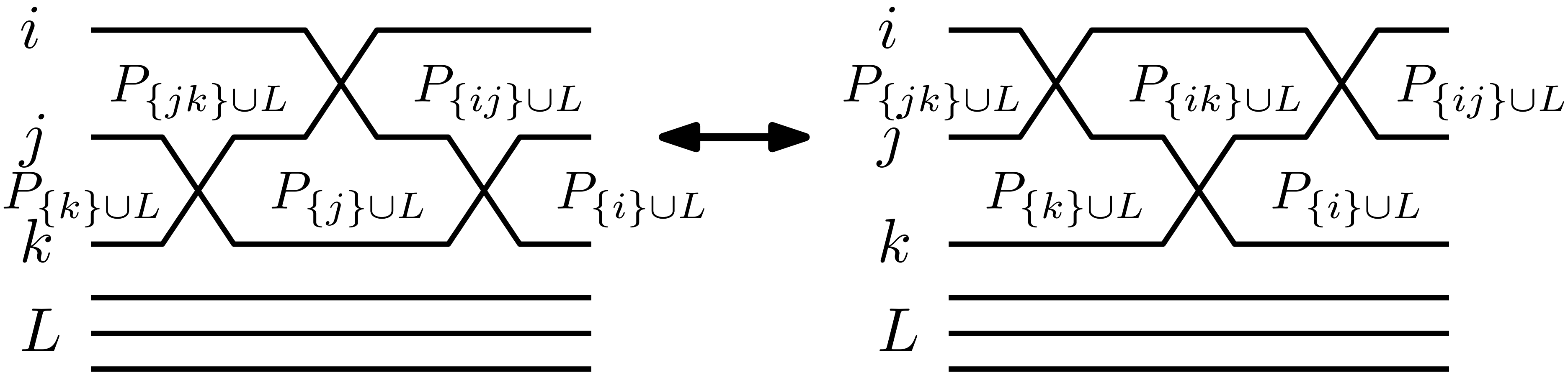
Flag totally positive matrices



- Grassmann-Plücker-Relation:

$$P_j P_{ik} = P_k P_{ij} + P_{jk} P_i$$

Flag totally positive matrices



- Grassmann-Plücker-Relation:

$$P_j P_{ik} = P_k P_{ij} + P_{jk} P_i$$

- With additional pseudolines $L = \{l_1, ..., l_r\}$ below:

$$P_{\{j\}\cup L} P_{\{ik\}\cup L} = P_{\{k\}\cup L} P_{\{ij\}\cup L} + P_{\{jk\}\cup L} P_{\{i\}\cup L}$$

Totally positive matrices in GL_n

Problem III: For $A \in \mathrm{GL}_n$, check efficiently whether A is totally positive, i.e. all minors are positive.

Totally positive matrices in GL_n

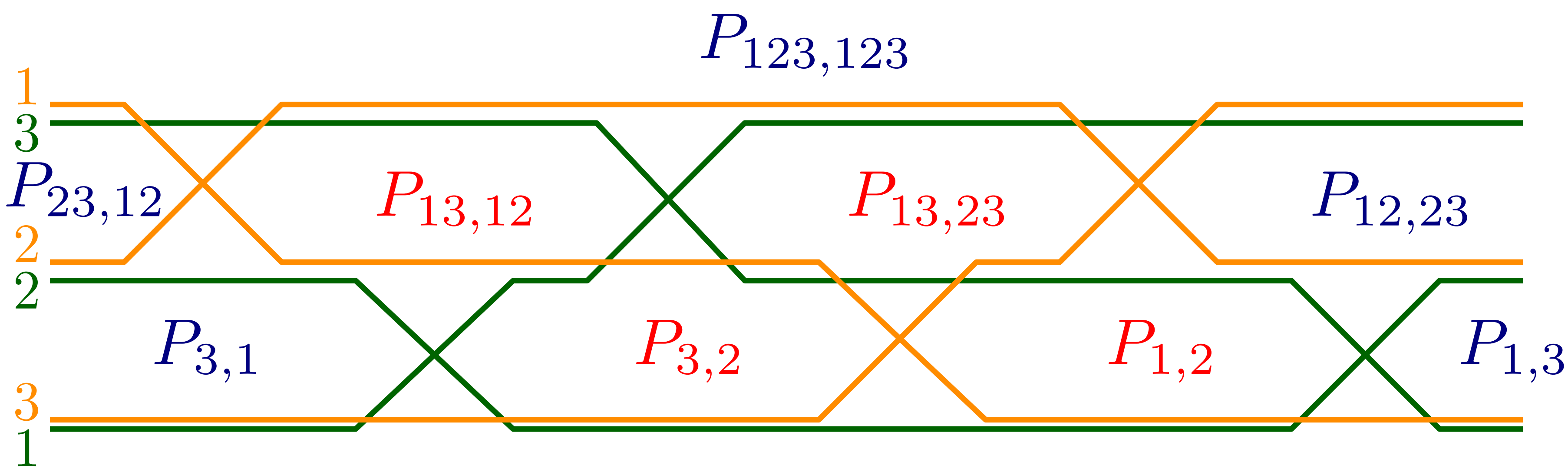
Problem III: For $A \in \mathrm{GL}_n$, check efficiently whether A is totally positive, i.e. all minors are positive.

- **Claim:** All minors are positive if and only if all n^2 minors appearing as cambers of a *double wiring diagram* of size n are positive.

Totally positive matrices in GL_n

Problem III: For $A \in \mathrm{GL}_n$, check efficiently whether A is totally positive, i.e. all minors are positive.

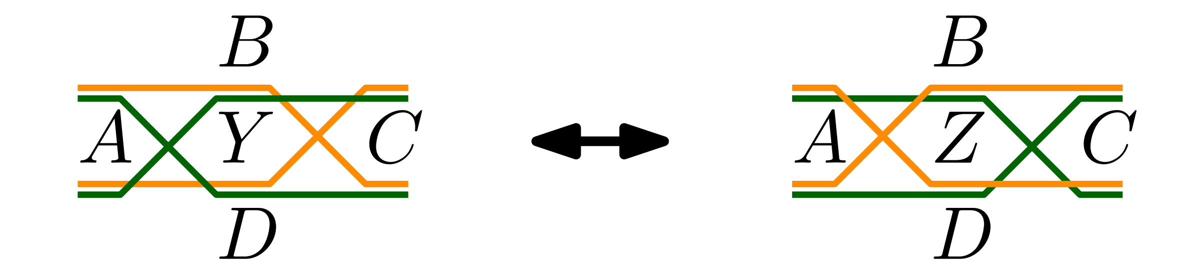
- **Claim:** All minors are positive if and only if all n^2 minors appearing as cambers of a *double wiring diagram* of size n are positive.



Frozen variables *Cluster variables*

Totally positive matrices in GL_n

Double wiring diagrams connected by following flips:



Grassmann-Plücker-relations:

$$YZ = AC + BD$$

Totally positive matrices in GL_n

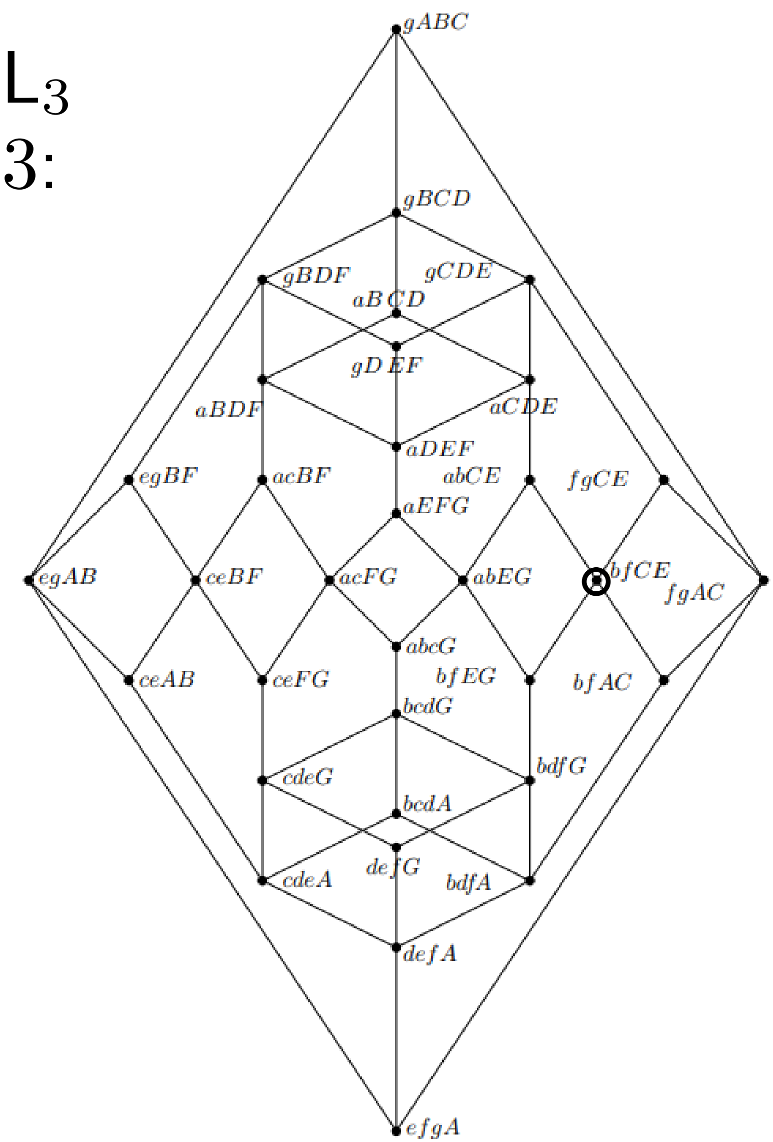
Flip graph of *positivity tests* on GL_3
or *double wiring diagrams* of size 3:

Cluster variables:

$a = P_{1,1}$ $\quad A = P_{23,23}$
$b = P_{1,2}$ $\quad B = P_{23,13}$
$c = P_{2,1}$ $\quad C = P_{13,23}$
$d = P_{2,2}$ $\quad D = P_{13,13}$
$e = P_{2,3}$ $\quad E = P_{13,12}$
$f = P_{3,2}$ $\quad F = P_{12,13}$
$g = P_{3,3}$ $\quad G = P_{12,12}$

Frozen variables:

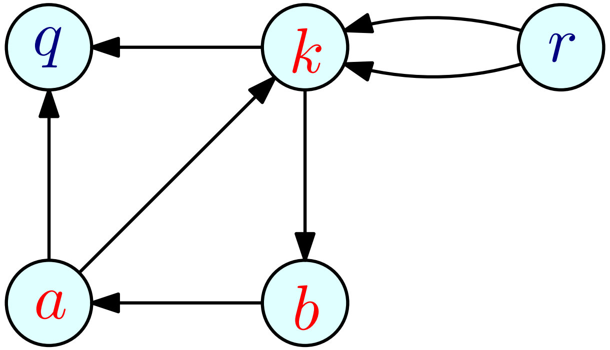
$P_{3,1}, P_{1,3}, P_{23,12}$
$P_{12,23}, P_{123,123}$



Quiver mutations

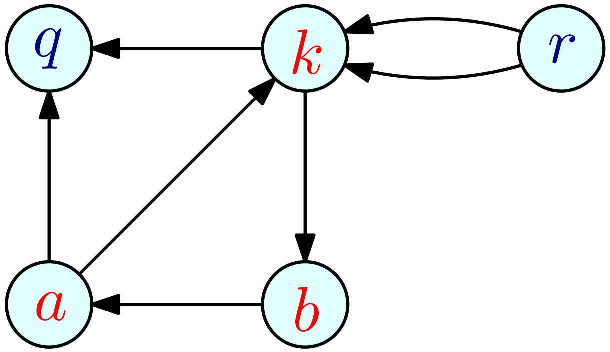
- **Quiver:** Finite directed graph
  - Equally oriented multiedges allowed
  - Loops not allowed
  - Not necessarily connected
  - Vertices either *frozen* or *mutable*.
  - No edges between frozen vertices

Example:



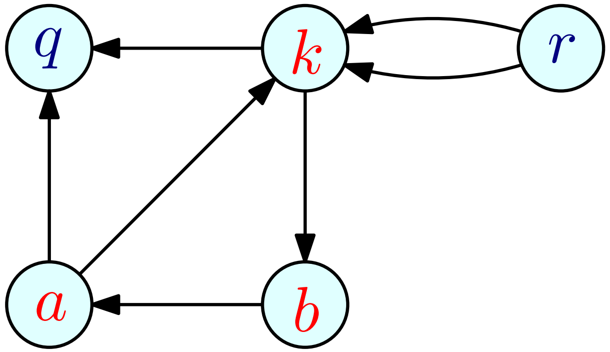
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:



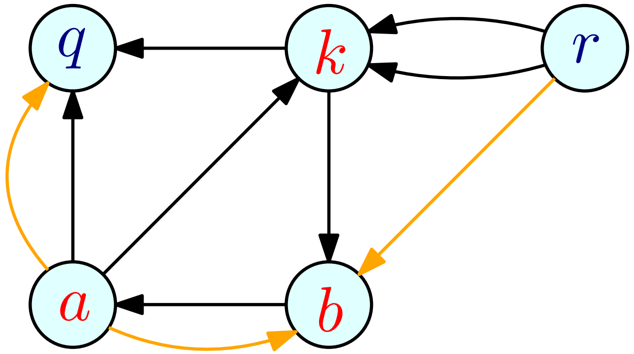
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge“* $i \to j$ (unless both i and j are frozen)



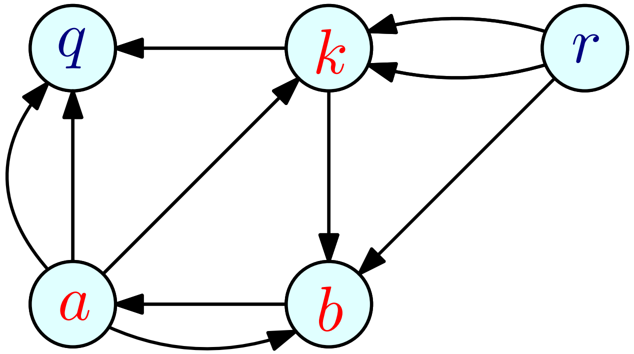
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge“* $i \to j$ (unless both i and j are frozen)



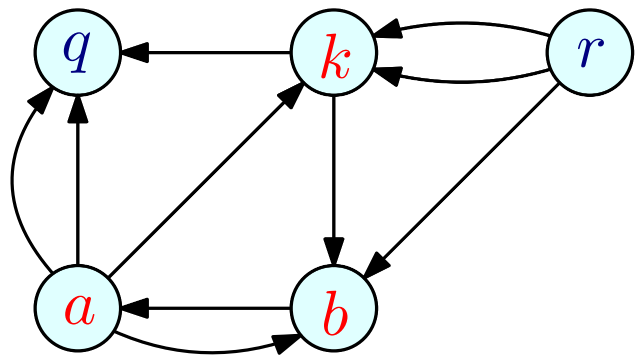
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge“* $i \to j$ (unless both i and j are frozen)



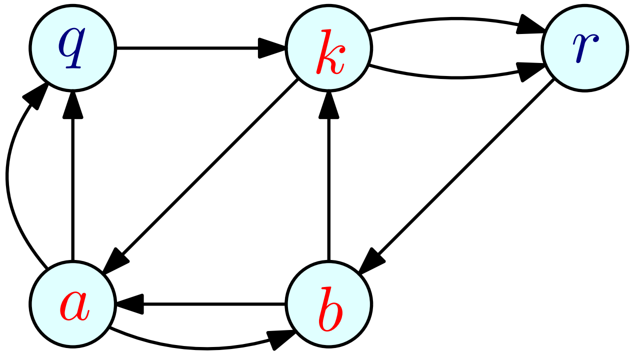
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge“* $i \to j$ (unless both i and j are frozen)
  - **Step II:** Reverse all arcs incident to k



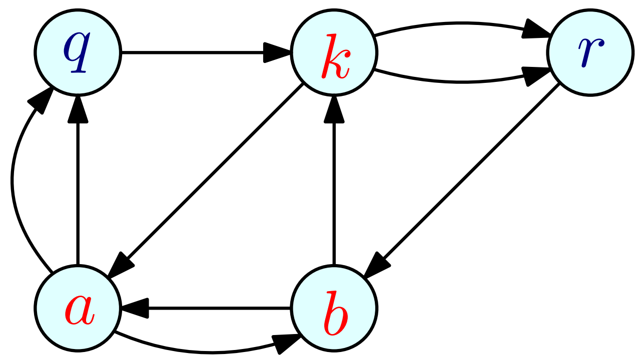
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge“* $i \to j$ (unless both i and j are frozen)
  - **Step II:** Reverse all arcs incident to k



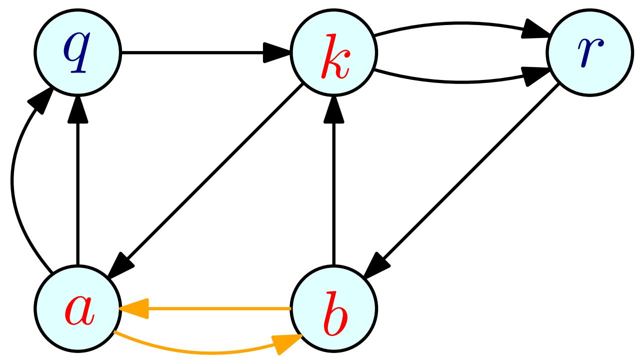
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge“* $i \to j$ (unless both i and j are frozen)
  - **Step II:** Reverse all arcs incident to k
  - **Step III:** Eliminate 2-cycles



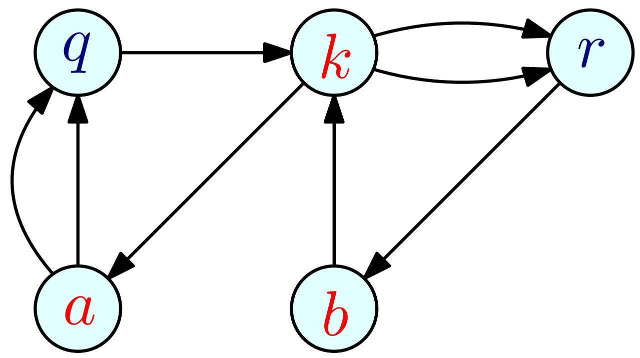
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge"* $i \to j$ (unless both i and j are frozen)
  - **Step II:** Reverse all arcs incident to k
  - **Step III:** Eliminate 2-cycles



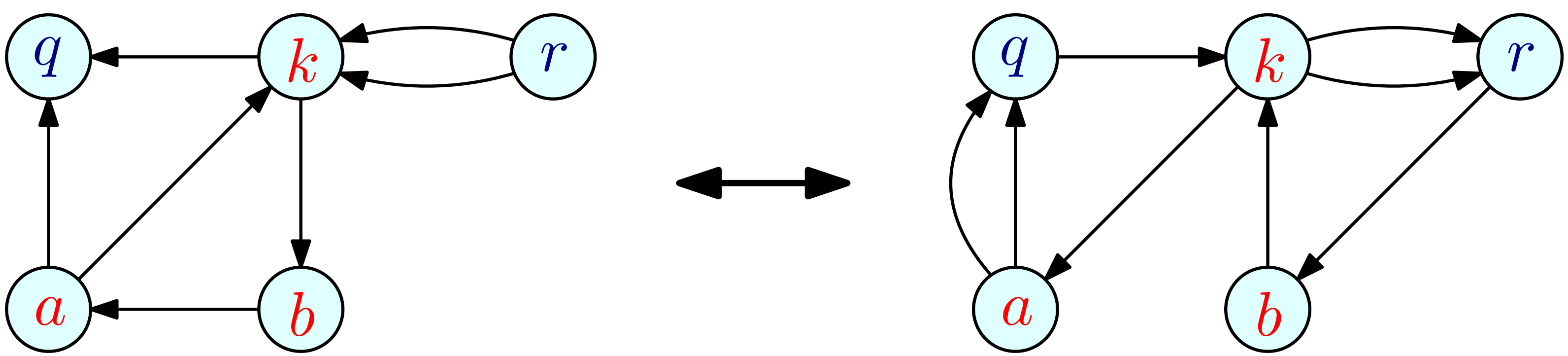
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge“* $i \to j$ (unless both i and j are frozen)
  - **Step II:** Reverse all arcs incident to k
  - **Step III:** Eliminate 2-cycles



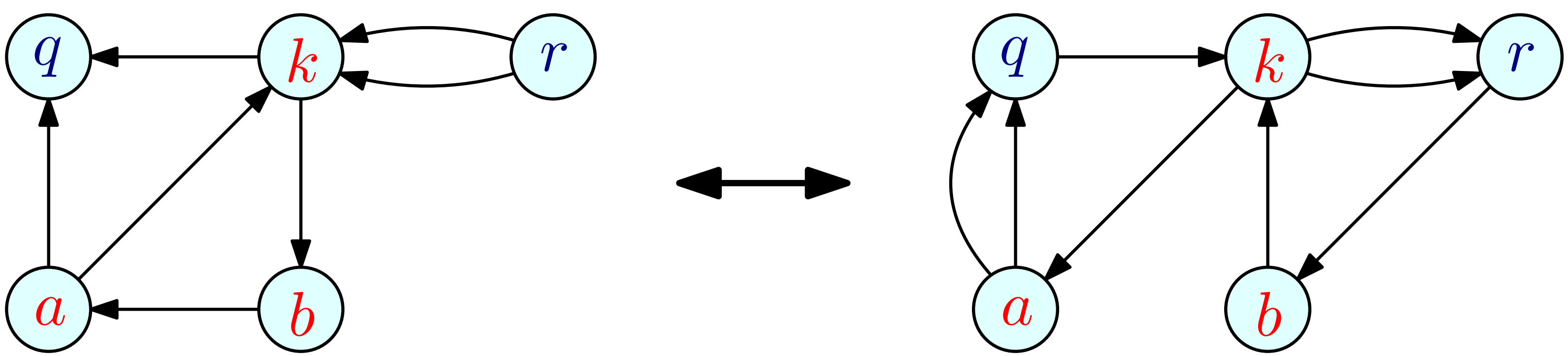
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge“* $i \to j$ (unless both i and j are frozen)
  - **Step II:** Reverse all arcs incident to k
  - **Step III:** Eliminate 2-cycles



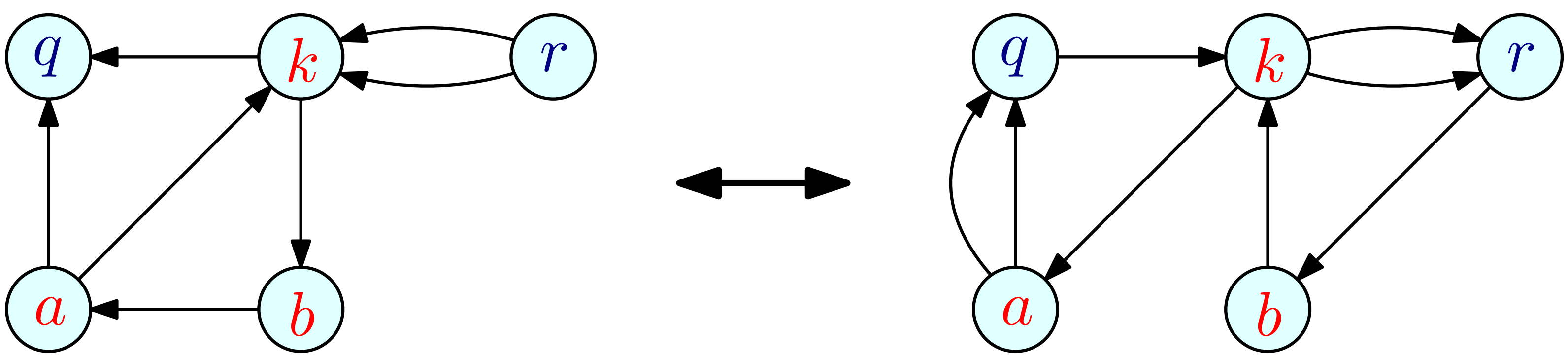
Quiver mutations

- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge“* $i \to j$ (unless both i and j are frozen)
  - **Step II:** Reverse all arcs incident to k
  - **Step III:** Eliminate 2-cycles
- **Properties:**
  - $\mu_k \circ \mu_k = \text{Id}$
  - For u, v non-adjacent: $\mu_u \circ \mu_v = \mu_v \circ \mu_u$



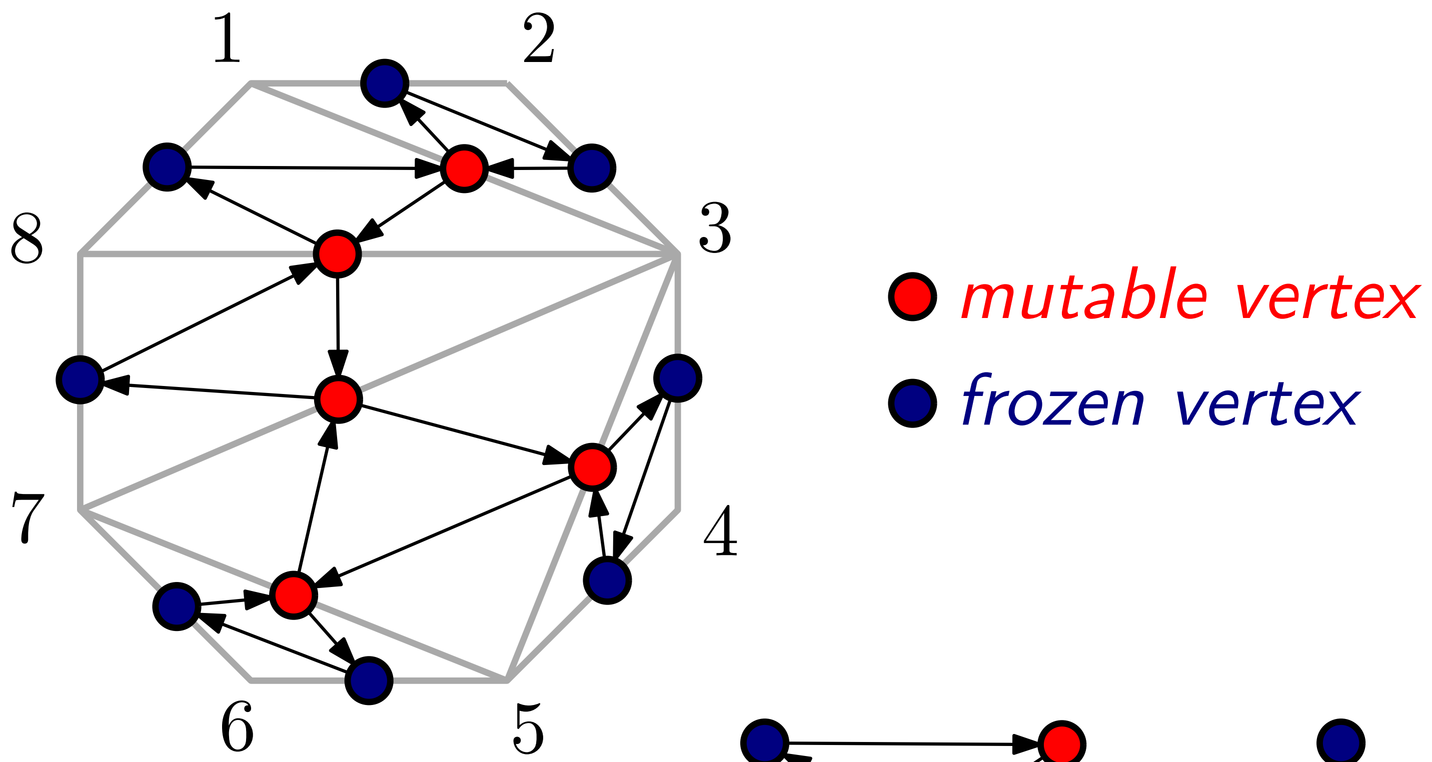
Quiver mutations

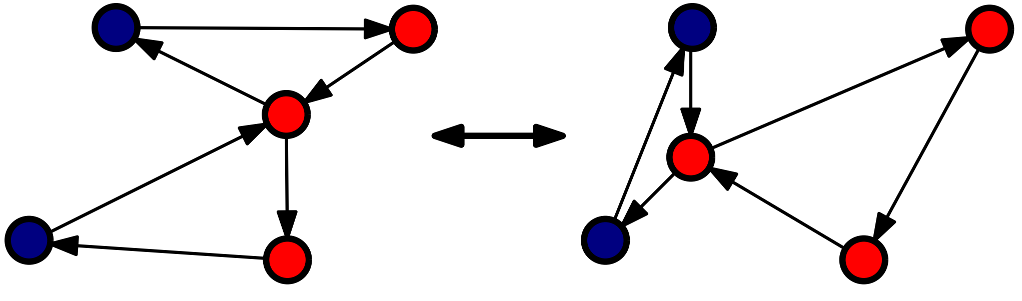
- **Quiver mutation** μ_k for mutable vertex k:
  - **Step I:** For all 2-paths $i \to k \to j$, add *„transitive edge“* $i \to j$ (unless both i and j are frozen)
  - **Step II:** Reverse all arcs incident to k
  - **Step III:** Eliminate 2-cycles
- **Properties:**
  - $\mu_k \circ \mu_k =$ Id
  - For u, v non-adjacent: $\mu_u \circ \mu_v = \mu_v \circ \mu_u$
- **Surprise:** Previously shown flips can be seen as quiver mutations!



Quiver mutations

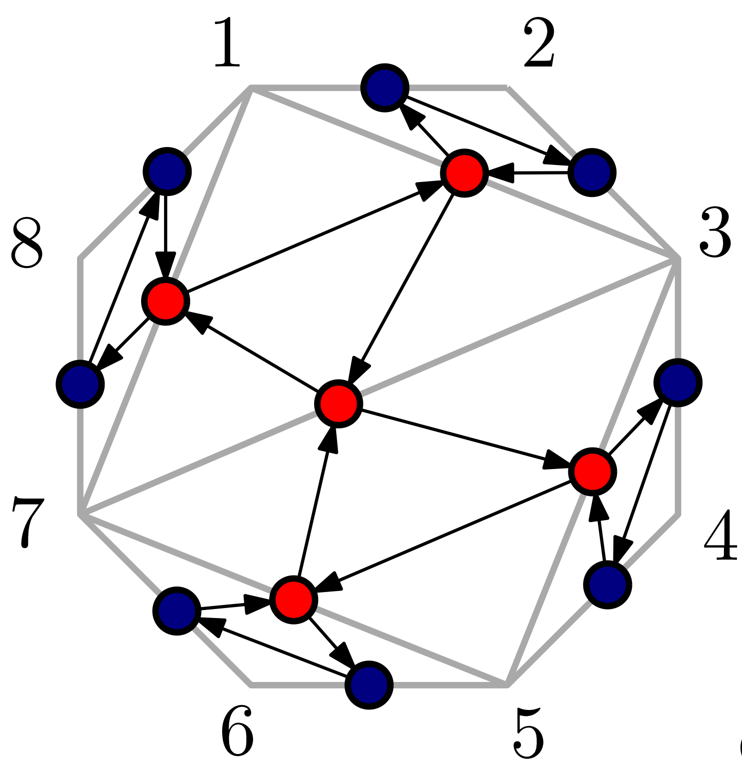
1. Triangulations



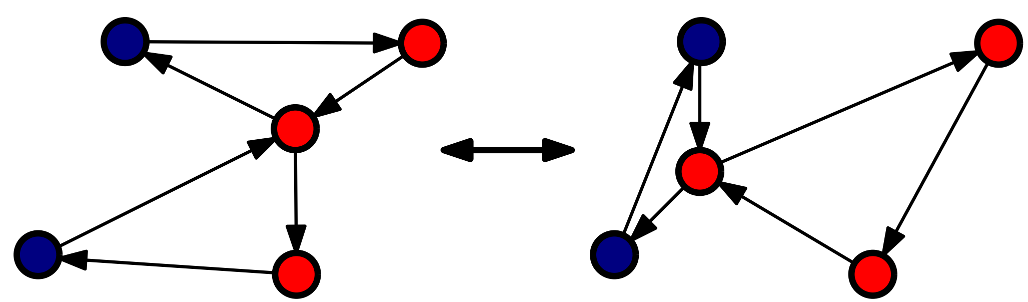


Quiver mutations

1. Triangulations



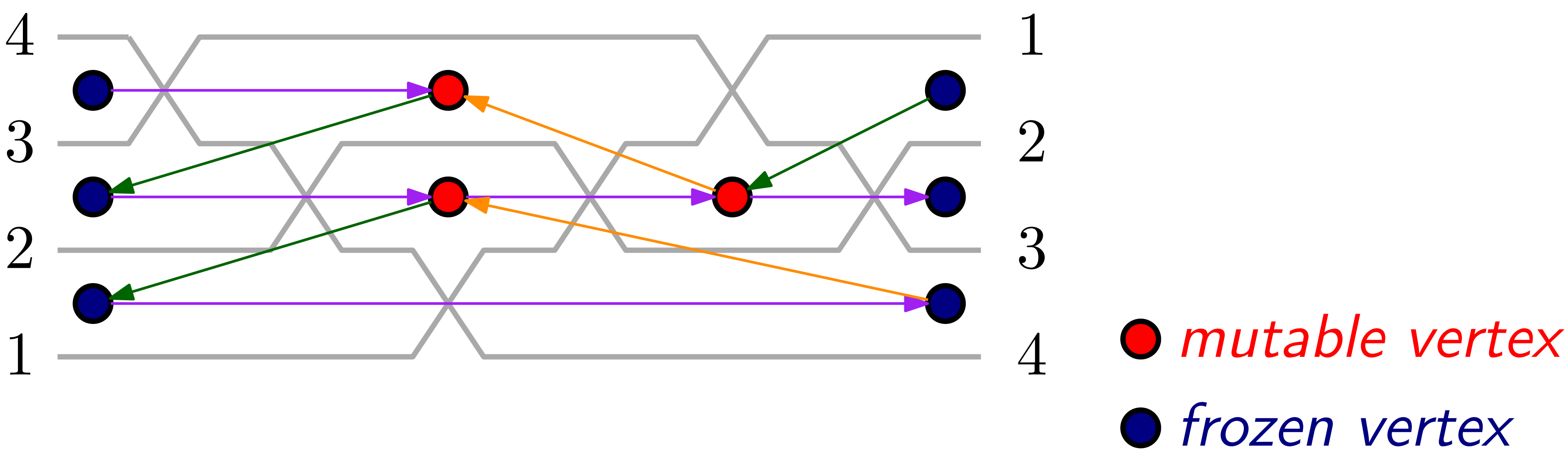
- *mutable vertex*
- *frozen vertex*



Quiver mutations

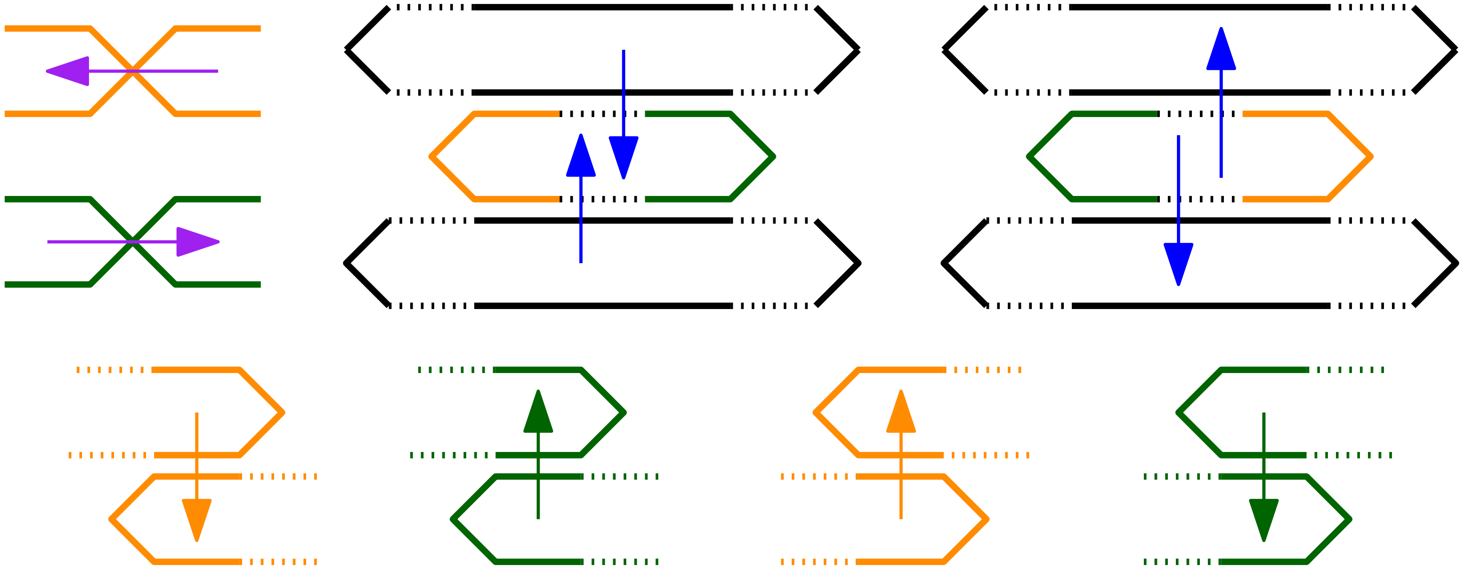
2. Wiring diagrams

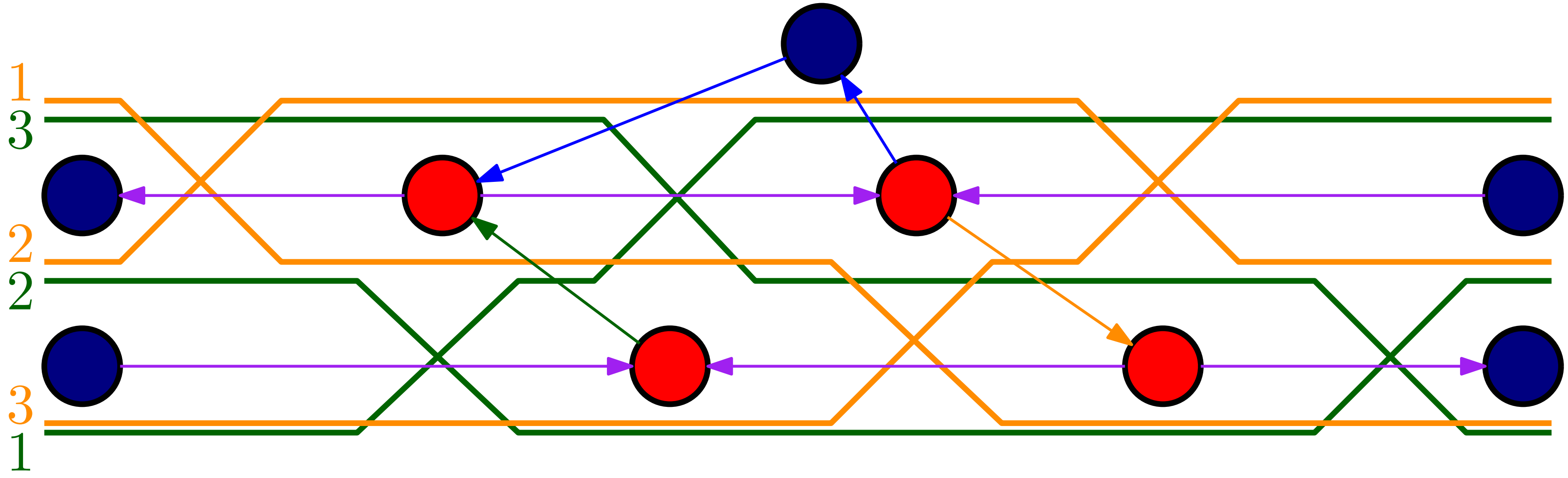
- Bounded cells *mutable*, unbounded cells *frozen*
- Directed edge $u \to v$ if...
  - left end of u equals right end of v, or
  - left end of u above v and right end of v below u, or
  - left end of u below v and right end of v above u.



Quiver mutations

3. Double wiring diagrams





Mutation equivalence

Definition:

- Quivers Q, Q' are *mutation equivalent* ($Q \sim Q'$), if a sequence of mutations transforms Q into a quiver isomorphic to Q'.
- Quiver Q is of *finite mutation type*, if its equivalence class $[Q]$ is finite.

Mutation equivalence

Definition:

- Quivers Q, Q' are *mutation equivalent* ($Q \sim Q'$), if a sequence of mutations transforms Q into a quiver isomorphic to Q'.
- Quiver Q is of *finite mutation type*, if its equivalence class $[Q]$ is finite.

Theorem (Caldero & Keller, 2006):
If Q, Q' are acylic quivers and $Q \sim Q'$, then Q can be transformed into a quiver isomorphic to Q' by only mutations on sources and sinks.

Mutation equivalence

Definition:

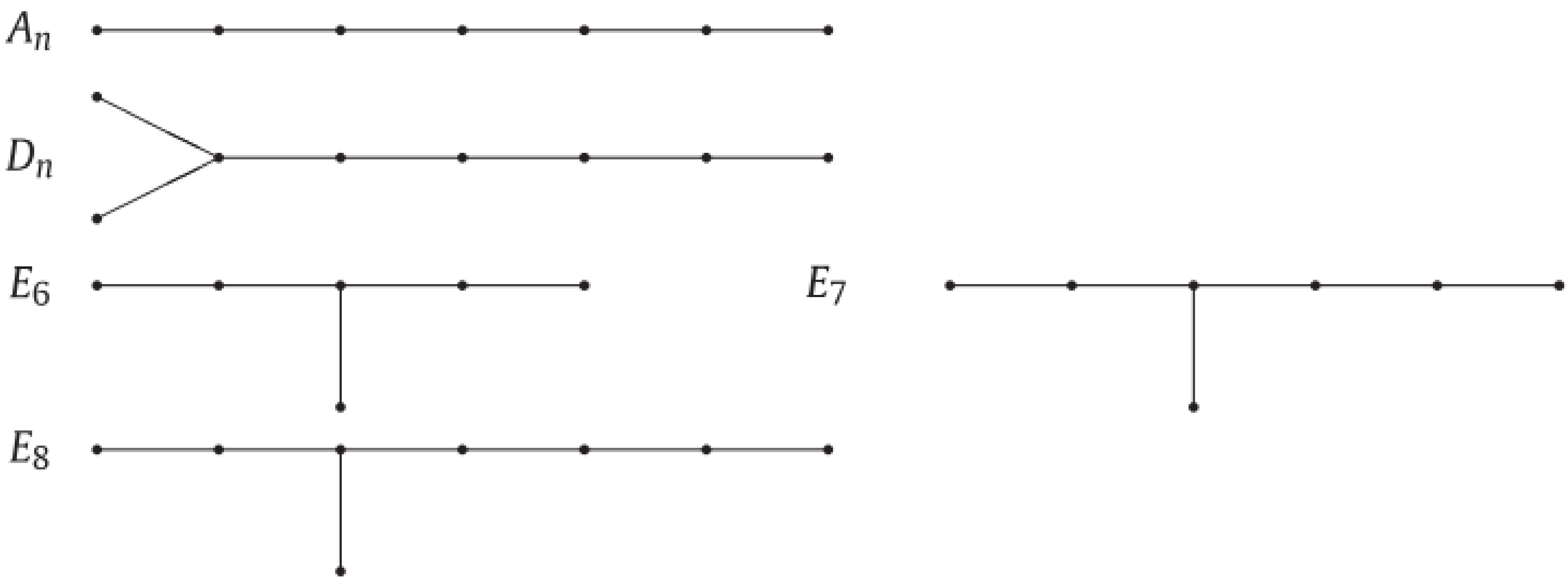
- Quivers Q, Q' are *mutation equivalent* ($Q \sim Q'$), if a sequence of mutations transforms Q into a quiver isomorphic to Q'.
- Quiver Q is of *finite mutation type*, if its equivalence class $[Q]$ is finite.

Theorem (Caldero & Keller, 2006):
If Q, Q' are acylic quivers and $Q \sim Q'$, then Q can be transformed into a quiver isomorphic to Q' by only mutations on sources and sinks.

Corollary: Equivalent acyclic quivers have the same underlying undirected graph.

Mutation equivalence

Finite type classification (Fomin, Zevelinsky, 2002):
Classification of finite type quivers in terms of *Dynkin diagrams*



Labeled seeds and cluster algebras

- Encode Q with its *exchange matrix* $B(Q) = [b_{ij}]$, where

$$b_{ij} = \begin{cases} l & \text{if there are } l \text{ edges from } i \text{ to } j \\ -l & \text{if there are } l \text{ edges from } j \text{ to } i \\ 0 & \text{otherwise} \end{cases}$$

Labeled seeds and cluster algebras

- Encode Q with its *exchange matrix* $B(Q) = [b_{ij}]$, where

$$b_{ij} = \begin{cases} l & \text{if there are } l \text{ edges from } i \text{ to } j \\ -l & \text{if there are } l \text{ edges from } j \text{ to } i \\ 0 & \text{otherwise} \end{cases}$$

- If k is mutable, then $B'(\mu_k(Q)) = [b'_{ij}]$, where

$$b'_{ij} = \begin{cases} -b_{ij} & \text{if } i = k \text{ or } j = k; \\ b_{ij} + b_{ik}b_{kj} & \text{if } b_{ik} > 0 \text{ and } b_{kj} > 0; \\ b_{ij} - b_{ik}b_{kj} & \text{if } b_{ik} < 0 \text{ and } b_{kj} < 0 \\ b_{ij} & \text{otherwise} \end{cases}$$

Labeled seeds and cluster algebras

- Encode Q with its *exchange matrix* $B(Q) = [b_{ij}]$, where

$$b_{ij} = \begin{cases} l & \text{if there are } l \text{ edges from } i \text{ to } j \\ -l & \text{if there are } l \text{ edges from } j \text{ to } i \\ 0 & \text{otherwise} \end{cases}$$

- If k is mutable, then $B'(\mu_k(Q)) = [b'_{ij}]$, where

$$b'_{ij} = \begin{cases} -b_{ij} & \text{if } i = k \text{ or } j = k; \\ b_{ij} + b_{ik}b_{kj} & \text{if } b_{ik} > 0 \text{ and } b_{kj} > 0; \\ b_{ij} - b_{ik}b_{kj} & \text{if } b_{ik} < 0 \text{ and } b_{kj} < 0 \\ b_{ij} & \text{otherwise} \end{cases}$$

- **Property:** $\det(B(\mu_k(Q)) = \det(B(Q))$

Labeled seeds and cluster algebras

- Let $m, n \in \mathbb{N}, m \geq n$
- Let $\mathscr{F} = \mathbb{C}(x_1, ..., x_m)$ field of rational functions over $\mathbb{C}$ in m variables

Labeled seeds and cluster algebras

- Let $m, n \in \mathbb{N}, m \geq n$
- Let $\mathscr{F} = \mathbb{C}(x_1, ..., x_m)$ field of rational functions over $\mathbb{C}$ in m variables

Definition: A *labeled seed in* $\mathscr{F}$ is a pair (x, B), where

- $B = B(Q)$ for some quiver Q in which
  - vertices $1, ..., n$ are mutable
  - vertices $n+1, ..., m$ are frozen
- $x = (x_1, ..., x_m)$, alg. indep. generators of $\mathscr{F}$

Labeled seeds and cluster algebras

- Let $m, n \in \mathbb{N}, m \geq n$
- Let $\mathscr{F} = \mathbb{C}(x_1, ..., x_m)$ field of rational functions over $\mathbb{C}$ in m variables

Definition: A *labeled seed in* $\mathscr{F}$ is a pair (x, B), where

- $B = B(Q)$ for some quiver Q in which
  - vertices $1, ..., n$ are mutable
  - vertices $n+1, ..., m$ are frozen
- $x = (x_1, ..., x_m)$, alg. indep. generators of $\mathscr{F}$

Terminology:

- *Extended cluster:* $x = (x_1, \cdots, x_m)$
- *Cluster variables:* $x_1, \cdots, x_n$
- *Frozen variables:* $x_{n+1}, \cdots, x_m$

Labeled seeds and cluster algebras

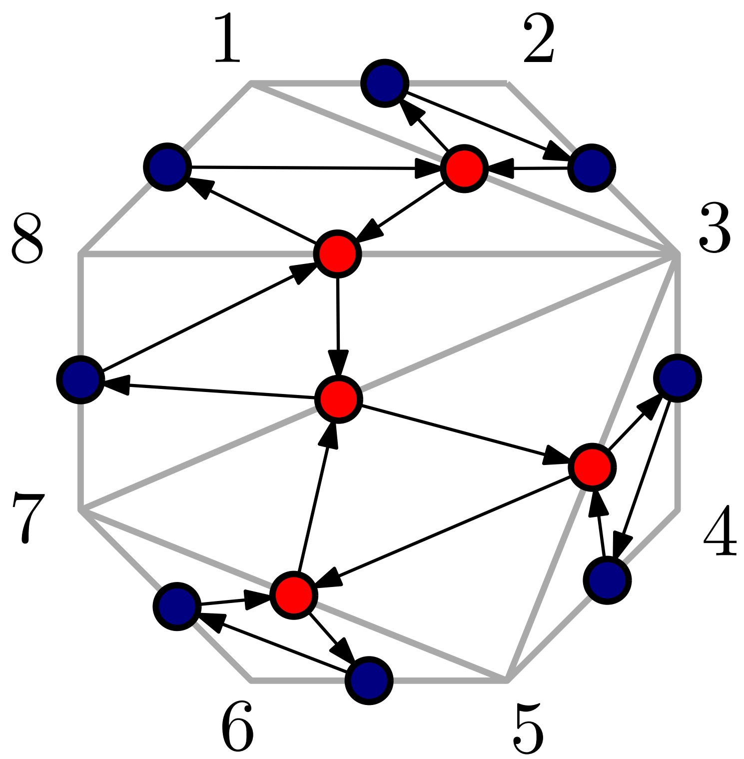
Definition: Let (x, B) be a labeled seed, k mutable. The *seed mutation* μ_k transforms (x, B) into the labeled seed (x', B'), where

- $B' = \mu_k(B)$ (Quiver mutation on k)
- $x'_i = x_i$ for all $i \neq k$
- x'_k determined by *exchange relation*

$$x_k x'_k = \prod_{b_{ik}>0} x_i^{b_{ik}} + \prod_{b_{ik}<0} x_i^{-b_{ik}}$$

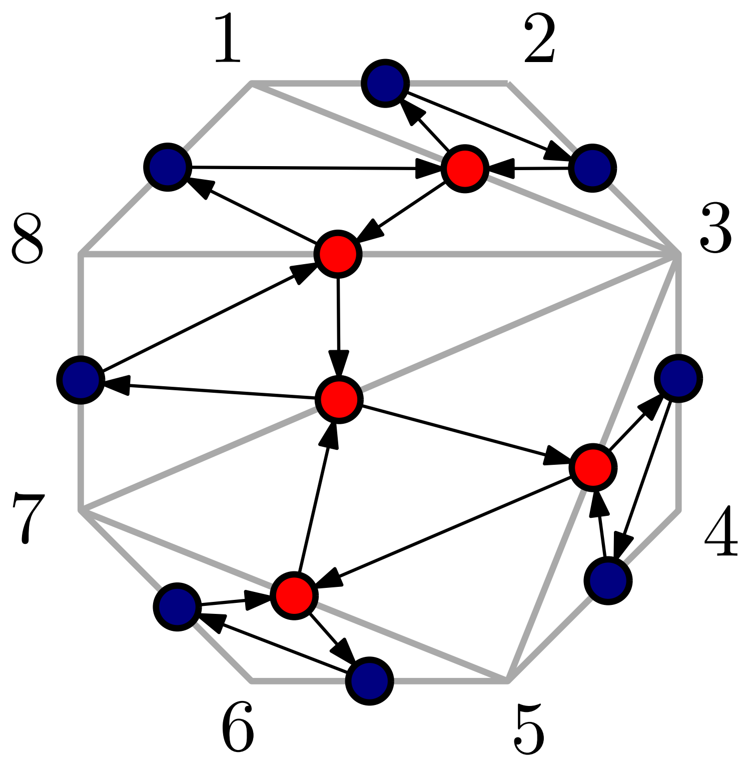
Labeled seeds and cluster algebras

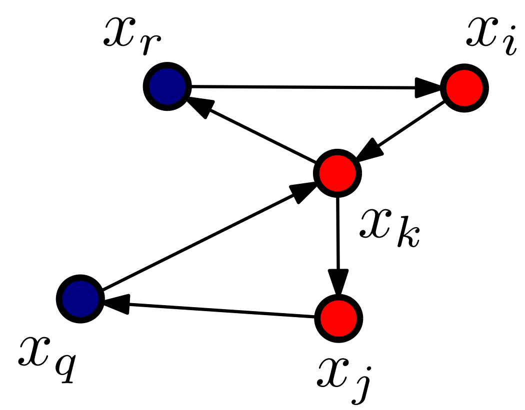
Example:



Labeled seeds and cluster algebras

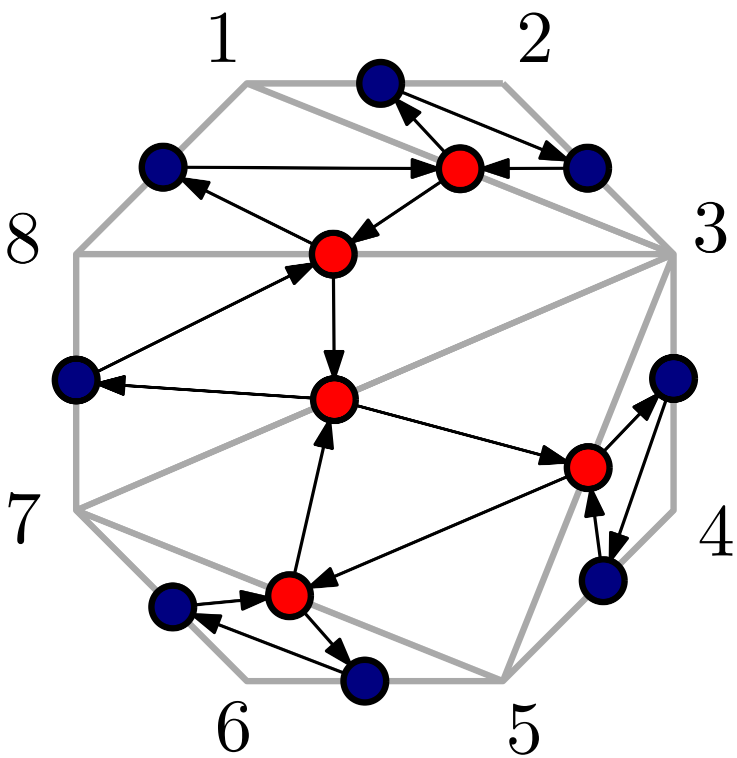
Example:

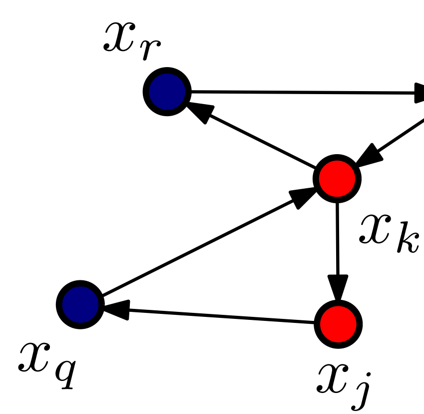




Labeled seeds and cluster algebras

Example:





x

$x_r = P_{18}$

$x_i = P_{13}$

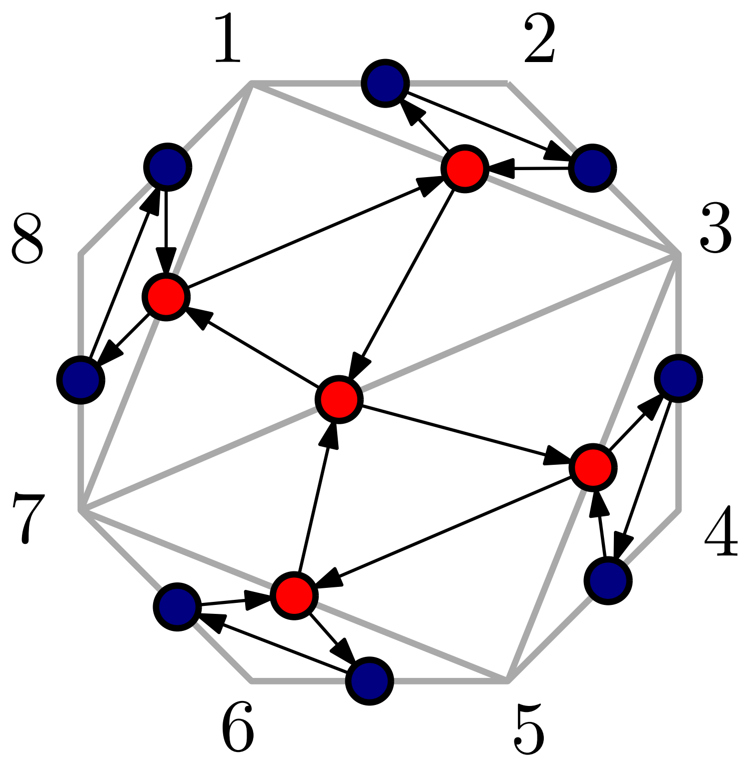
$x_j = P_{37}$

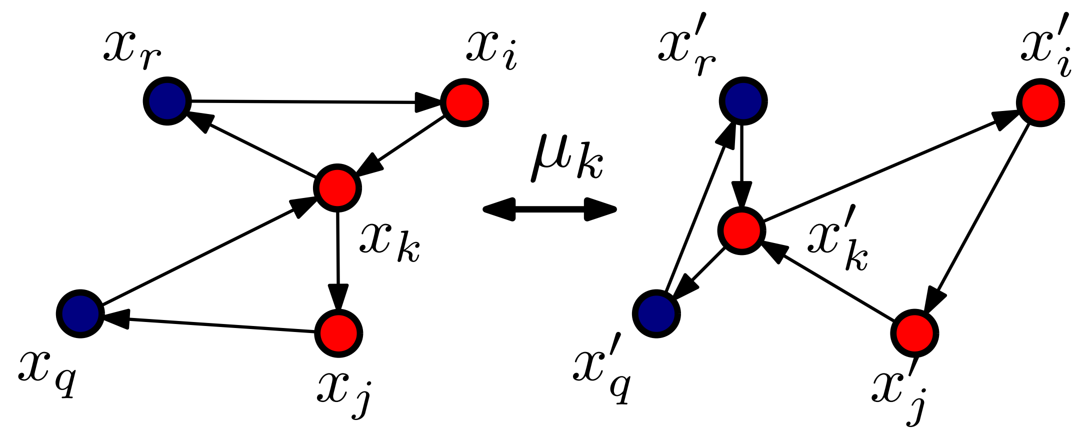
$x_q = P_{78}$

$x_k = P_{38}$

Labeled seeds and cluster algebras

Example:





x

$x_r = P_{18}$

$x_i = P_{13}$

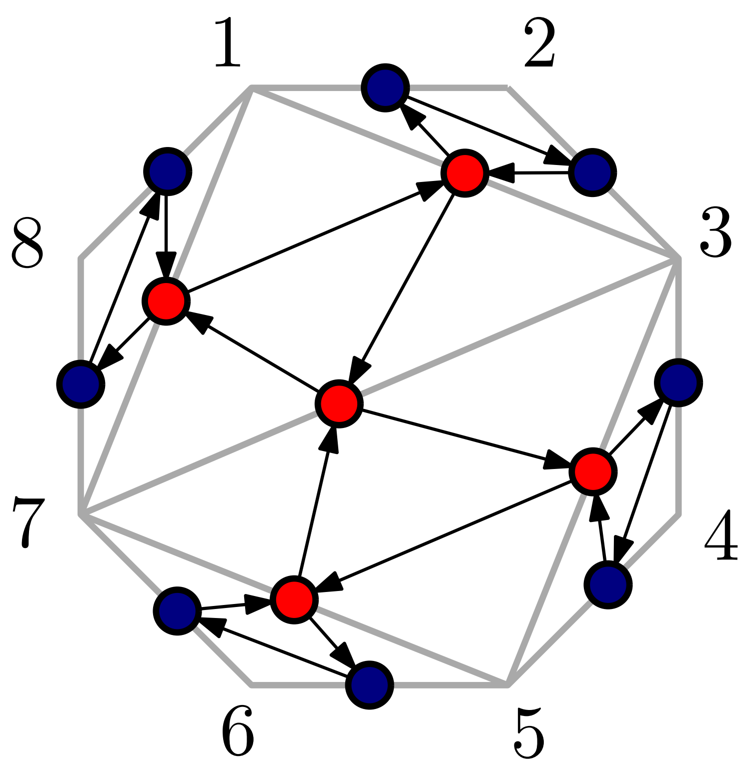
$x_j = P_{37}$

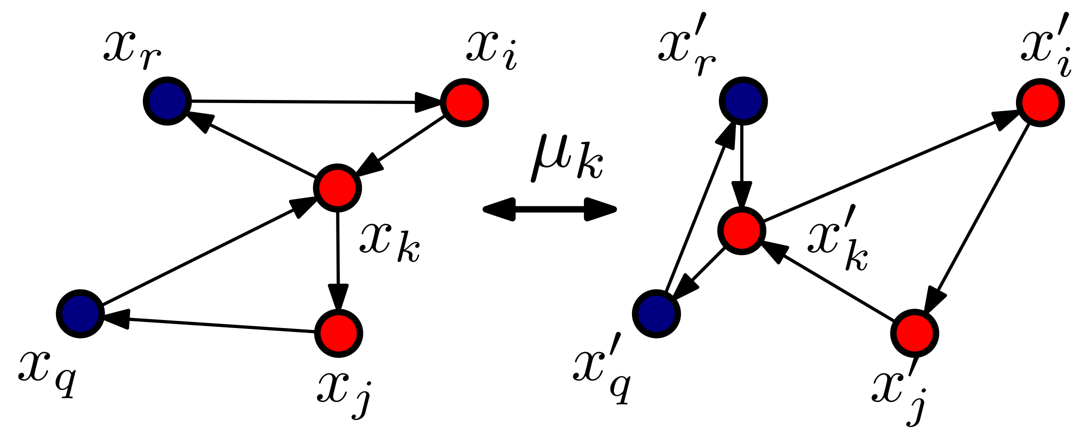
$x_q = P_{78}$

$x_k = P_{38}$

Labeled seeds and cluster algebras

Example:





x	$x' = \mu_k(x)$
$x_r = P_{18}$	$x'_r = P_{18}$
$x_i = P_{13}$	$x'_i = P_{13}$
$x_j = P_{37}$	$x'_j = P_{37}$
$x_q = P_{78}$	$x'_q = P_{78}$
$x_k = P_{38}$	$x'_k = \dfrac{x_r x_j + x_i x_q}{x_k}$

$$= \frac{P_{18}P_{37} + P_{13}P_{78}}{P_{38}} = P_{17}$$

Labeled seeds and cluster algebras

- Let (x, B) labeled seed (*initial seed*)

Labeled seeds and cluster algebras

- Let (x, B) labeled seed (*initial seed*)
- Let $\mathscr{R} := \mathbb{C}[x_{n+1}, \cdots, x_m]$ polynomial ring generated by the frozen variables.

Labeled seeds and cluster algebras

- Let (x, B) labeled seed (*initial seed*)
- Let $\mathscr{R} := \mathbb{C}[x_{n+1}, \cdots, x_m]$ polynomial ring generated by the frozen variables.
- Let
$$x(t) = (x_1(t), \cdots, x_m(t))$$
be the extended cluster after performing the sequence of mutations t on (x, B).

Labeled seeds and cluster algebras

- Let (x, B) labeled seed (*initial seed*)
- Let $\mathscr{R} := \mathbb{C}[x_{n+1}, \cdots, x_m]$ polynomial ring generated by the frozen variables.
- Let
$$x(t) = (x_1(t), \cdots, x_m(t))$$
be the extended cluster after performing the sequence of mutations t on (x, B).
- Let
$$\mathscr{X} = \bigcup_t \{x_1(t), \cdots, x_m(t)\}$$
the set of variables appearing when doing all possible mutations.

Labeled seeds and cluster algebras

- Let (x, B) labeled seed (*initial seed*)
- Let $\mathscr{R} := \mathbb{C}[x_{n+1}, \cdots, x_m]$ polynomial ring generated by the frozen variables.
- Let
$$x(t) = (x_1(t), \cdots, x_m(t))$$
be the extended cluster after performing the sequence of mutations t on (x, B).
- Let
$$\mathscr{X} = \bigcup_t \{x_1(t), \cdots, x_m(t)\}$$
the set of variables appearing when doing all possible mutations.

Definition: The *cluster algebra* $\mathscr{A} := \mathscr{R}[\mathscr{X}]$ is the $\mathscr{R}$-subalgebra of $\mathscr{F}$ generated by all cluster variables.

Labeled seeds and cluster algebras

- Initial seeds $(x, B(Q)), (x', B(Q))$ (same quiver) yield isomorphic clusteralgebras $\mathscr{A} \cong \mathscr{A}'$.

Labeled seeds and cluster algebras

- Initial seeds $(x, B(Q)), (x', B(Q))$ (same quiver) yield isomorphic clusteralgebras $\mathscr{A} \cong \mathscr{A}'$.
- Therefore, $\mathscr{A}$ depends only on the equivalence class $[Q]$.

Labeled seeds and cluster algebras

- Initial seeds $(x, B(Q)), (x', B(Q))$ (same quiver) yield isomorphic clusteralgebras $\mathscr{A} \cong \mathscr{A}'$.
- Therefore, $\mathscr{A}$ depends only on the equivalence class $[Q]$.

Previous examples:

- Cluster algebra from triangulations equals *Plücker ring* $R_{2,m}$ (ring generated by $\{P_{ij}\}$).

Labeled seeds and cluster algebras

- Initial seeds $(x, B(Q)), (x', B(Q))$ (same quiver) yield isomorphic clusteralgebras $\mathscr{A} \cong \mathscr{A}'$.
- Therefore, $\mathscr{A}$ depends only on the equivalence class $[Q]$.

Previous examples:

- Cluster algebra from triangulations equals *Plücker ring* $R_{2,m}$ (ring generated by $\{P_{ij}\}$).
- Cluster algebra from wiring diagrams equals *ring of invariants* $\mathbb{C}[\mathsf{SL}_n]^U$.

Labeled seeds and cluster algebras

- Initial seeds $(x, B(Q)), (x', B(Q))$ (same quiver) yield isomorphic clusteralgebras $\mathscr{A} \cong \mathscr{A}'$.
- Therefore, $\mathscr{A}$ depends only on the equivalence class $[Q]$.

Previous examples:

- Cluster algebra from triangulations equals *Plücker ring* $R_{2,m}$ (ring generated by $\{P_{ij}\}$).
- Cluster algebra from wiring diagrams equals *ring of invariants* $\mathbb{C}[\mathsf{SL}_n]^U$.
- Cluster algebra from double wiring diagrams equals polynomial ring $\mathbb{C}(x_{11}, x_{12}, \cdots, x_{nn})$.

Labeled seeds and cluster algebras

Theorem („Laurent phenomenon"):
Let $\mathscr{A}$ be a cluster algebra. All cluster variables in $\mathscr{A}$ can be expressed as a Laurent polynomial with integer coefficients in the elements of any extended cluster $x = (x_1, \cdots, x_m)$. In particular:

$$\mathscr{A} \subseteq \mathbb{C}(x_1, x_1^{-1}, \cdots, x_n, x_n^{-1})$$

Questions?