UNIVERSITÄT DES SAARLANDES

MAX PLANCK INSTITUTE FOR INFORMATICS

Stable Matching

Mechanism Design Without Money

Kurt Mehlhorn, **Javier Cembrano**, Golnoosh Shahkarami

May 13, 2025

Several proofs and examples of this lecture are taken from Thomas Kesselheim's lecture notes

Matching Markets and Stability

- job applicants A, companies X

Matching Markets and Stability

- job applicants A, companies X
- applicant $a \in A$ has strict preferences $\succ_a \in \mathscr{L}(X)$

$\mathscr{L}(X)$: set of binary relations $\succ$ satisfying

- either $x \succ y$ or $y \succ x$ for every $x, y \in X$ with $x \neq y$
- $x \succ z$ whenever $x \succ y$ and $y \succ z$

Matching Markets and Stability

- job applicants A, companies X
- applicant $a \in A$ has strict preferences $\succ_a \in \mathscr{L}(X)$ and company $x \in X$ has strict preferences $\succ_x \in \mathscr{L}(A)$

$\mathscr{L}(X)$: set of binary relations $\succ$ satisfying

- either $x \succ y$ or $y \succ x$ for every $x, y \in X$ with $x \neq y$
- $x \succ z$ whenever $x \succ y$ and $y \succ z$

Matching Markets and Stability

- job applicants A, companies X
- applicant $a \in A$ has strict preferences $\succ_a \in \mathscr{L}(X)$ and company $x \in X$ has strict preferences $\succ_x \in \mathscr{L}(A)$

$x \succ_a y \succ_a z$ $\qquad$ $a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\qquad$ $c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\qquad$ $b \succ_z a \succ_z c$

Matching Markets and Stability

- job applicants A, companies X
- applicant $a \in A$ has strict preferences $\succ_a \in \mathscr{L}(X)$
  and company $x \in X$ has strict preferences $\succ_x \in \mathscr{L}(A)$
- **goal:** match applicants and companies

$$x \succ_a y \succ_a z \qquad a \succ_x b \succ_x c$$

$$x \succ_b z \succ_b y \qquad c \succ_y a \succ_y b$$

$$z \succ_c x \succ_c y \qquad b \succ_z a \succ_z c$$

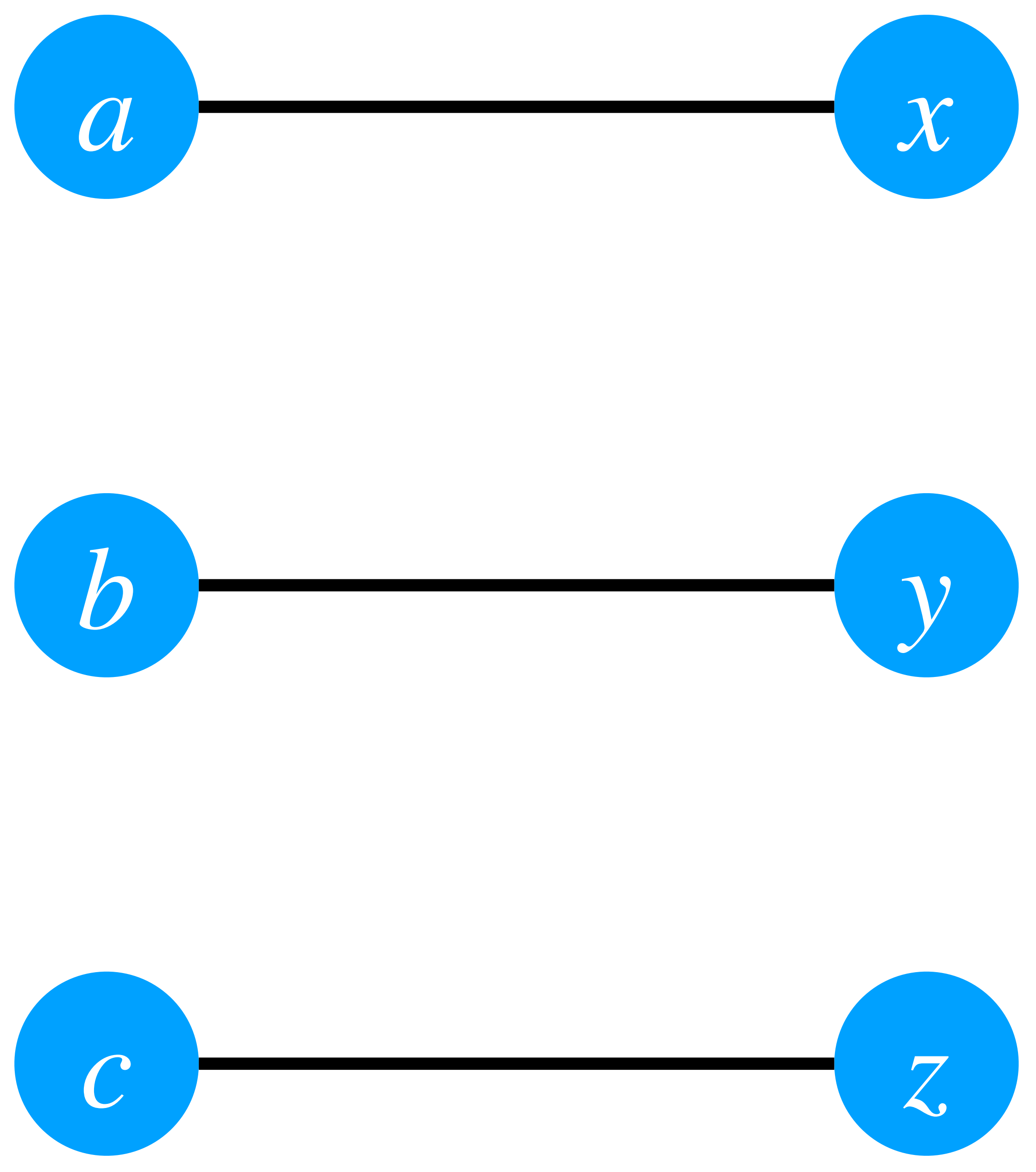
Matching Markets and Stability

- job applicants A, companies X
- applicant $a \in A$ has strict preferences $\succ_a \in \mathscr{L}(X)$ and company $x \in X$ has strict preferences $\succ_x \in \mathscr{L}(A)$
- **goal:** match applicants and companies

$x \succ_a y \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad$ $c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad$ $b \succ_z a \succ_z c$



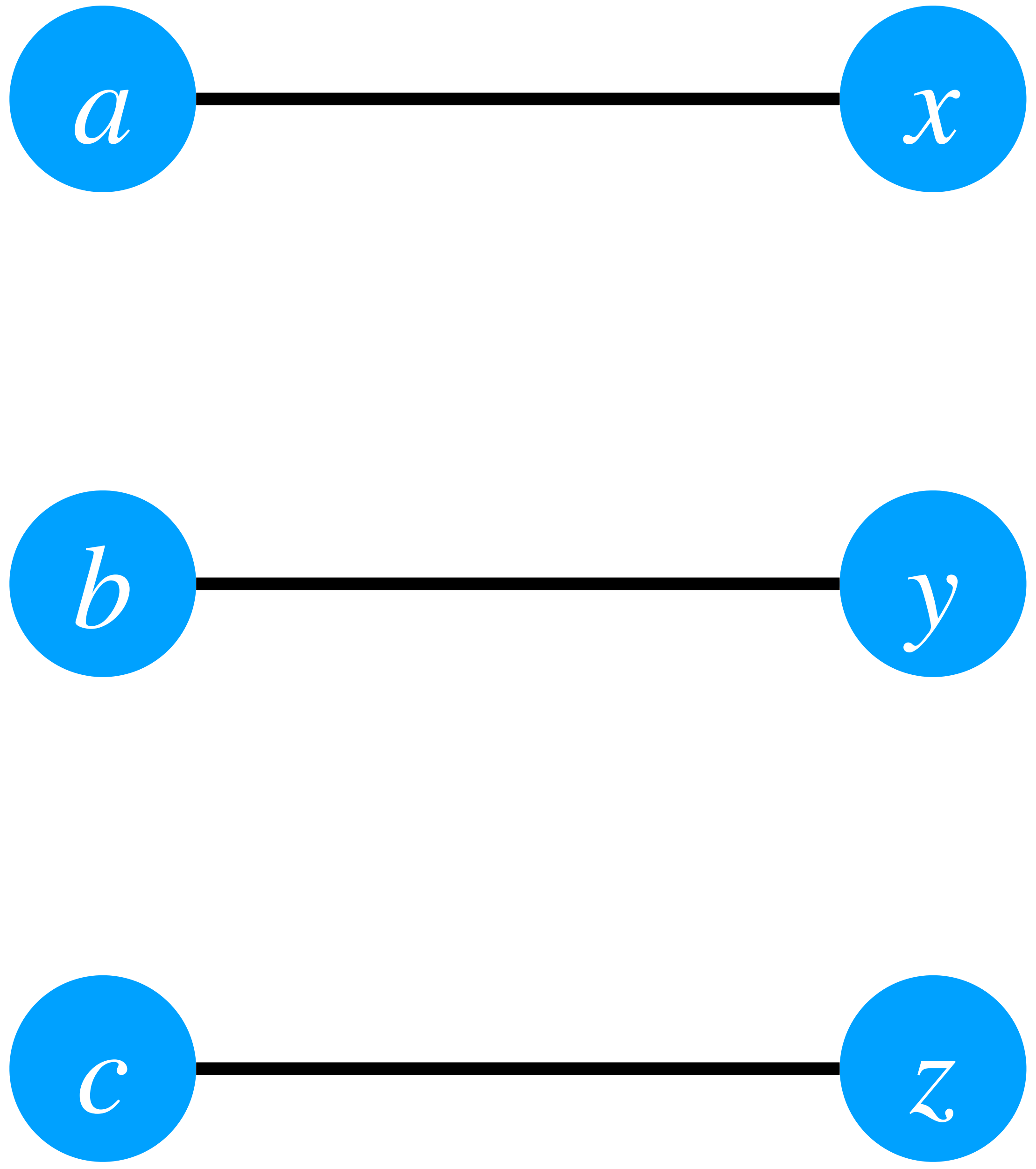
Matching Markets and Stability

- job applicants A, companies X
- applicant $a \in A$ has strict preferences $\succ_a \in \mathscr{L}(X)$ and company $x \in X$ has strict preferences $\succ_x \in \mathscr{L}(A)$
- **goal:** match applicants and companies
- (a, x) is a **blocking pair** if they prefer each other over their current assignment

$x \succ_a y \succ_a z$ $\quad a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad b \succ_z a \succ_z c$



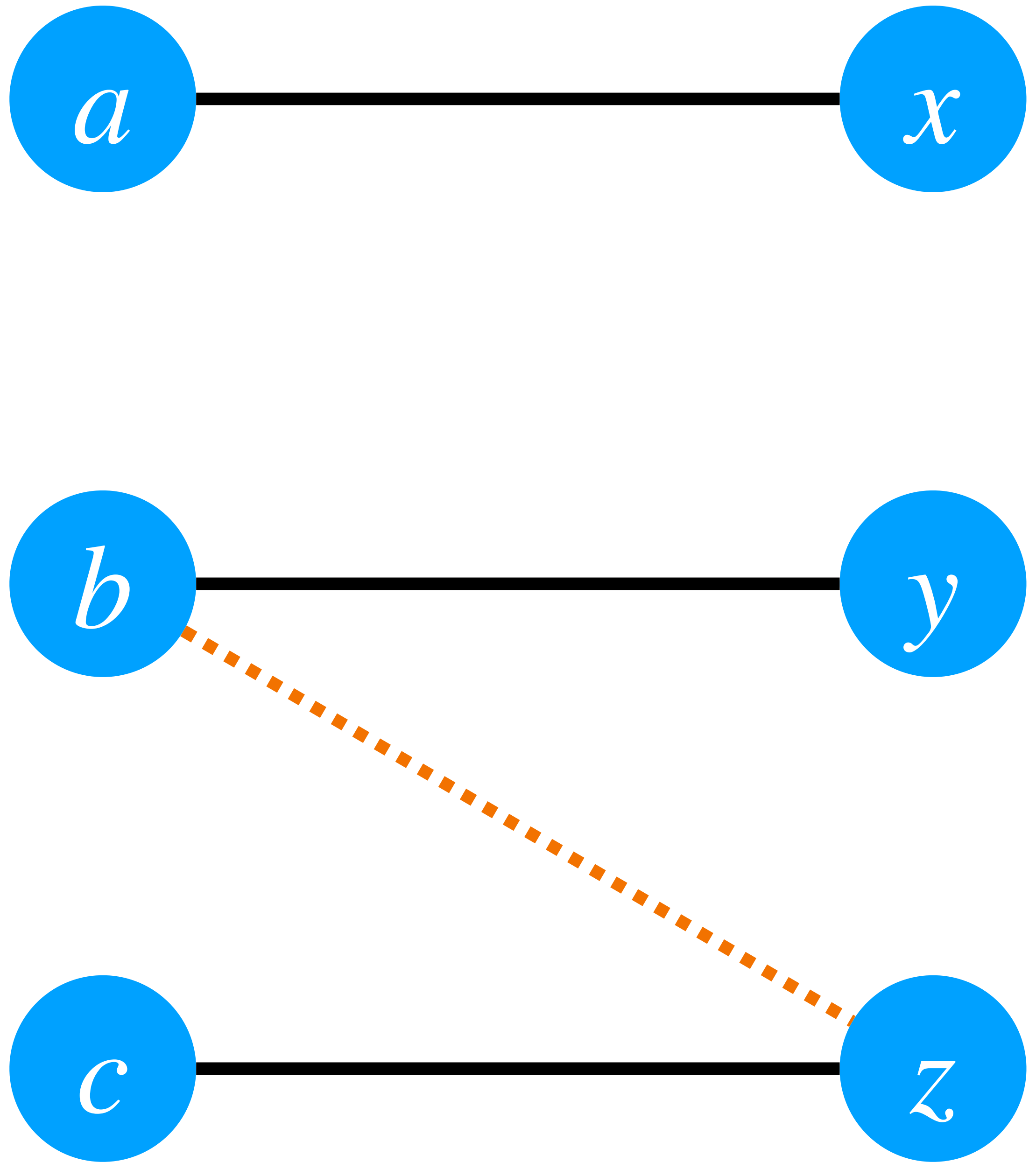
Matching Markets and Stability

- job applicants A, companies X
- applicant $a \in A$ has strict preferences $\succ_a \in \mathscr{L}(X)$ and company $x \in X$ has strict preferences $\succ_x \in \mathscr{L}(A)$
- **goal:** match applicants and companies
- (a, x) is a **blocking pair** if they prefer each other over their current assignment

$x \succ_a y \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad$ $c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad$ $b \succ_z a \succ_z c$



Matching Markets and Stability

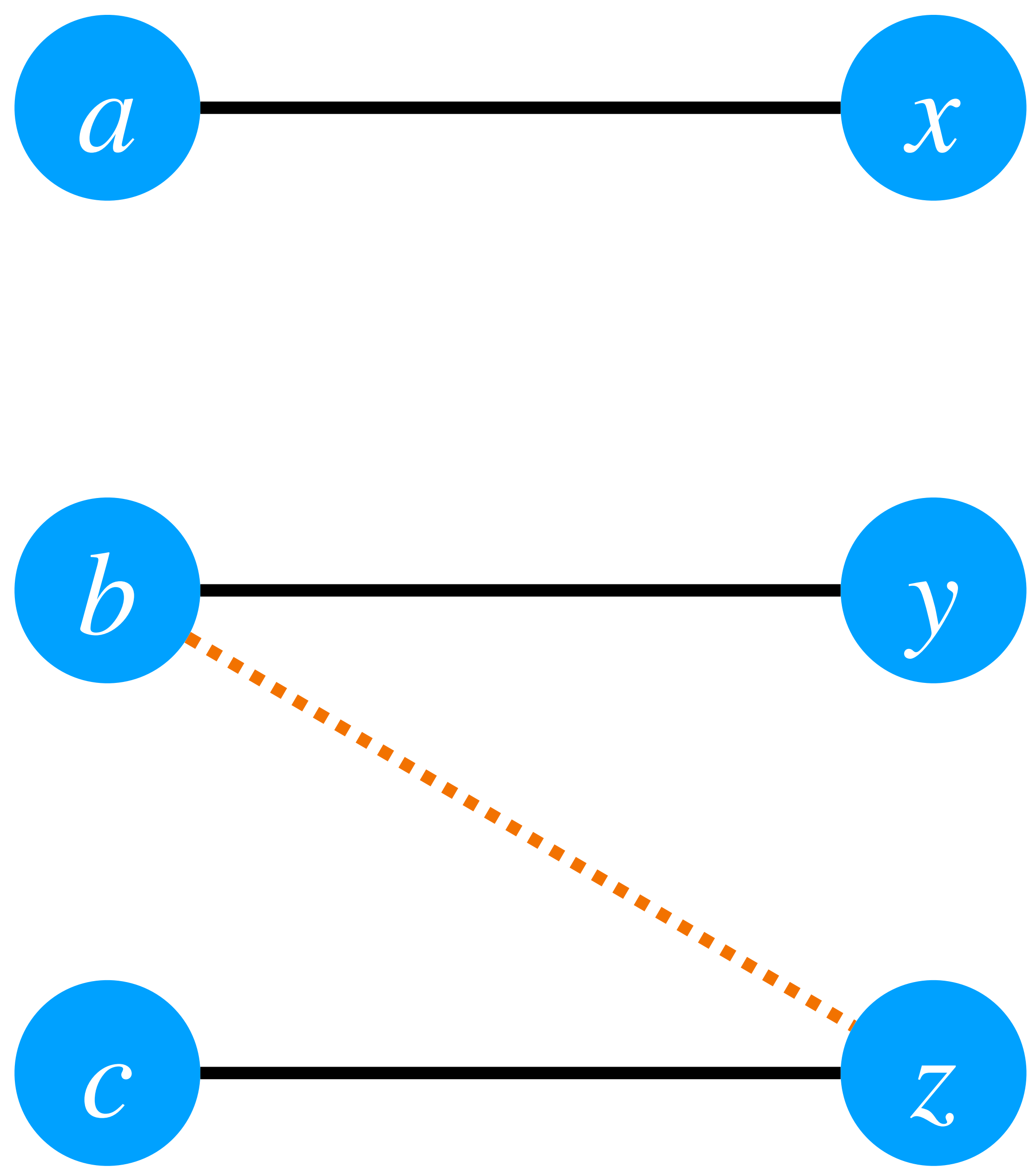
- job applicants A, companies X
- applicant $a \in A$ has strict preferences $\succ_a \in \mathscr{L}(X)$ and company $x \in X$ has strict preferences $\succ_x \in \mathscr{L}(A)$
- **goal:** match applicants and companies
- (a, x) is a **blocking pair** if they prefer each other over their current assignment
- a matching is **stable** if there is no blocking pair

  a.k.a. no justified envy

$x \succ_a y \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad$ $c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad$ $b \succ_z a \succ_z c$



Matching Markets and Stability

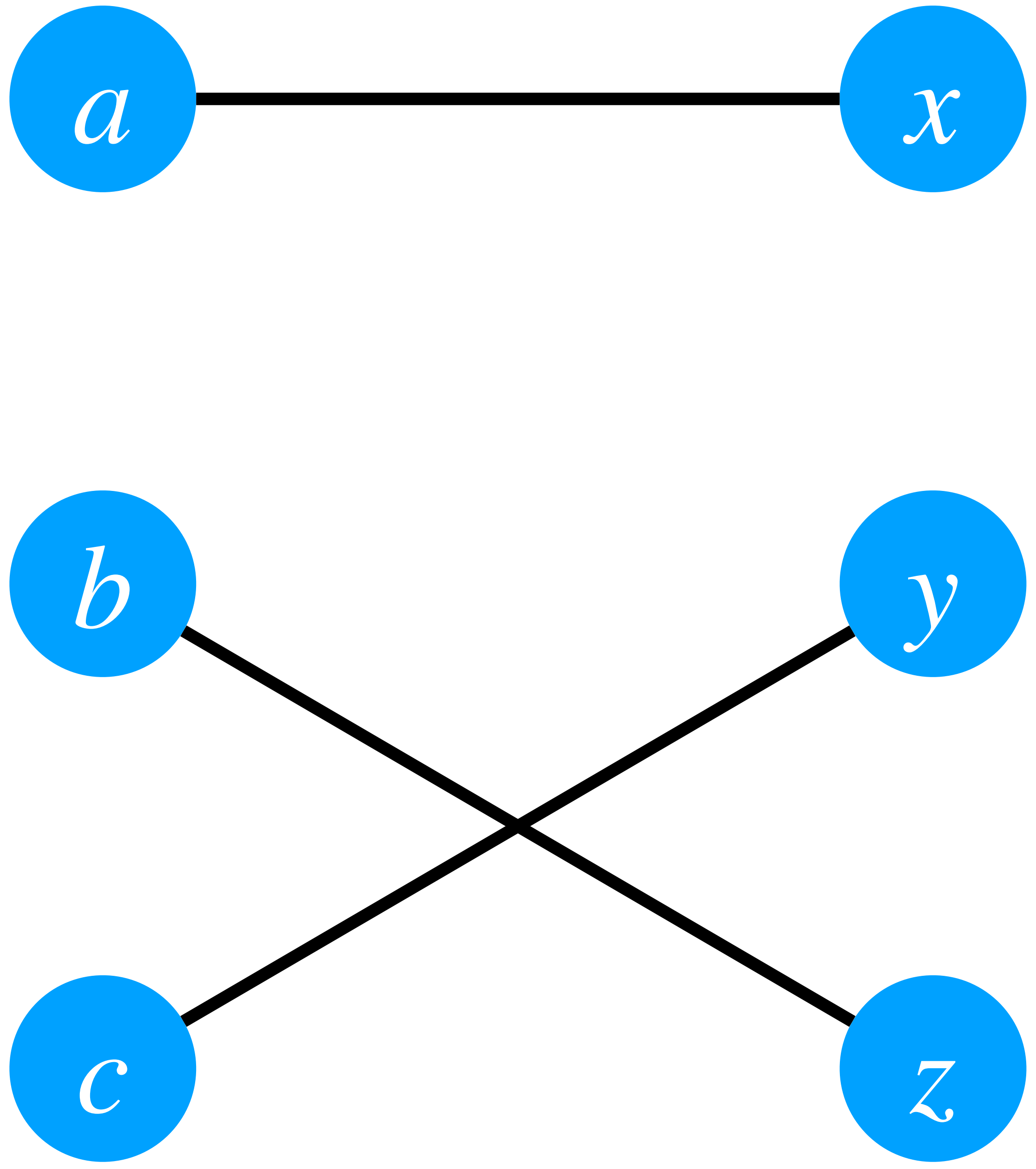
- job applicants A, companies X
- applicant $a \in A$ has strict preferences $\succ_a \in \mathscr{L}(X)$ and company $x \in X$ has strict preferences $\succ_x \in \mathscr{L}(A)$
- **goal:** match applicants and companies
- (a, x) is a **blocking pair** if they prefer each other over their current assignment
- a matching is **stable** if there is no blocking pair

  a.k.a. no justified envy

$x \succ_a y \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad$ $c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad$ $b \succ_z a \succ_z c$



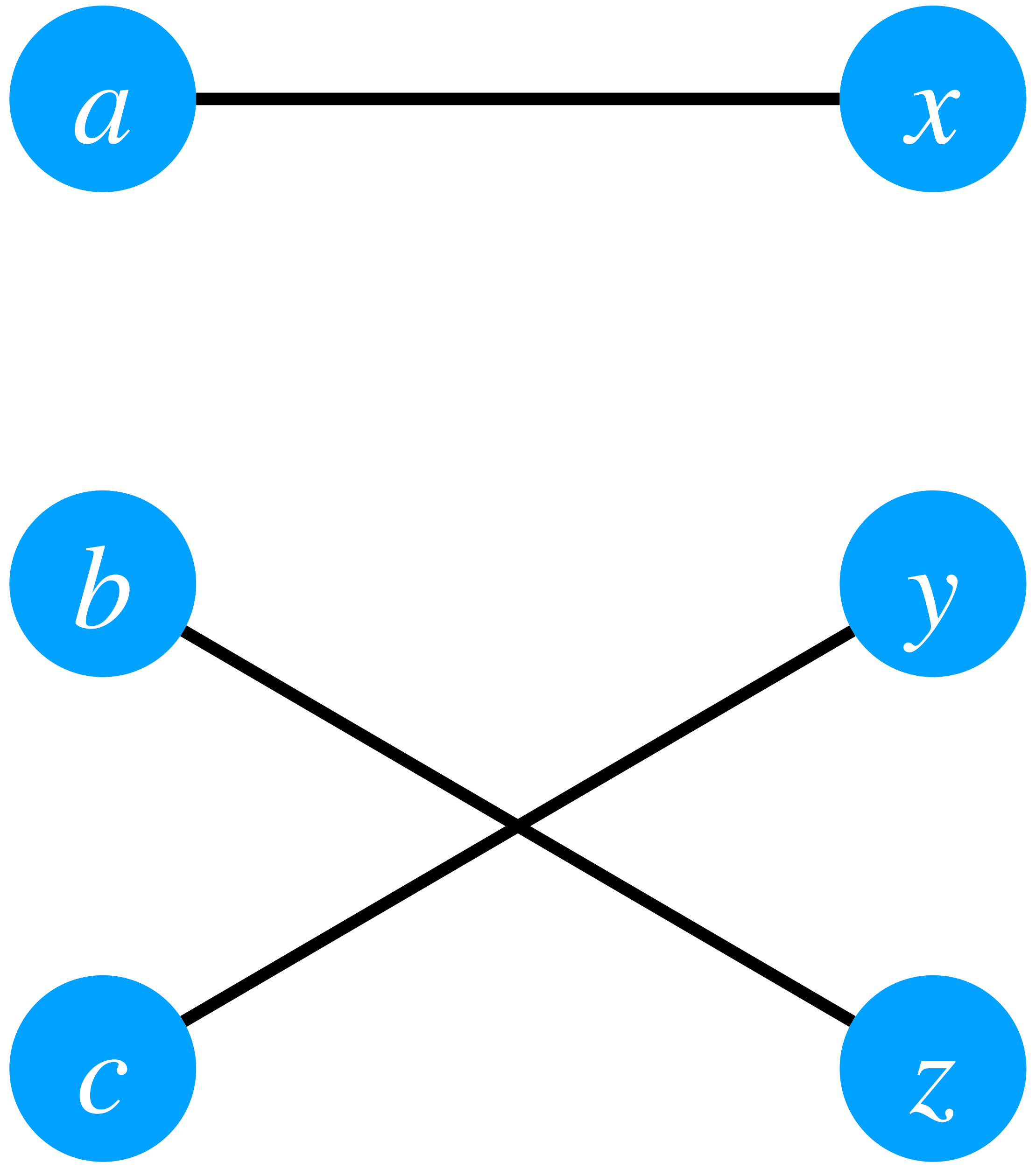
Matching Markets and Stability

- job applicants A, companies X
- applicant $a \in A$ has strict preferences $\succ_a \in \mathscr{L}(X)$ and company $x \in X$ has strict preferences $\succ_x \in \mathscr{L}(A)$
- **goal:** match applicants and companies
- (a, x) is a **blocking pair** if they prefer each other over their current assignment
- a matching is **stable** if there is no blocking pair
  a.k.a. no justified envy
- a mechanism is stable if it produces stable matchings
  analogously for other properties

$x \succ_a y \succ_a z \qquad a \succ_x b \succ_x c$

$x \succ_b z \succ_b y \qquad c \succ_y a \succ_y b$

$z \succ_c x \succ_c y \qquad b \succ_z a \succ_z c$



Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur

  at that point return the provisional matching

Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur

  at that point return the provisional matching

  - each $a \in A$ proposes to their most preferred company that has not yet rejected them

Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur

  at that point return the provisional matching

  - each $a \in A$ proposes to their most preferred company that has not yet rejected them
  - each $x \in X$ provisionally accepts their favorite proposer and rejects the rest

Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

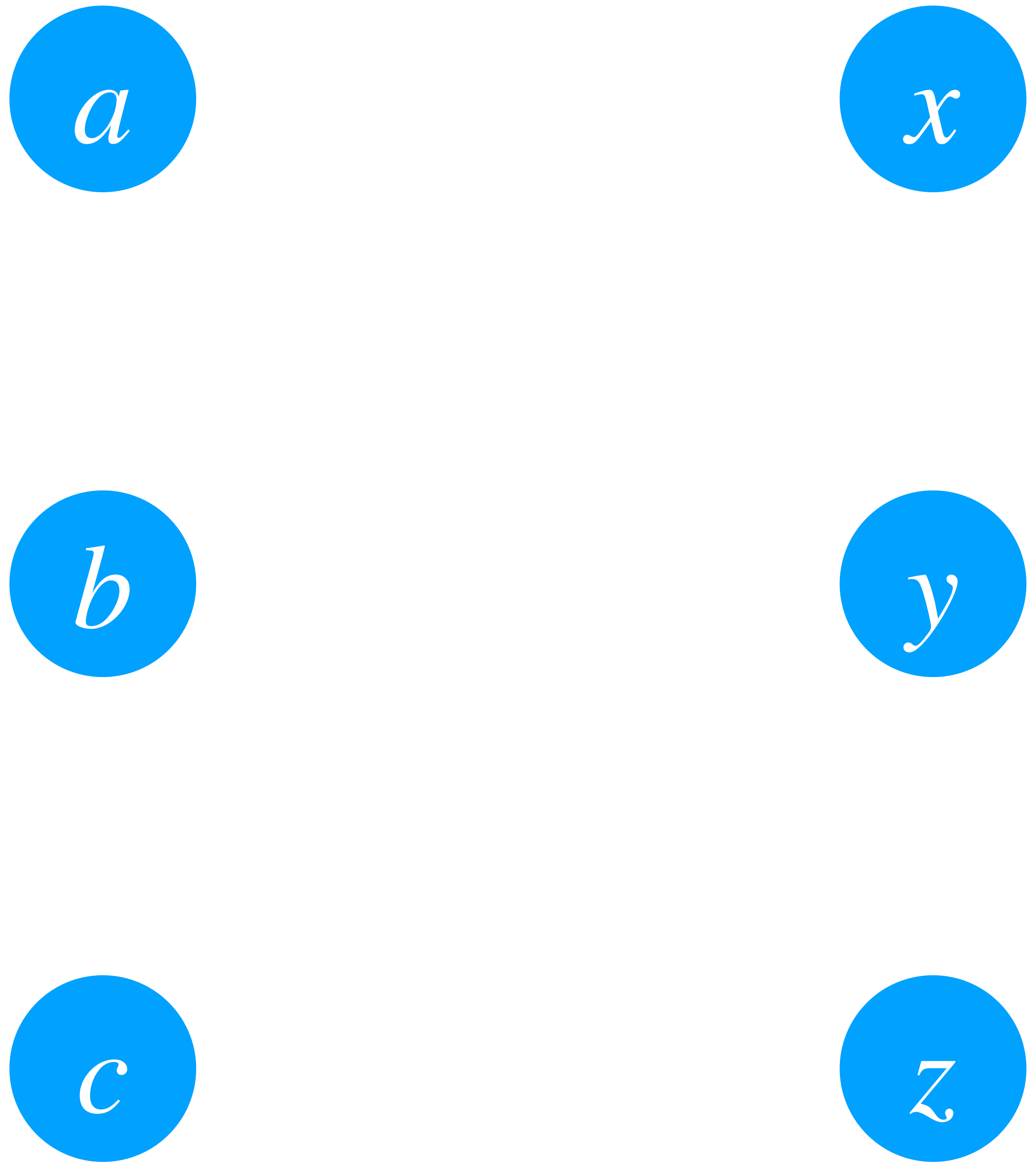
- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur
  at that point return the provisional matching
  - each $a \in A$ proposes to their most preferred company that has not yet rejected them
  - each $x \in X$ provisionally accepts their favorite proposer and rejects the rest

$x \succ_a y \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad$ $c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad$ $b \succ_z a \succ_z c$



Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur

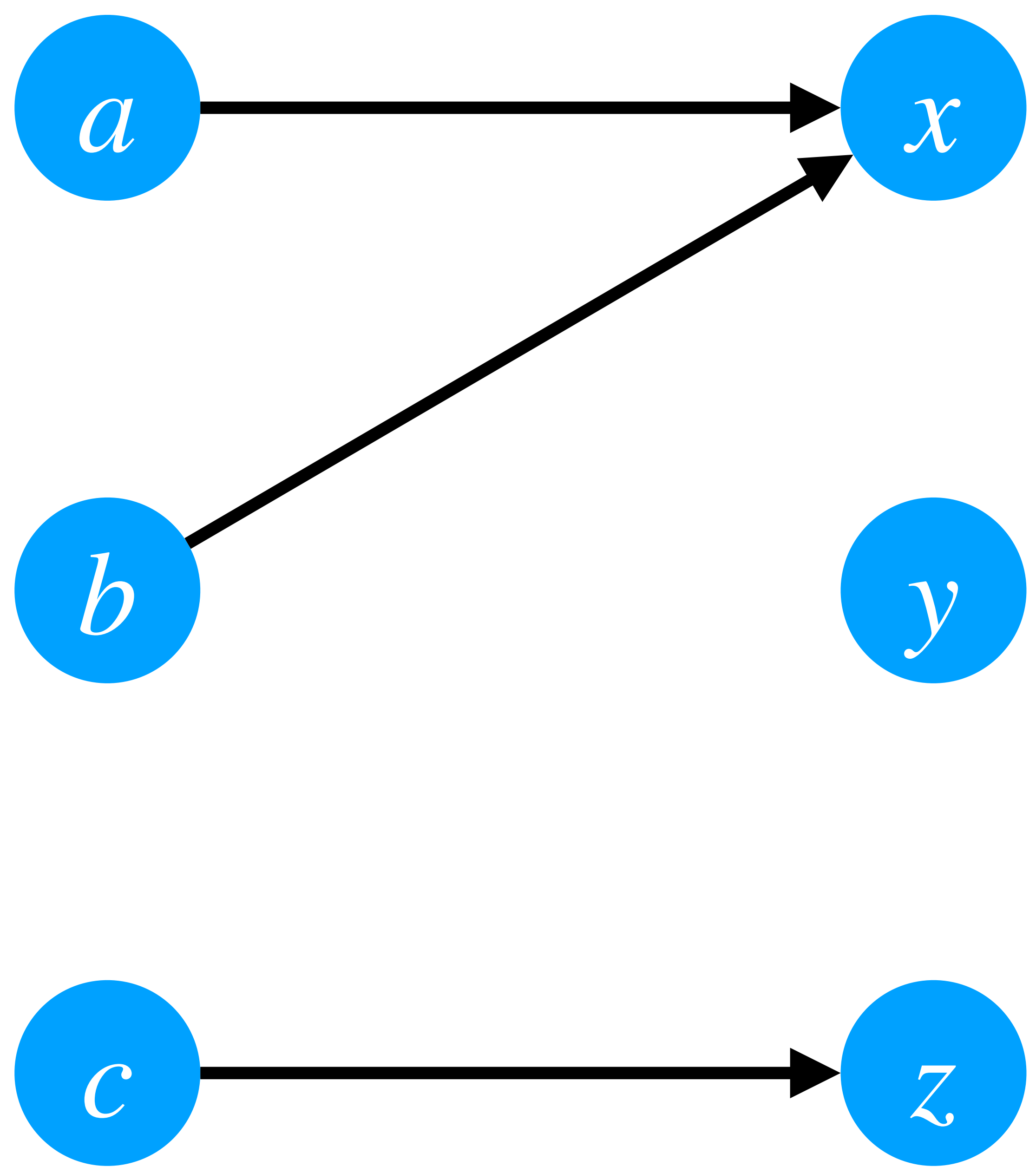
  at that point return the provisional matching

  - each $a \in A$ proposes to their most preferred company that has not yet rejected them
  - each $x \in X$ provisionally accepts their favorite proposer and rejects the rest

$x \succ_a y \succ_a z$ $\quad a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad b \succ_z a \succ_z c$



Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

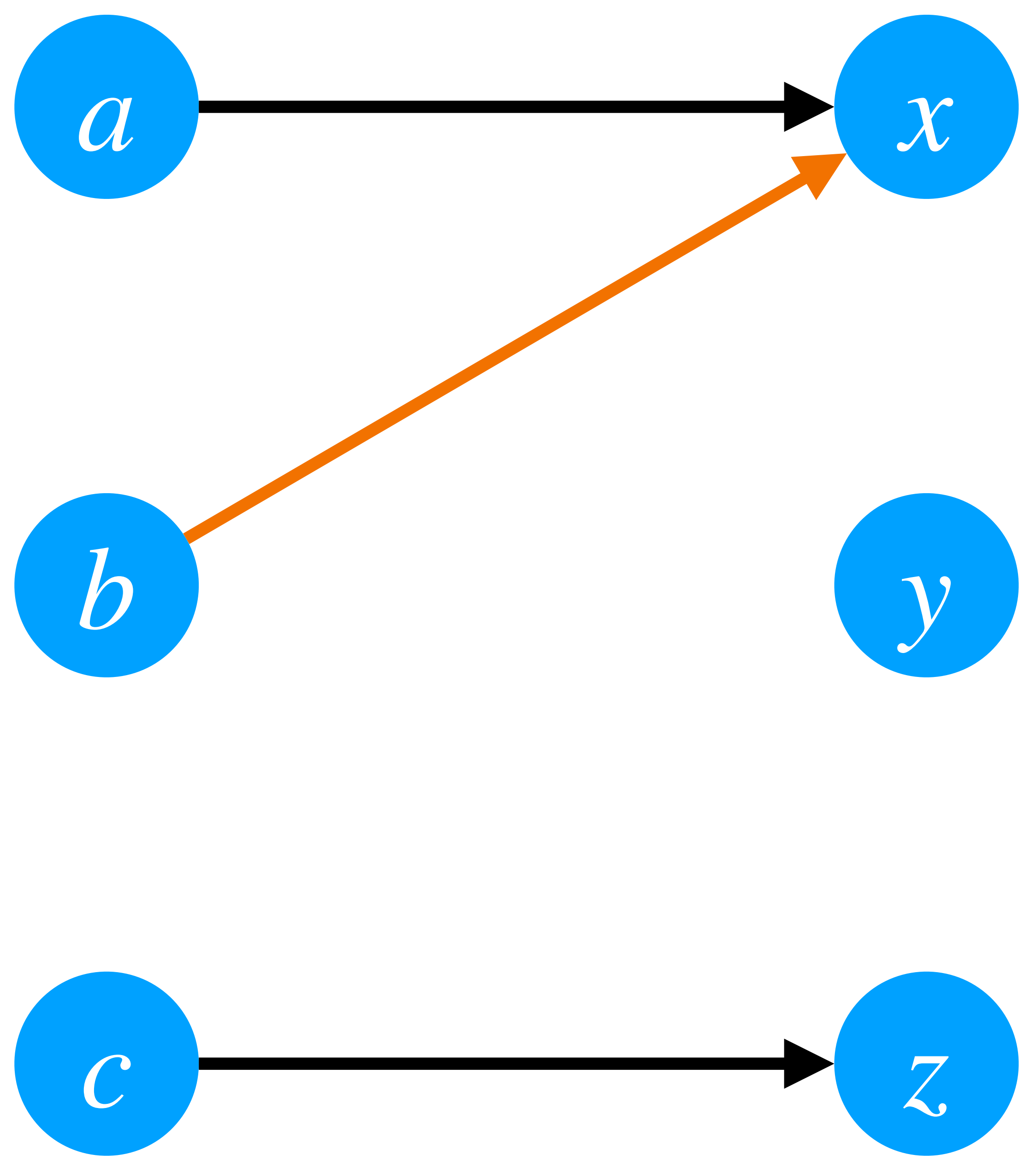
- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur
  at that point return the provisional matching
  - each $a \in A$ proposes to their most preferred company that has not yet rejected them
  - each $x \in X$ provisionally accepts their favorite proposer and rejects the rest

$x \succ_a y \succ_a z$ $\quad a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad b \succ_z a \succ_z c$



Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur

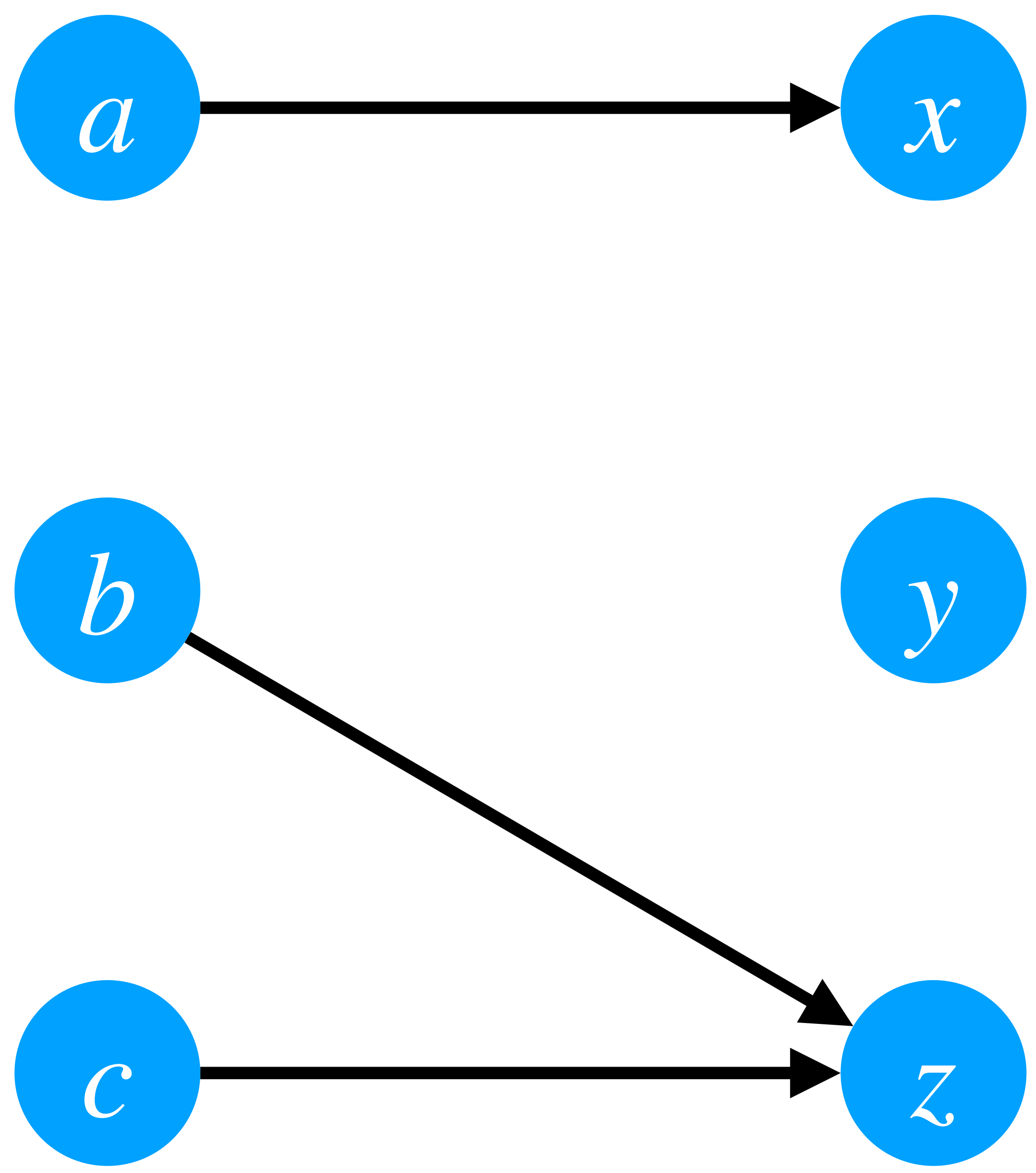
  at that point return the provisional matching

  - each $a \in A$ proposes to their most preferred company that has not yet rejected them
  - each $x \in X$ provisionally accepts their favorite proposer and rejects the rest

$x \succ_a y \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad$ $c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad$ $b \succ_z a \succ_z c$



Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur

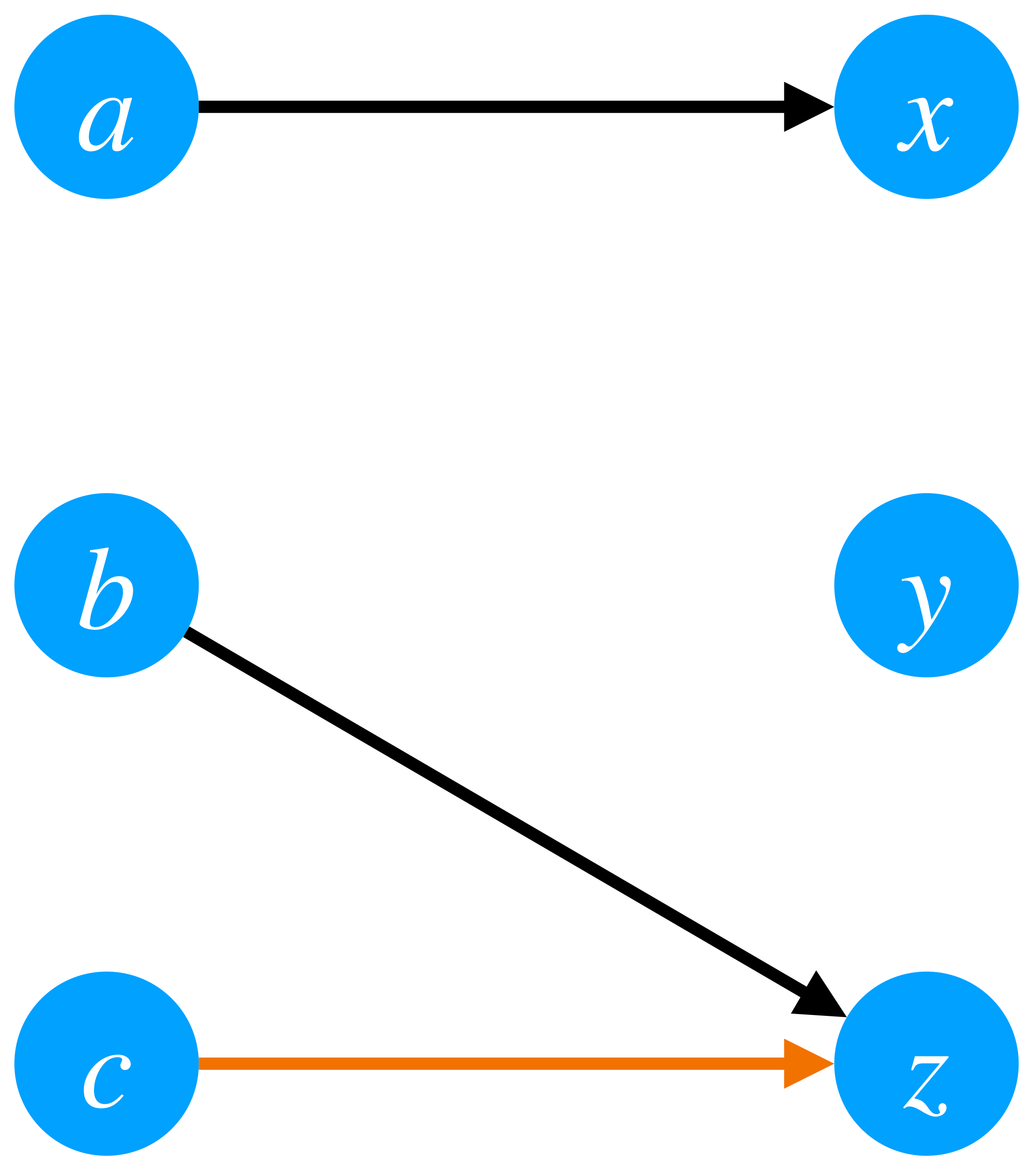
  at that point return the provisional matching

  - each $a \in A$ proposes to their most preferred company that has not yet rejected them
  - each $x \in X$ provisionally accepts their favorite proposer and rejects the rest

$x \succ_a y \succ_a z$ $\quad a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad b \succ_z a \succ_z c$



Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur

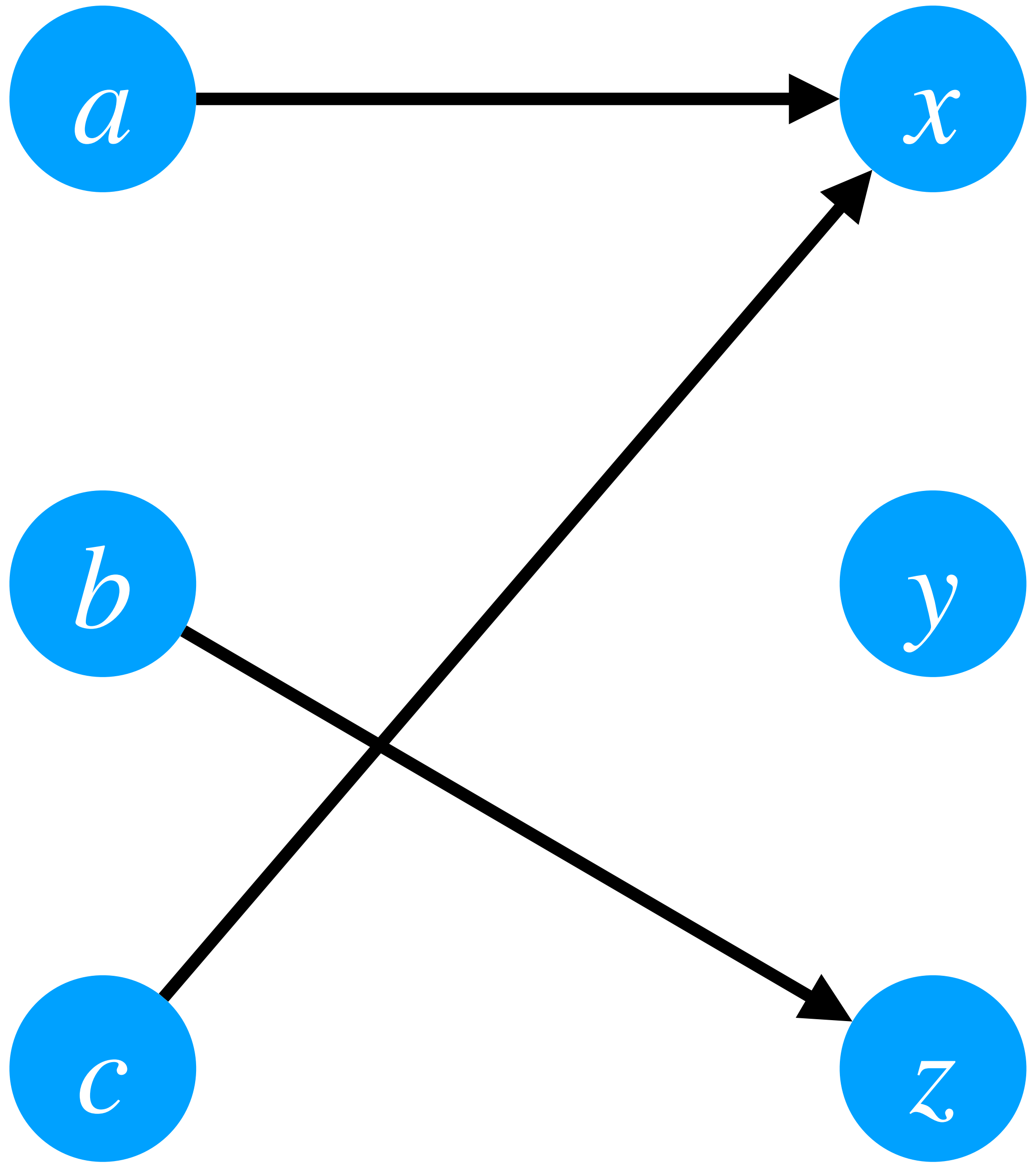
  at that point return the provisional matching

  - each $a \in A$ proposes to their most preferred company that has not yet rejected them
  - each $x \in X$ provisionally accepts their favorite proposer and rejects the rest

$x \succ_a y \succ_a z \qquad a \succ_x b \succ_x c$

$x \succ_b z \succ_b y \qquad c \succ_y a \succ_y b$

$z \succ_c x \succ_c y \qquad b \succ_z a \succ_z c$



Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur

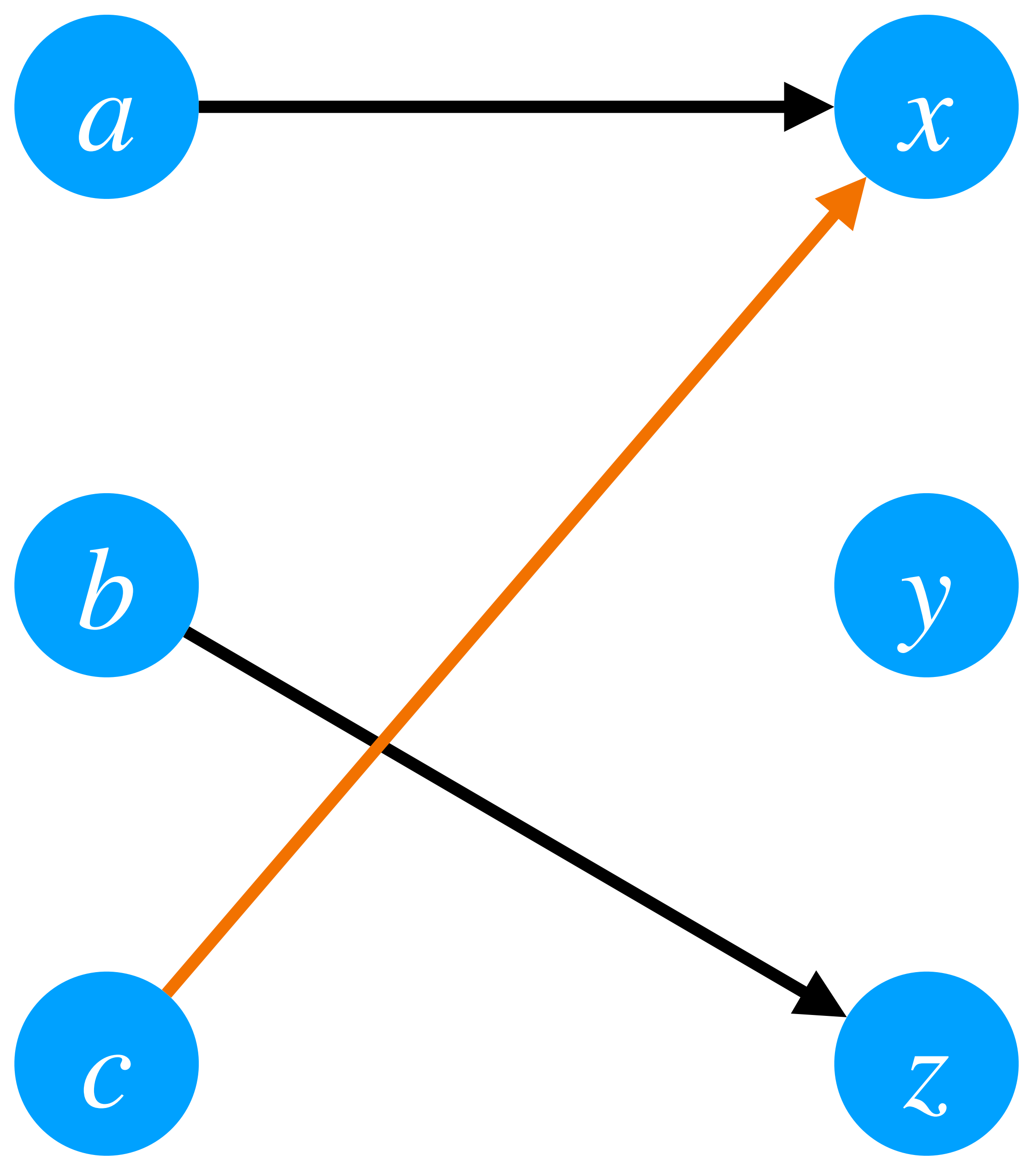
  at that point return the provisional matching

  - each $a \in A$ proposes to their most preferred company that has not yet rejected them
  - each $x \in X$ provisionally accepts their favorite proposer and rejects the rest

$x \succ_a y \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $\quad$ $c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $\quad$ $b \succ_z a \succ_z c$



Gale-Shapley's Deferred Acceptance

Theorem [Gale, Shapley '62]

A stable matching always exists and can be found efficiently.

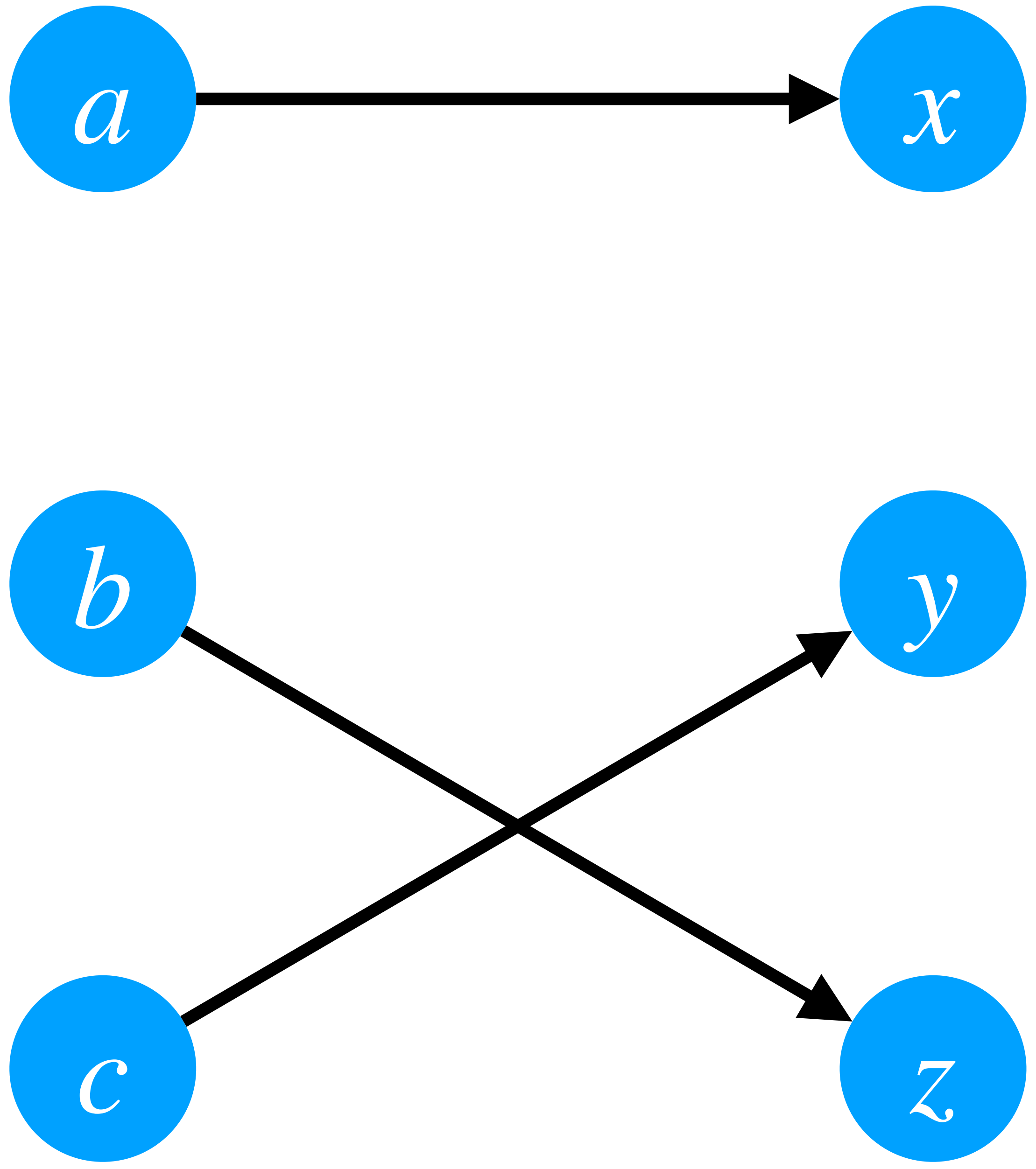
- proof is constructive and given by the **Deferred Acceptance (DA)** mechanism

- repeat the following until no rejections occur
  at that point return the provisional matching
  - each $a \in A$ proposes to their most preferred company that has not yet rejected them
  - each $x \in X$ provisionally accepts their favorite proposer and rejects the rest

$x \succ_a y \succ_a z$ $a \succ_x b \succ_x c$

$x \succ_b z \succ_b y$ $c \succ_y a \succ_y b$

$z \succ_c x \succ_c y$ $b \succ_z a \succ_z c$



Deferred Acceptance: Stability and Running Time

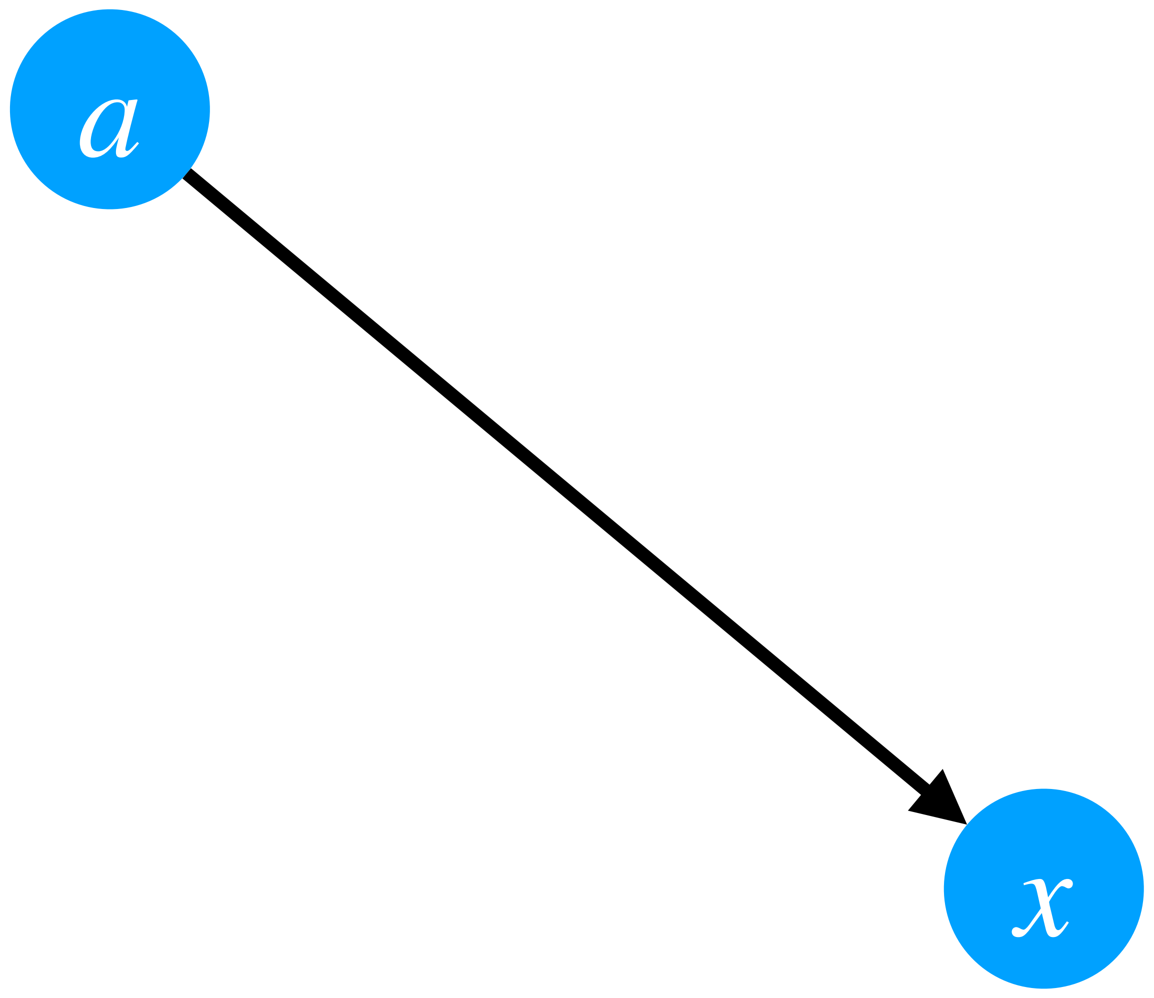
- suppose DA terminates with matching M
  and (a, x) prefer each other over M

Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M and (a, x) prefer each other over M
  - a must have proposed to x at some point

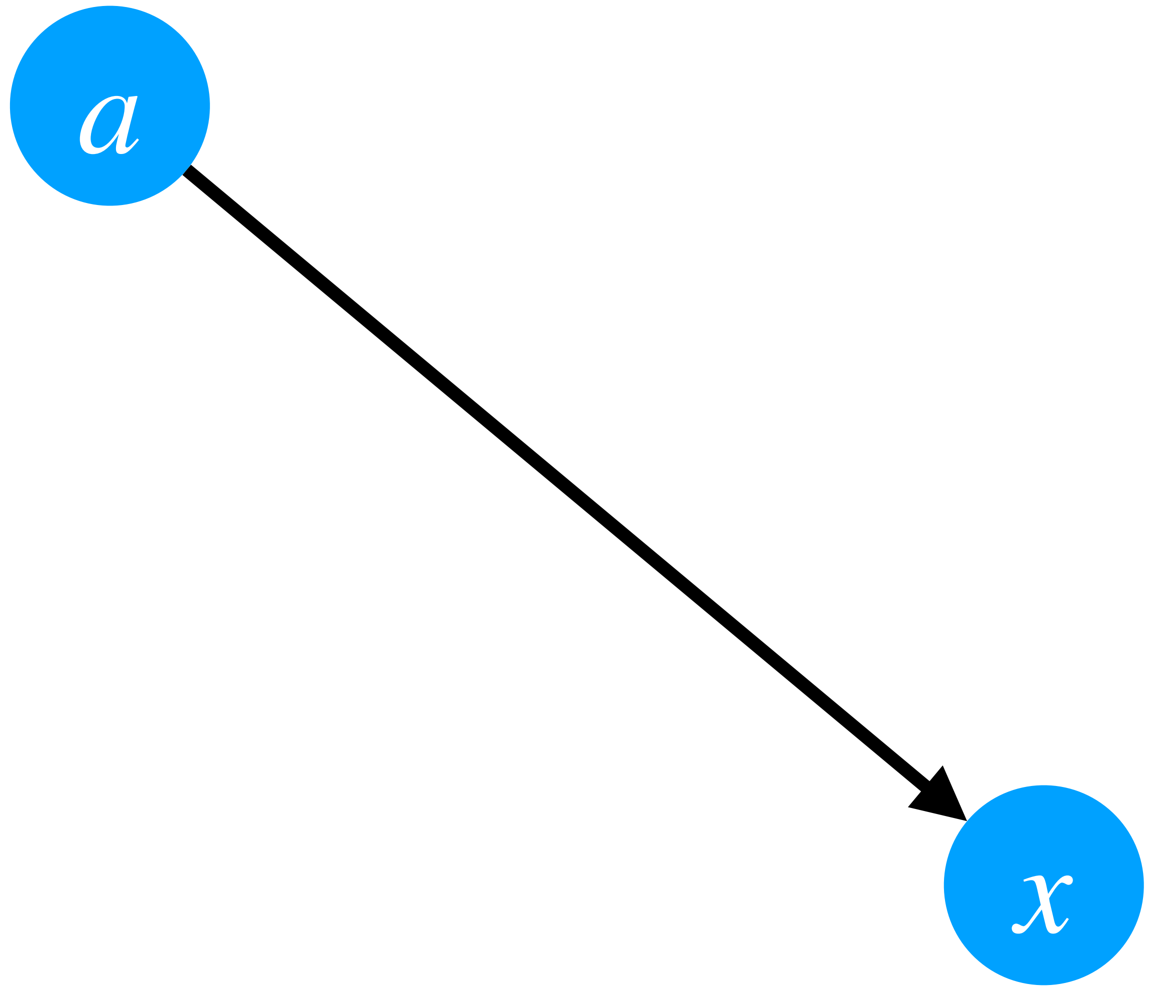
Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M and (a, x) prefer each other over M
  - a must have proposed to x at some point



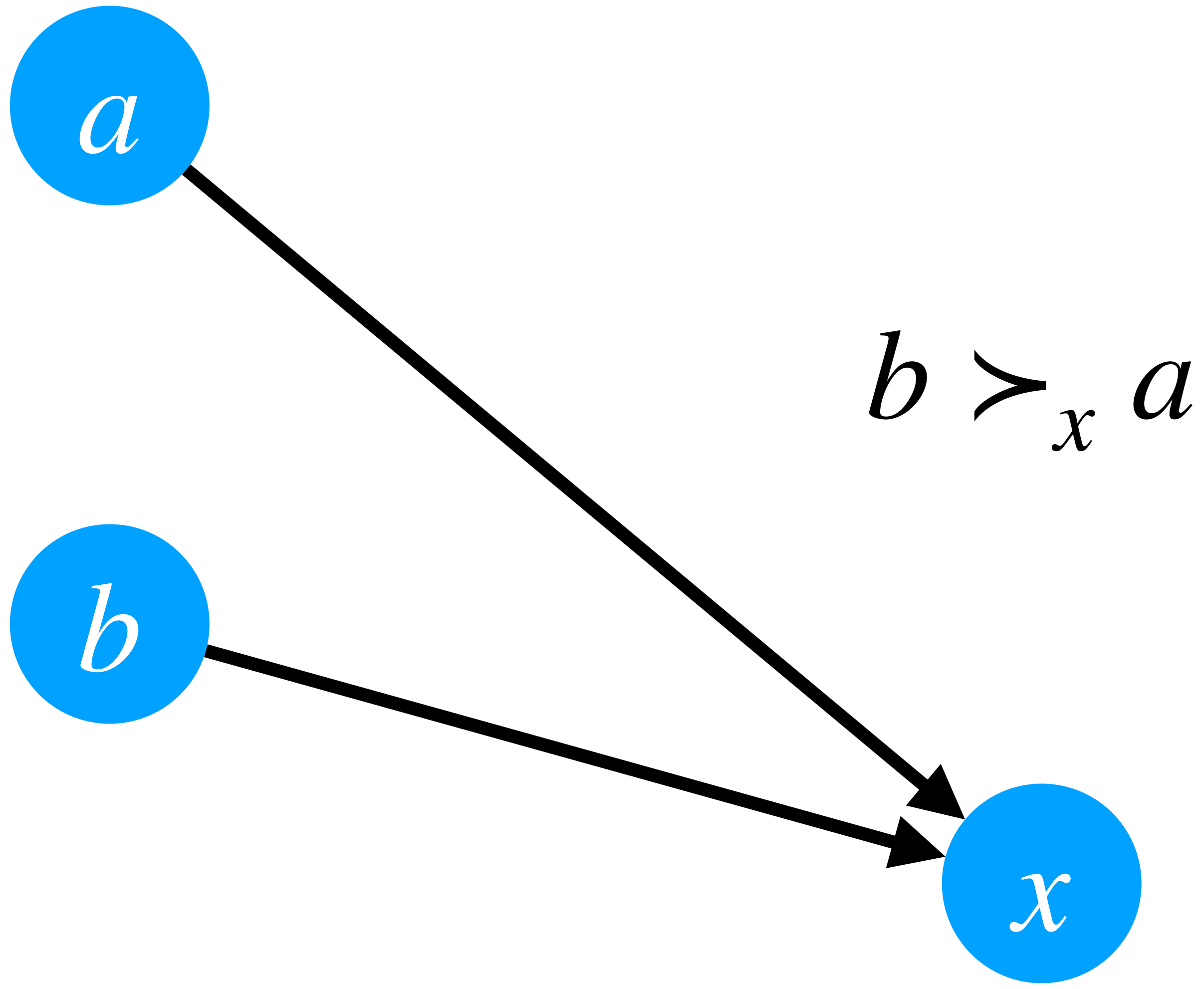
Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M
  and (a, x) prefer each other over M
  - a must have proposed to x at some point
  - x rejected a and got matched to a more preferred applicant



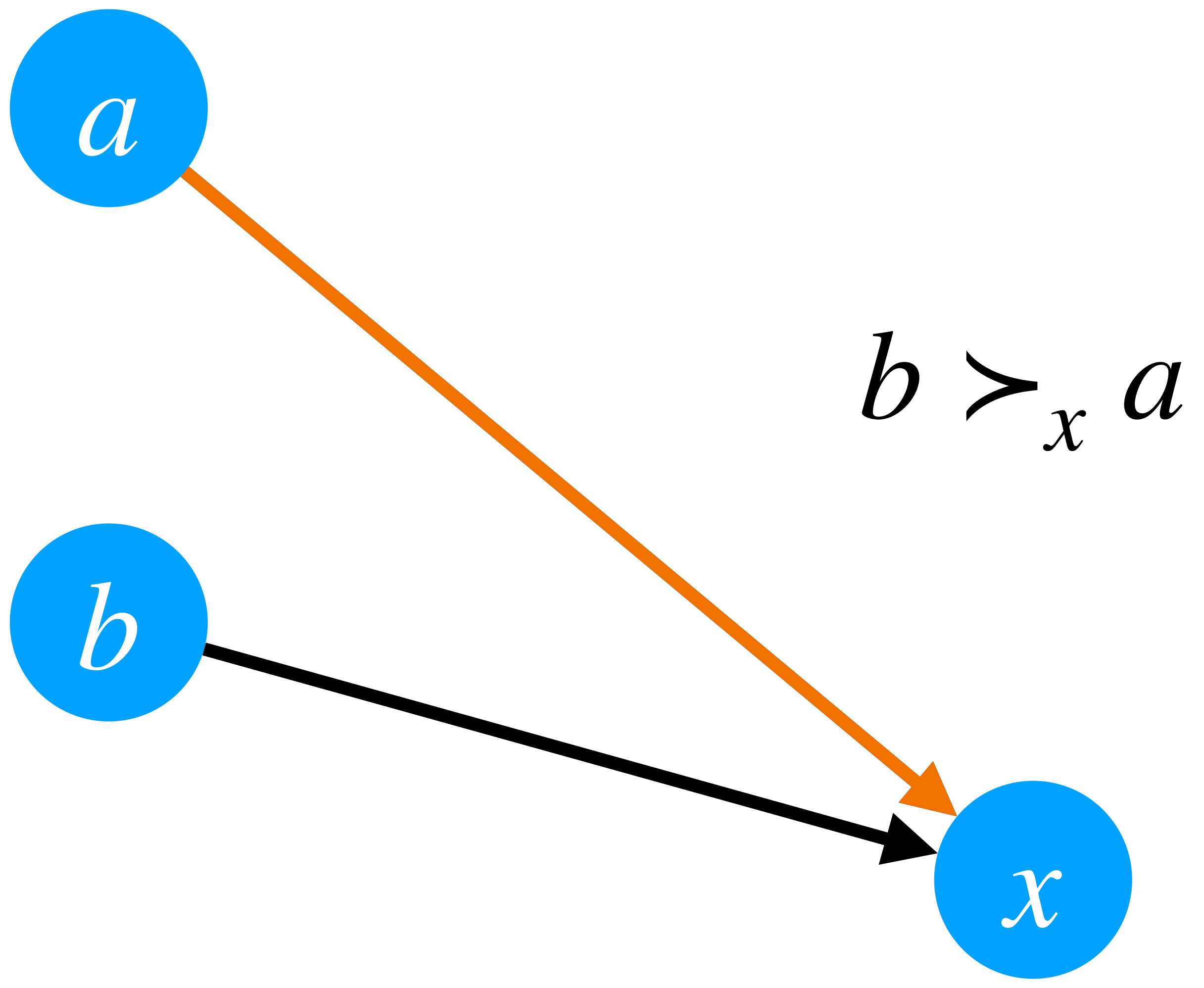
Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M
  and (a, x) prefer each other over M
  - a must have proposed to x at some point
  - x rejected a and got matched to a more preferred applicant



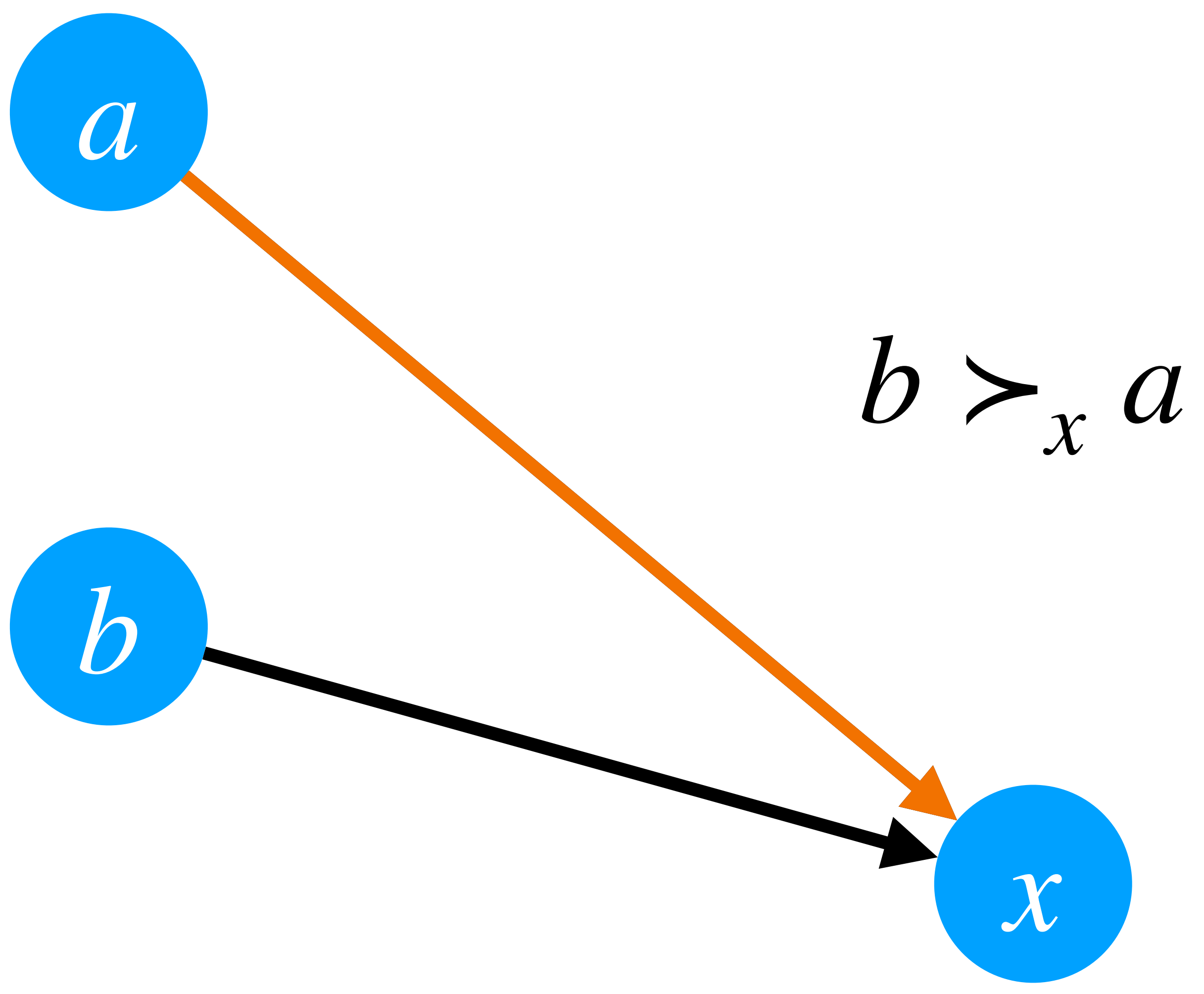
Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M and (a, x) prefer each other over M
  - a must have proposed to x at some point
  - x rejected a and got matched to a more preferred applicant



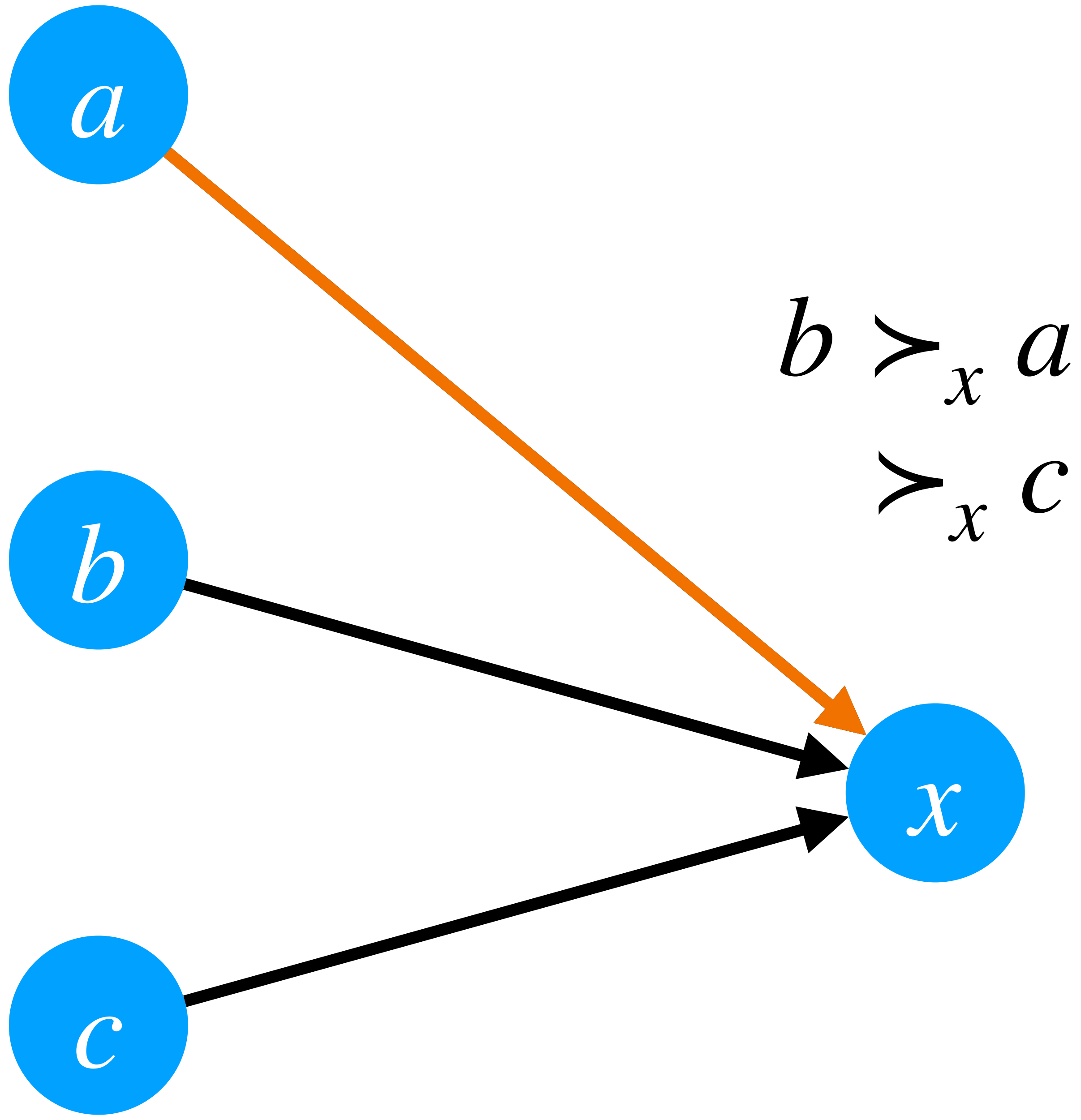
Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M
  and (a, x) prefer each other over M
    - a must have proposed to x at some point
    - x rejected a and got matched to a more preferred applicant
    - in later iterations, x can only be matched
      to a yet more preferred applicant



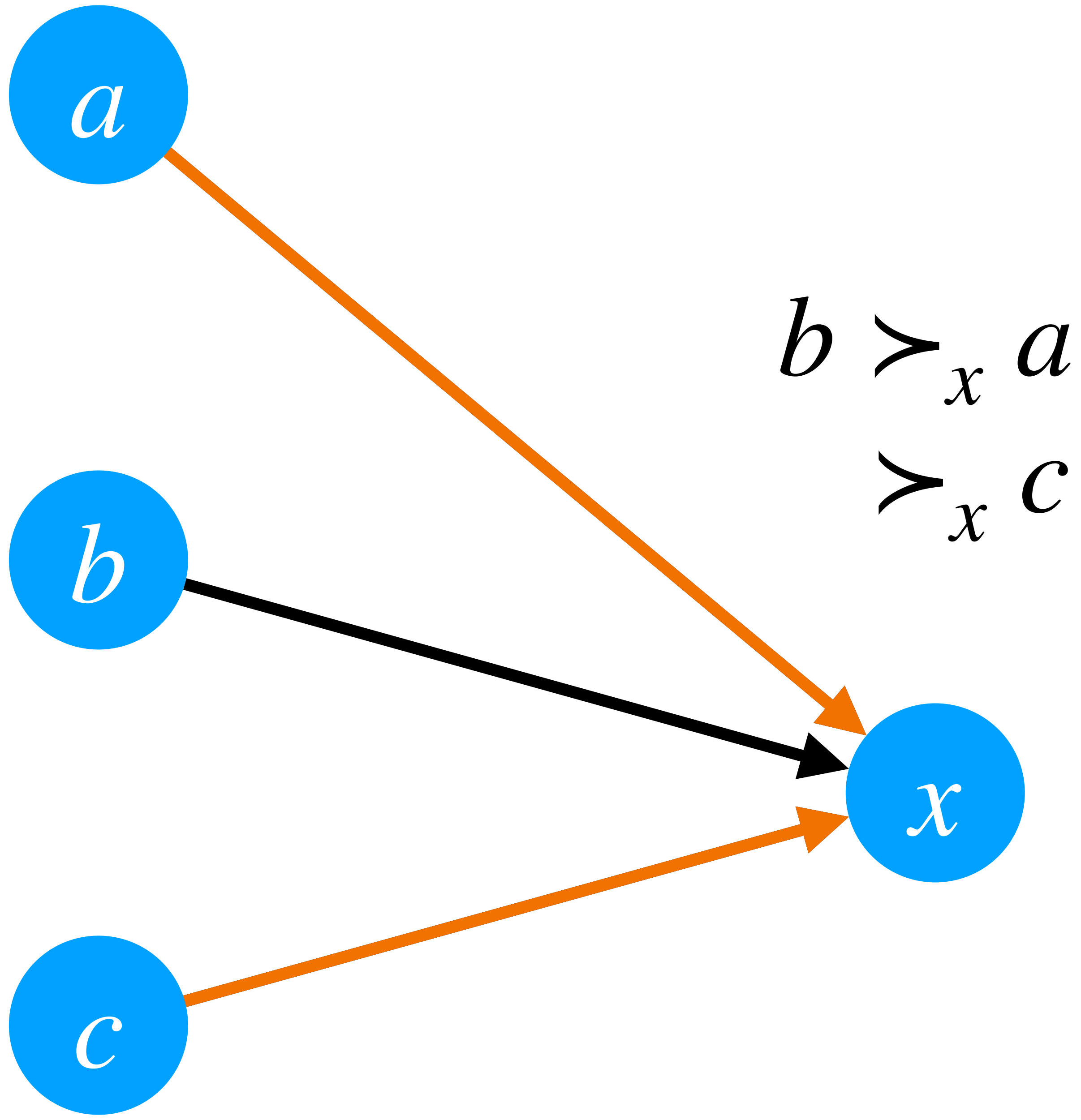
Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M and (a, x) prefer each other over M
  - a must have proposed to x at some point
  - x rejected a and got matched to a more preferred applicant
  - in later iterations, x can only be matched to a yet more preferred applicant



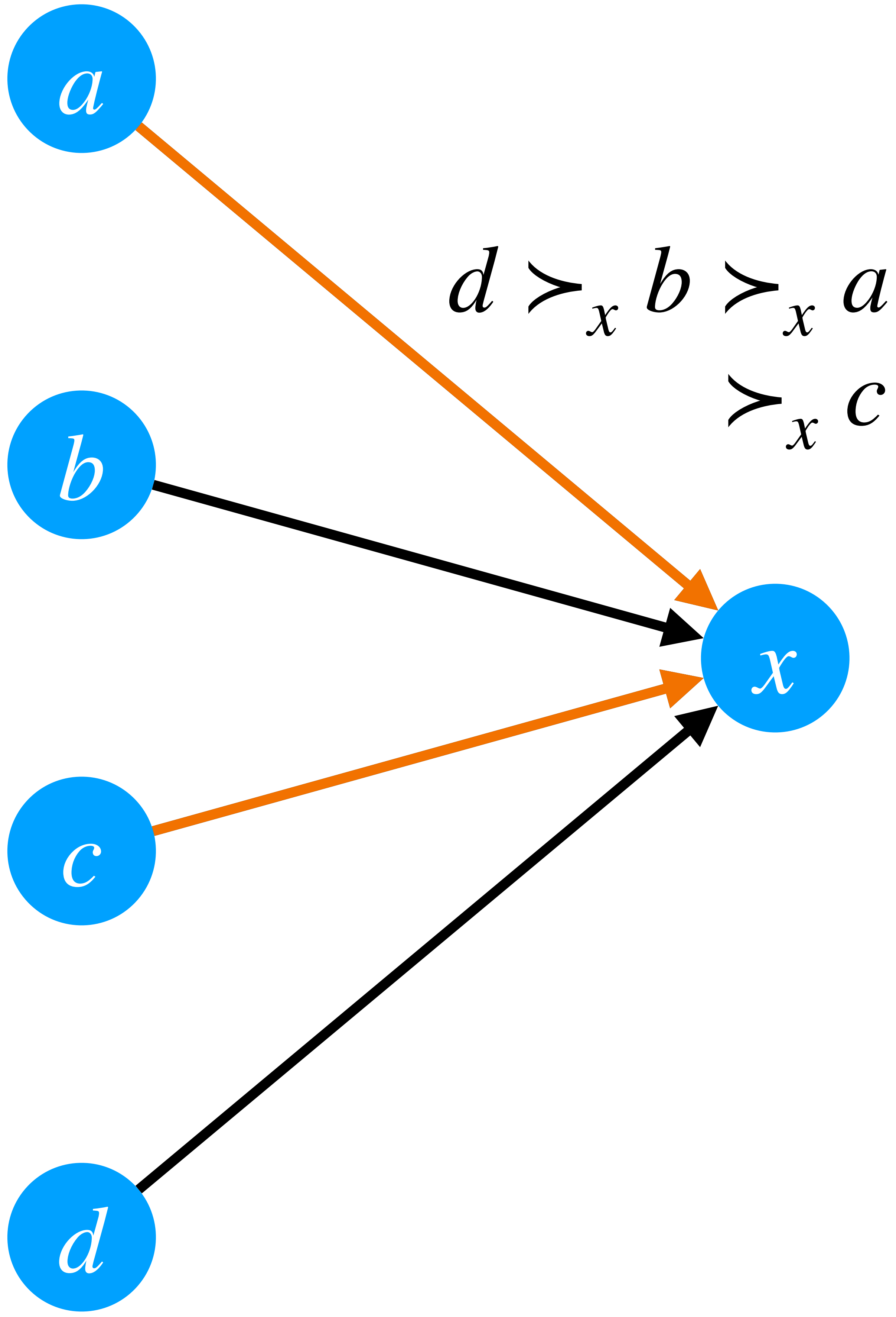
Deferred Acceptance: Stability and Running Time

- suspose DA terminates with matching M and (a, x) prefer each other over M
  - a must have proposed to x at some point
  - x rejected a and got matched to a more preferred applicant
  - in later iterations, x can only be matched to a yet more preferred applicant



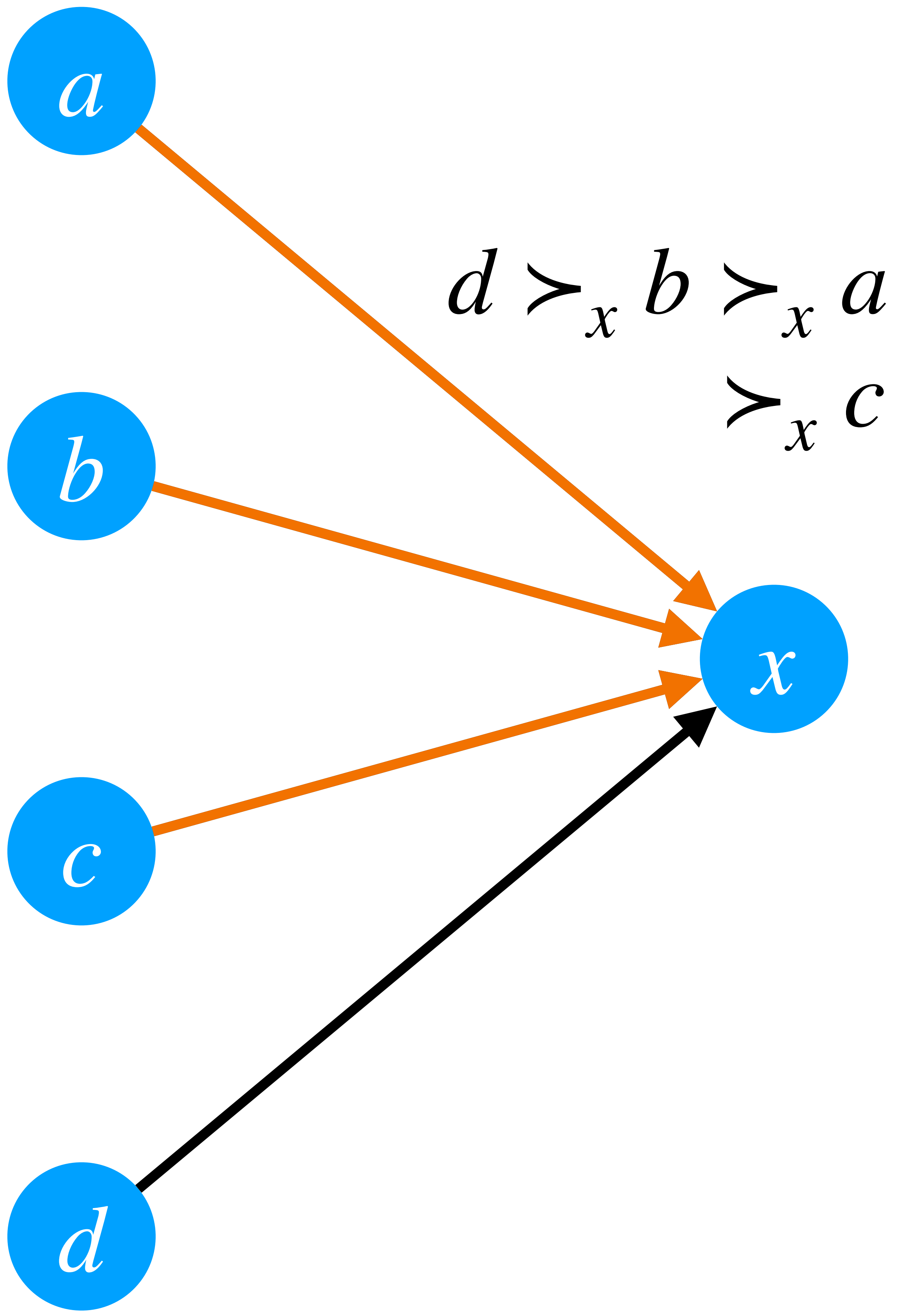
Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M and (a, x) prefer each other over M
  - a must have proposed to x at some point
  - x rejected a and got matched to a more preferred applicant
  - in later iterations, x can only be matched to a yet more preferred applicant



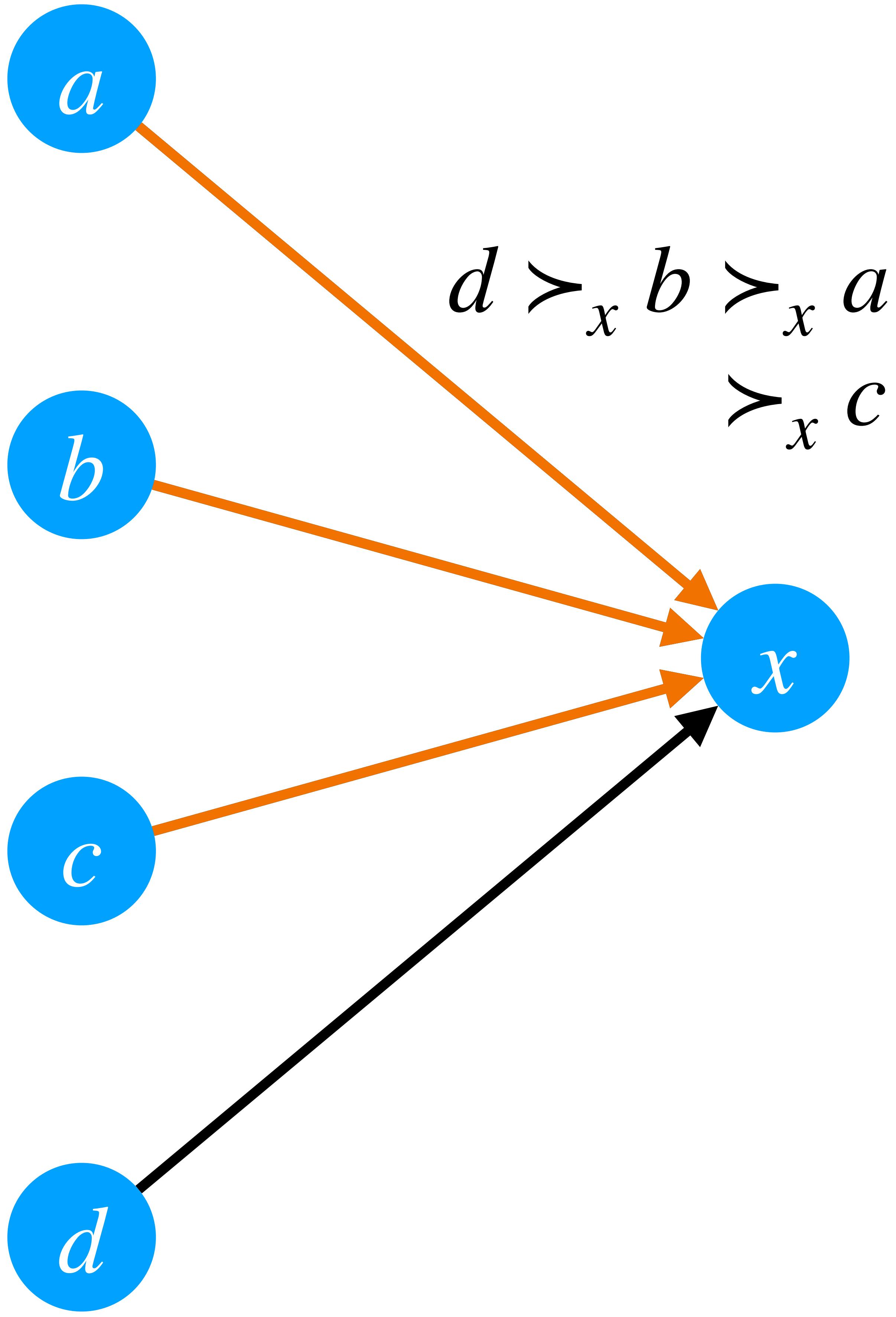
Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M
  and (a, x) prefer each other over M
  - a must have proposed to x at some point
  - x rejected a and got matched to a more preferred applicant
  - in later iterations, x can only be matched
    to a yet more preferred applicant



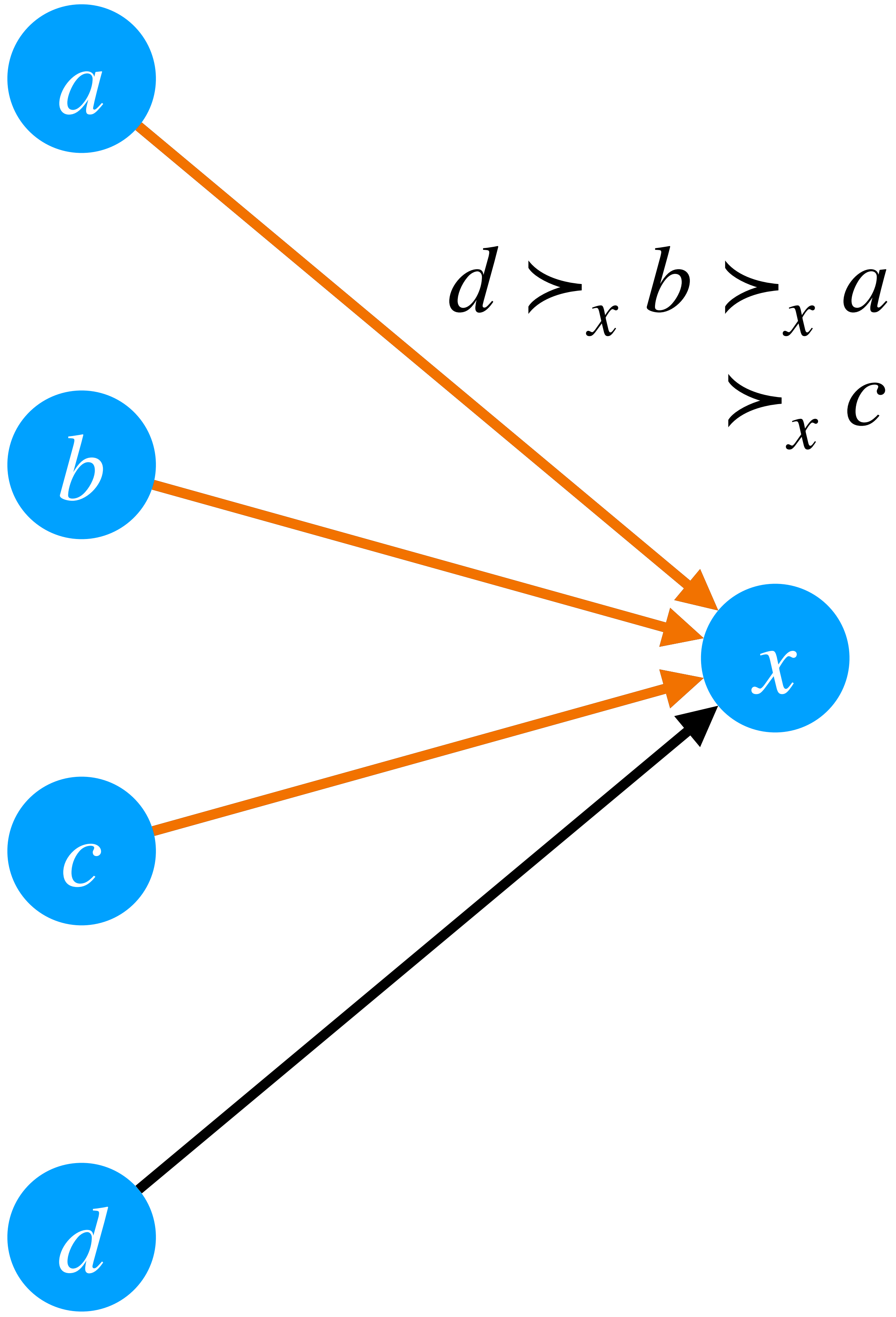
Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M and (a, x) prefer each other over M
  - a must have proposed to x at some point
  - x rejected a and got matched to a more preferred applicant
  - in later iterations, x can only be matched to a yet more preferred applicant
  - in the end, x is matched to an applicant they prefer over a



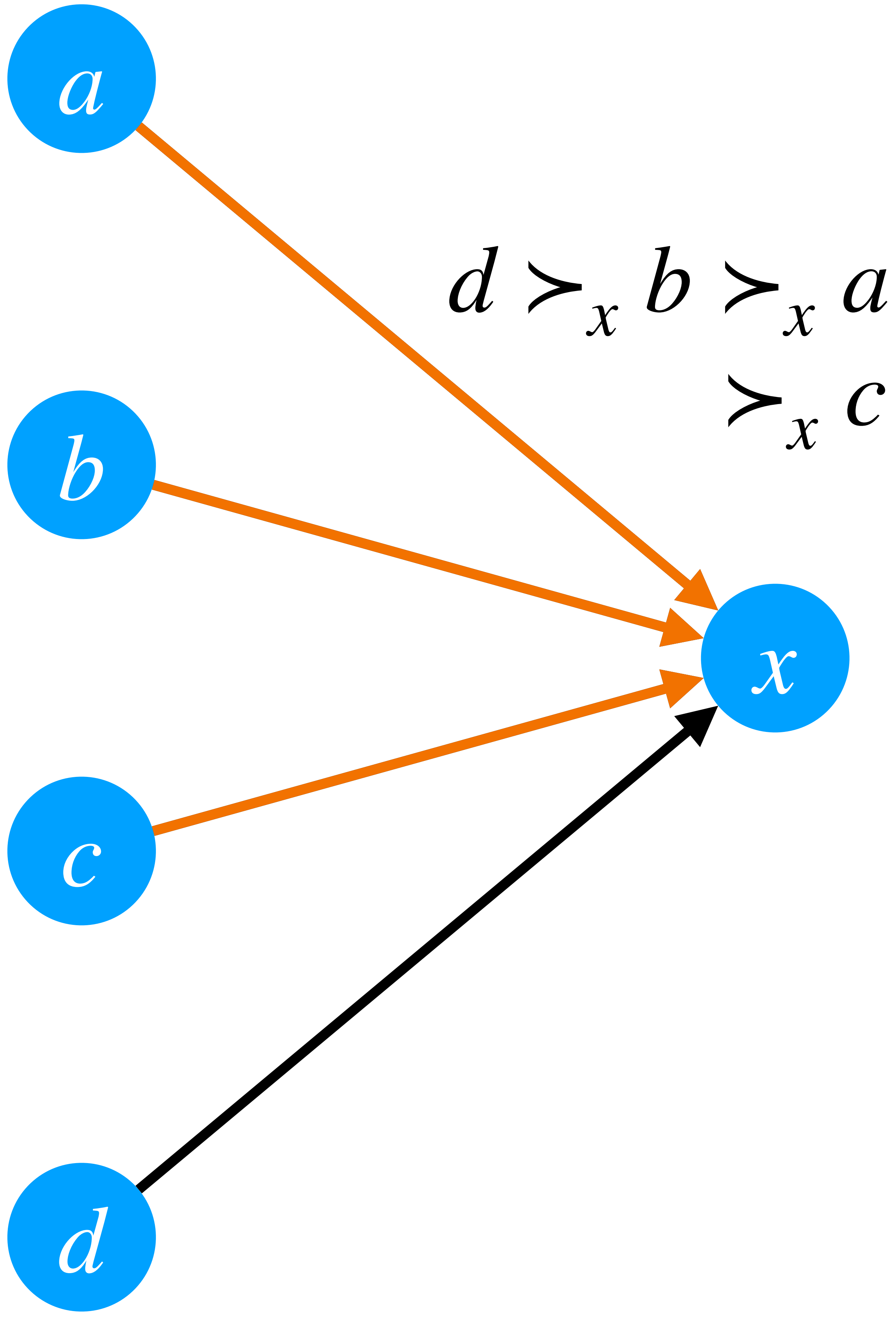
Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M and (a, x) prefer each other over M
  - a must have proposed to x at some point
  - x rejected a and got matched to a more preferred applicant
  - in later iterations, x can only be matched to a yet more preferred applicant
  - in the end, x is matched to an applicant they prefer over a
- DA terminates after at most $|A| \cdot |X| + 1$ steps



Deferred Acceptance: Stability and Running Time

- suppose DA terminates with matching M
  and (a, x) prefer each other over M
  - a must have proposed to x at some point
  - x rejected a and got matched to a more preferred applicant
  - in later iterations, x can only be matched
    to a yet more preferred applicant
  - in the end, x is matched to an applicant they prefer over a
- DA terminates after at most $|A| \cdot |X| + 1$ steps
  - at least one new rejection between a pair (a, x)
    occurs in every step except for the last one



How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing

- for any agent, call an agent from the other side
  attainable if there exists a stable matching
  in which these two agents are matched

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing

- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
one can analogously run company-proposing

- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

- **Theorem**

  In the outcome of applicant-proposing DA:

  - every applicant gets matched to their most-preferred attainable company,

- by contradiction: an applicant gets rejected by their most-preferred attainable company

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,

- by contradiction: an applicant gets rejected by their most-preferred attainable company
- call (a, x) the first such pair
  and say x rejects a because of b

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,

- by contradiction: an applicant gets rejected by their most-preferred attainable company
- call (a, x) the first such pair
  and say x rejects a because of b
- b likes x at least as much as their most-preferred attainable company
  because of how we fixed (a, x)

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
one can analogously run company-proposing

- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,

- by contradiction: an applicant gets rejected by their most-preferred attainable company

- call (a, x) the first such pair
and say x rejects a because of b

- b likes x at least as much as their most-preferred attainable company
because of how we fixed (a, x)

- since x is attainable for a, there exists a stable matching in which they are matched and b is matched to $y \neq x$ or unmatched

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,

- by contradiction: an applicant gets rejected by their most-preferred attainable company
- call (a, x) the first such pair
  and say x rejects a because of b
- b likes x at least as much as their most-preferred attainable company
  because of how we fixed (a, x)
- since x is attainable for a, there exists a stable matching in which they are matched and b is matched to $y \neq x$ or unmatched
- (b, x) form a blocking pair in this matching

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing

- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing

- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

- by contradiction: $x \in X$ is matched to an applicant $a \in A$ that they prefer over some attainable $b \in A$

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

- by contradiction: $x \in X$ is matched to an applicant $a \in A$ that they prefer over some attainable $b \in A$
- there is a stable matching M where x is matched to b and a is matched to $y \neq x$ or unmatched

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
one can analogously run company-proposing

- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

- by contradiction: $x \in X$ is matched to an applicant $a \in A$ that they prefer over some attainable $b \in A$

- there is a stable matching M where x is matched to b and a is matched to $y \neq x$ or unmatched

- by the previous claim, $x \succ_a y$, and by assumption $a \succ_x b$

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

- by contradiction: $x \in X$ is matched to an applicant $a \in A$ that they prefer over some attainable $b \in A$
- there is a stable matching M where x is matched to b and
  a is matched to $y \neq x$ or unmatched
- by the previous claim, $x \succ_a y$,
  and by assumption $a \succ_x b$
- (a, x) form a blocking pair in M

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

Thus, the matched agents are the same in every stable matching.

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

Thus, the matched agents are the same in every stable matching.

- suppose there exists $x \in X$ that is matched to $a \in A$ in applicant-proposing DA
  and to no one in another stable matching M

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

Thus, the matched agents are the same in every stable matching.

- suppose there exists $x \in X$ that is matched to $a \in A$ in applicant-proposing DA and to no one in another stable matching M
- since x is a's most preferred attainable company, (a, x) is a blocking pair in M

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

Thus, the matched agents are the same in every stable matching.

- suppose there exists $x \in X$ that is matched to $a \in A$ in applicant-proposing DA
and to no one in another stable matching M
- since x is a's most preferred attainable company, (a, x) is a blocking pair in M
- companies matched in A-P DA $\subseteq$ companies matched in any stable matching

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

Thus, the matched agents are the same in every stable matching.

- suppose there exists $x \in X$ that is matched to $a \in A$ in applicant-proposing DA
  and to no one in another stable matching M
- since x is a's most preferred attainable company, (a, x) is a blocking pair in M
- companies matched in A-P DA $\subseteq$ companies matched in any stable matching
- since all stable matchings have size $\min\{\,|A|, |X|\,\}$, the set of matched companies is the same in all of them

How Good Is the Outcome for the Agents?

- we have run applicant-proposing DA;
  one can analogously run company-proposing
- for any agent, call an agent from the other side **attainable** if there exists a stable matching in which these two agents are matched

Theorem

In the outcome of applicant-proposing DA:

- every applicant gets matched to their most-preferred attainable company,
- every company gets matched to their least-preferred attainable applicant.

Thus, the matched agents are the same in every stable matching.

- suppose there exists $x \in X$ that is matched to $a \in A$ in applicant-proposing DA and to no one in another stable matching M
- since x is a's most preferred attainable company, (a, x) is a blocking pair in M
- companies matched in A-P DA $\subseteq$ companies matched in any stable matching
- since all stable matchings have size $\min\{|A|, |X|\}$, the set of matched companies is the same in all of them
- the proof is analogous for applicants (starting from company-proposing DA)

Strategic Behavior

- Is DA strategyproof?

Strategic Behavior

- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

Theorem [Roth '82]

No mechanism is stable and strategyproof for both sides.

- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

- **Theorem** [Roth ’82]

  No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings

Strategic Behavior

- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

Theorem [Roth ’82]

No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings
  - for any of them, an agent can misreport and force the other (better) one

- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

Theorem [Roth ’82]

No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a x \succ_a z$ $\quad$ $a \succ_x c \succ_x b$

$x \succ_b y \succ_b z$ $\quad$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad$ $a \succ_z b \succ_z c$

Strategic Behavior

- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

Theorem [Roth '82]

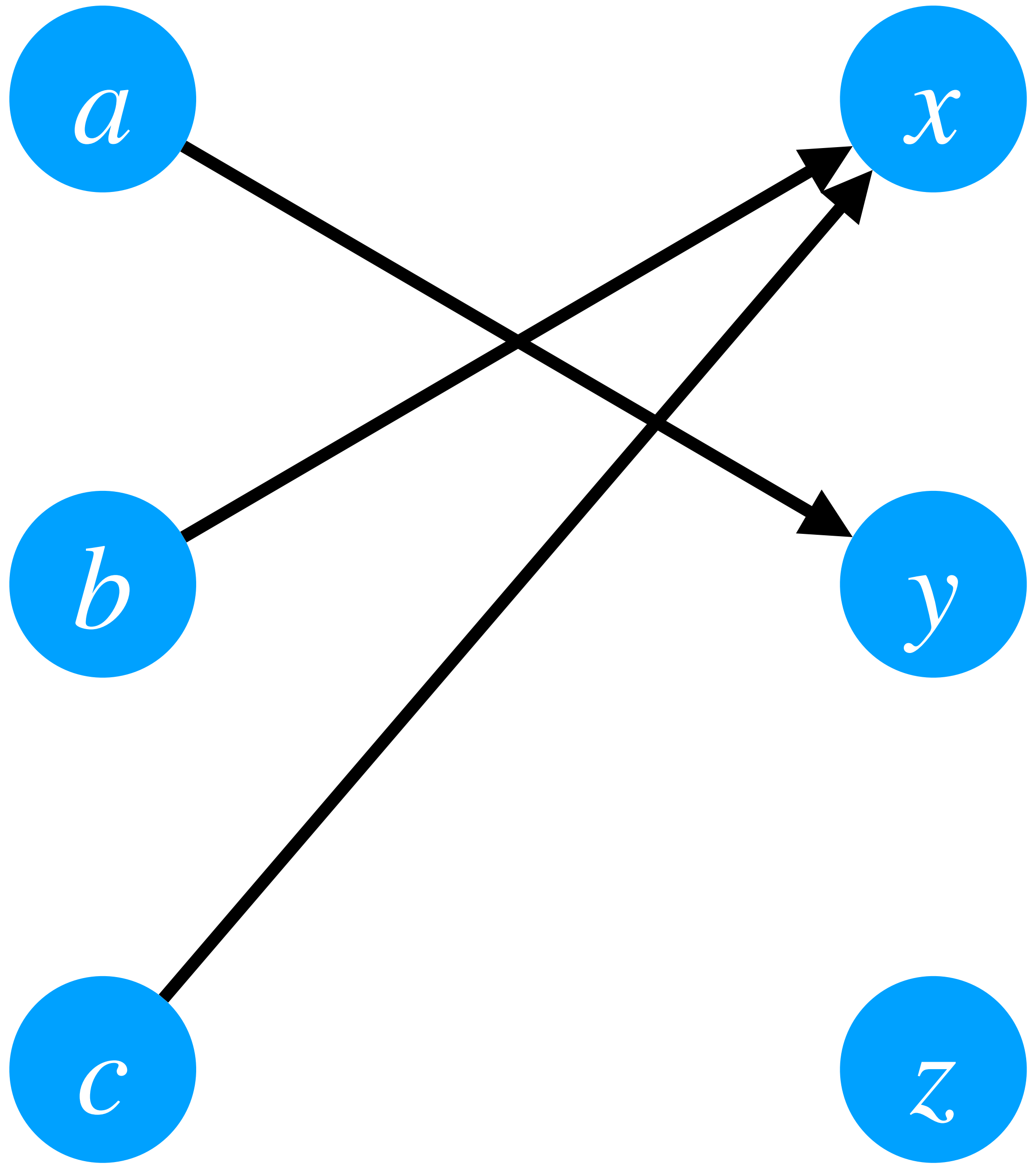
No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a x \succ_a z$ $\quad a \succ_x c \succ_x b$

$x \succ_b y \succ_b z$ $\quad c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad a \succ_z b \succ_z c$



Strategic Behavior

- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

Theorem [Roth '82]

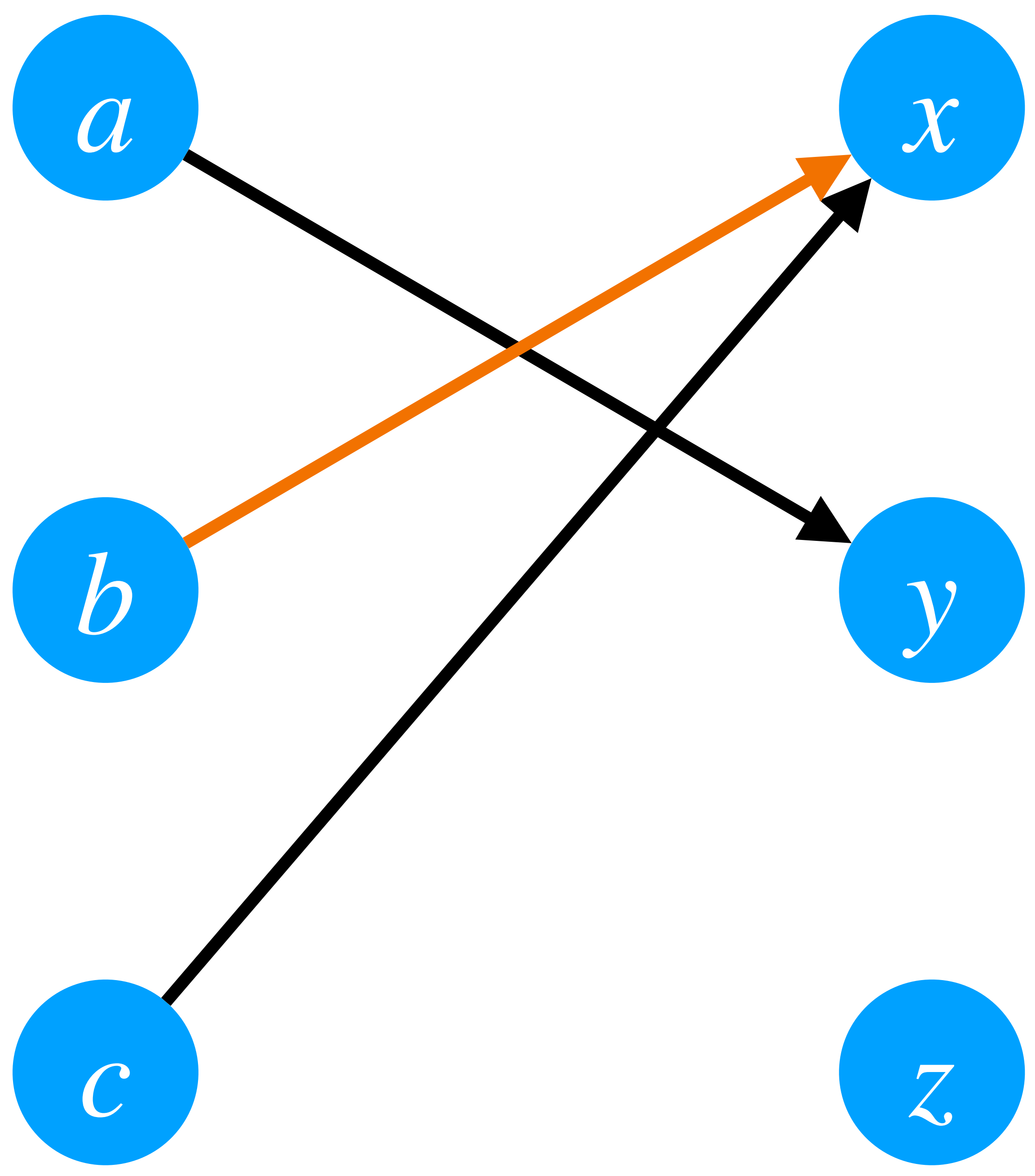
No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a x \succ_a z$ $\quad$ $a \succ_x c \succ_x b$

$x \succ_b y \succ_b z$ $\quad$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad$ $a \succ_z b \succ_z c$



- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

- **Theorem** [Roth '82]

  No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a x \succ_a z$

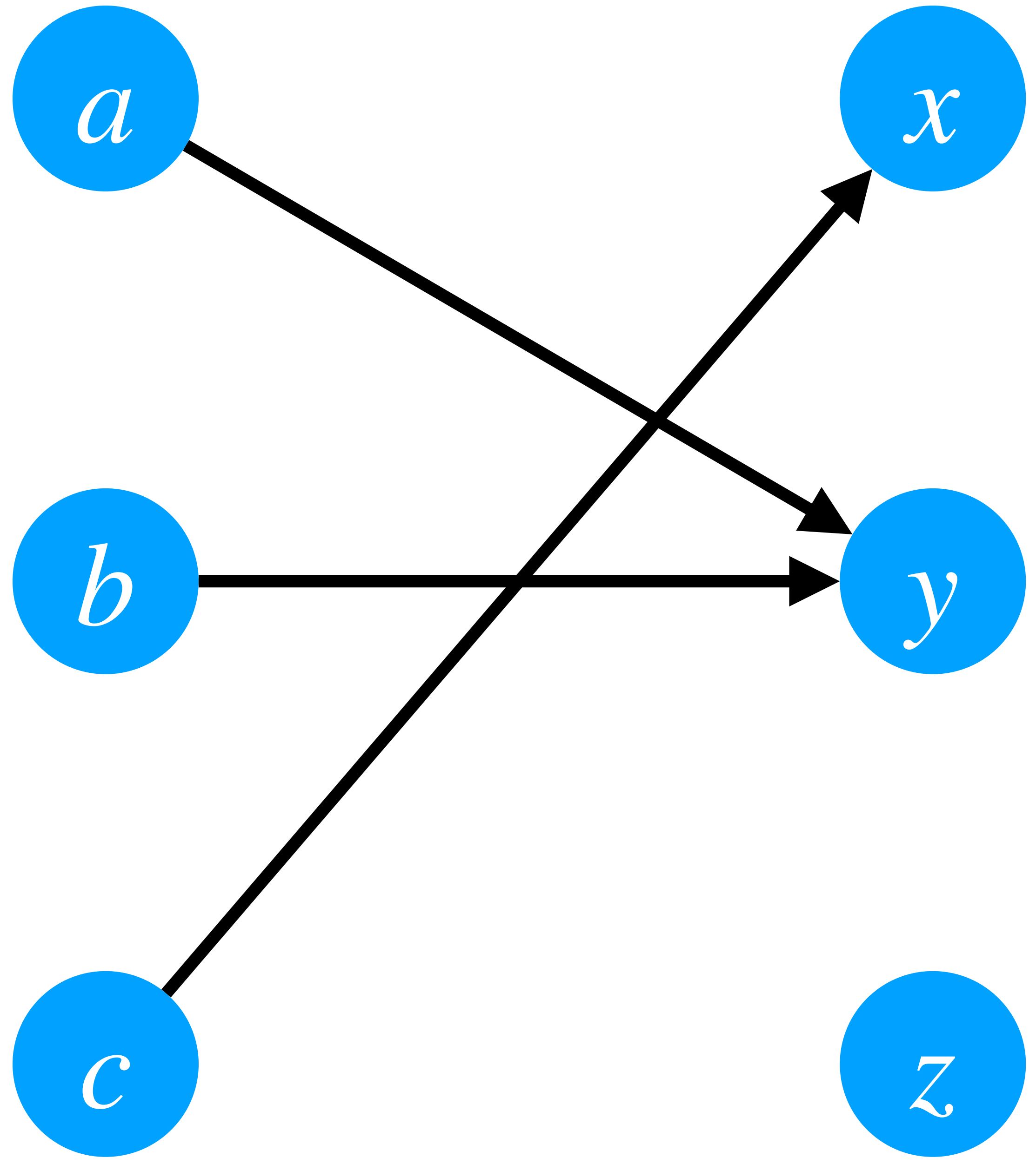
$x \succ_b y \succ_b z$

$x \succ_c y \succ_c z$

$a \succ_x c \succ_x b$

$c \succ_y a \succ_y b$

$a \succ_z b \succ_z c$



- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

Theorem [Roth '82]

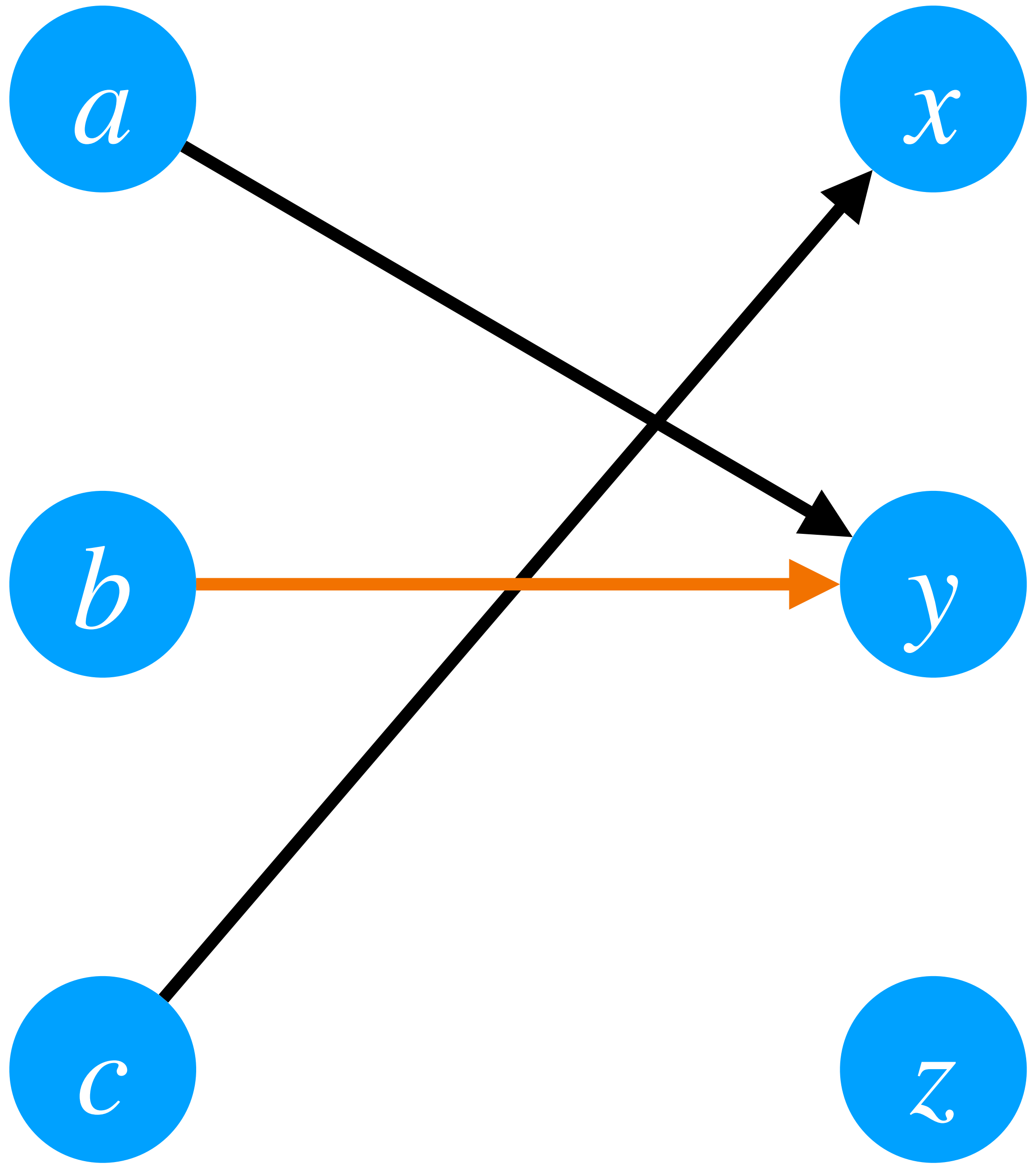
No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a x \succ_a z$ $a \succ_x c \succ_x b$

$x \succ_b y \succ_b z$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $a \succ_z b \succ_z c$



- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

- **Theorem** [Roth '82]

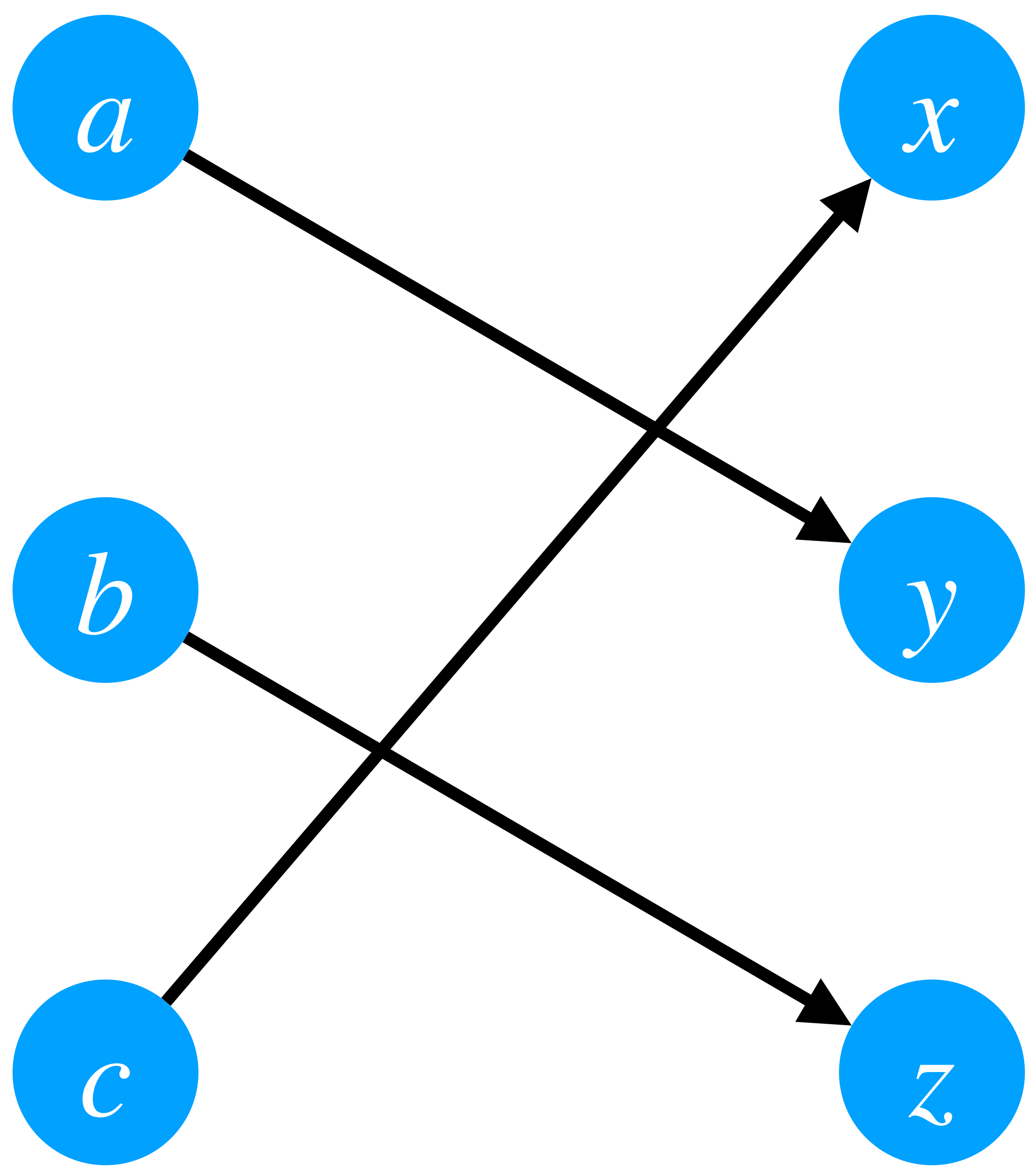
  No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a x \succ_a z$ $\quad a \succ_x c \succ_x b$

$x \succ_b y \succ_b z$ $\quad c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad a \succ_z b \succ_z c$



Strategic Behavior

- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

Theorem [Roth ’82]

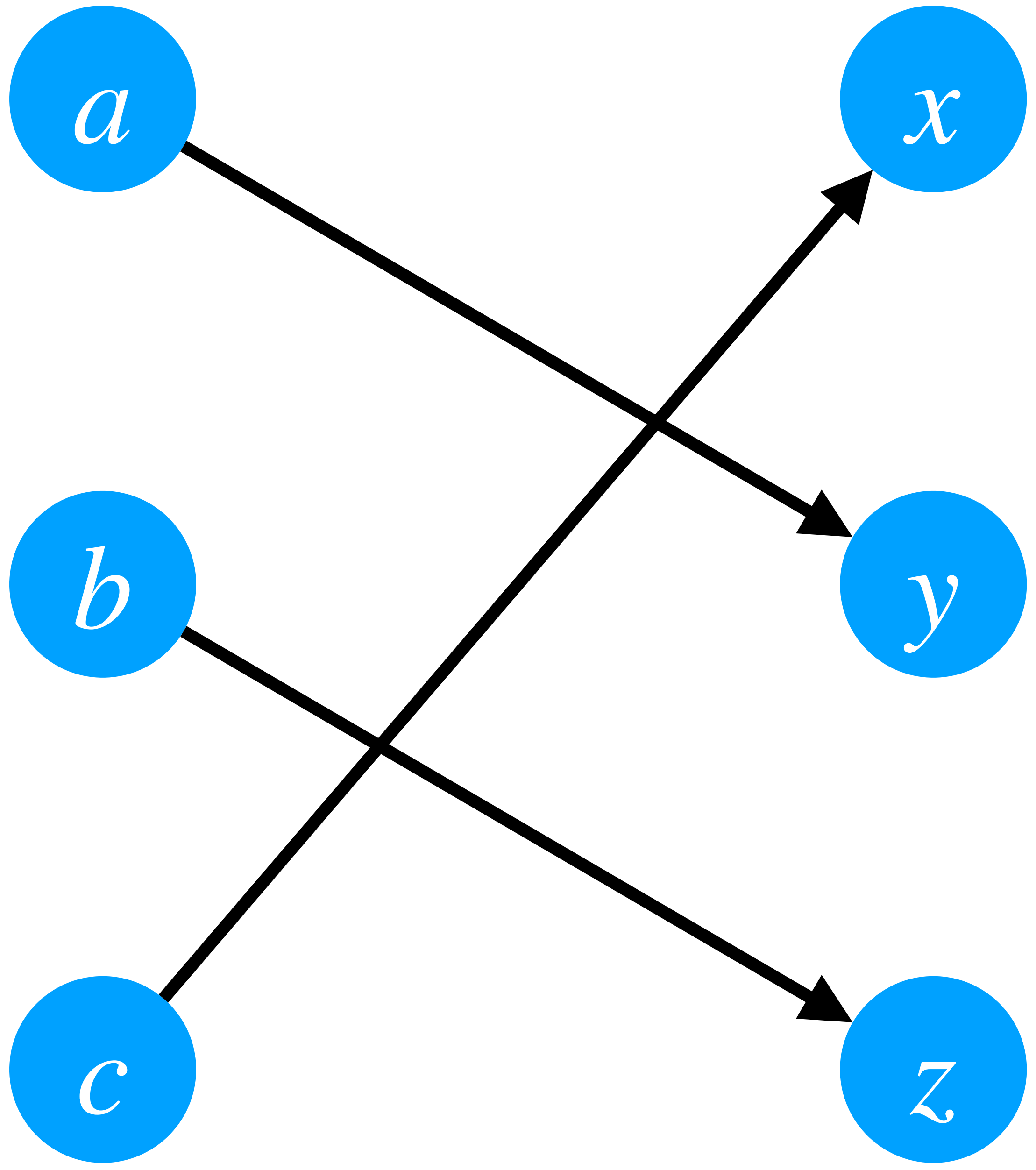
No mechanism is stable and strategyproof for both sides.

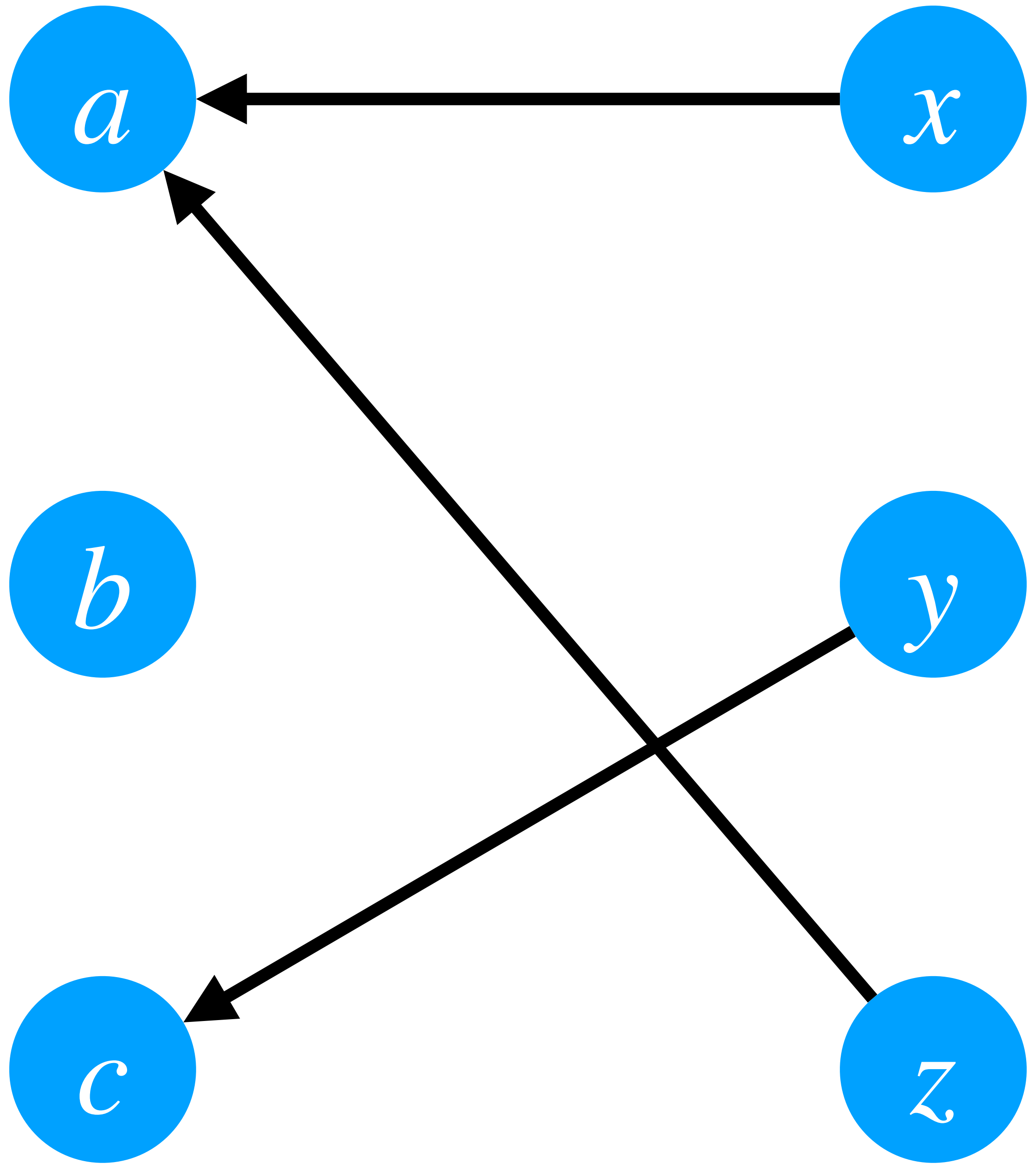
- proof idea:
  - instance with only two stable matchings
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a x \succ_a z$ $\quad$ $a \succ_x c \succ_x b$

$x \succ_b y \succ_b z$ $\quad$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad$ $a \succ_z b \succ_z c$





Strategic Behavior

- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

Theorem [Roth '82]

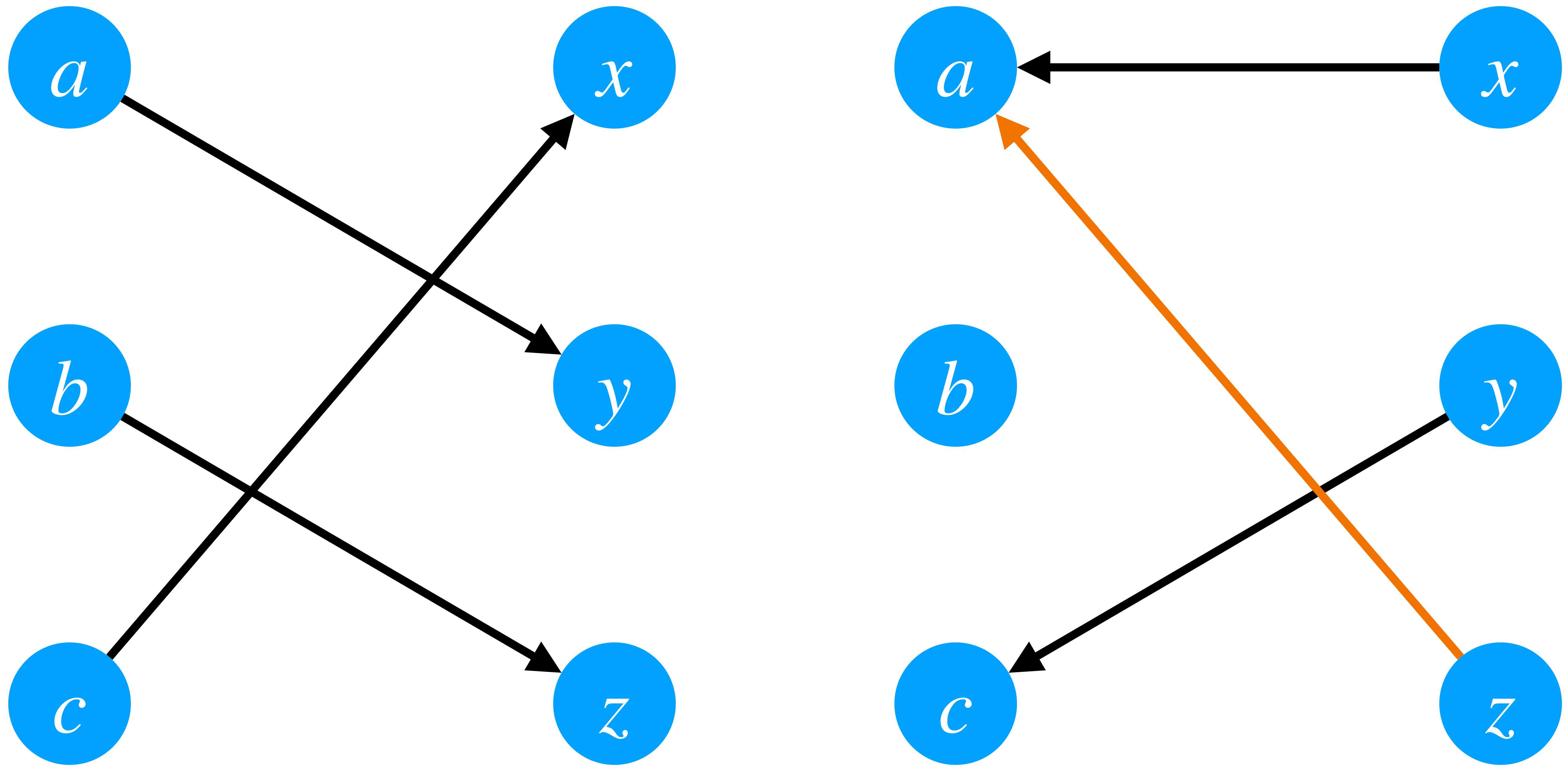
No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a x \succ_a z$ $\quad$ $a \succ_x c \succ_x b$

$x \succ_b y \succ_b z$ $\quad$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad$ $a \succ_z b \succ_z c$



- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

- **Theorem** [Roth '82]

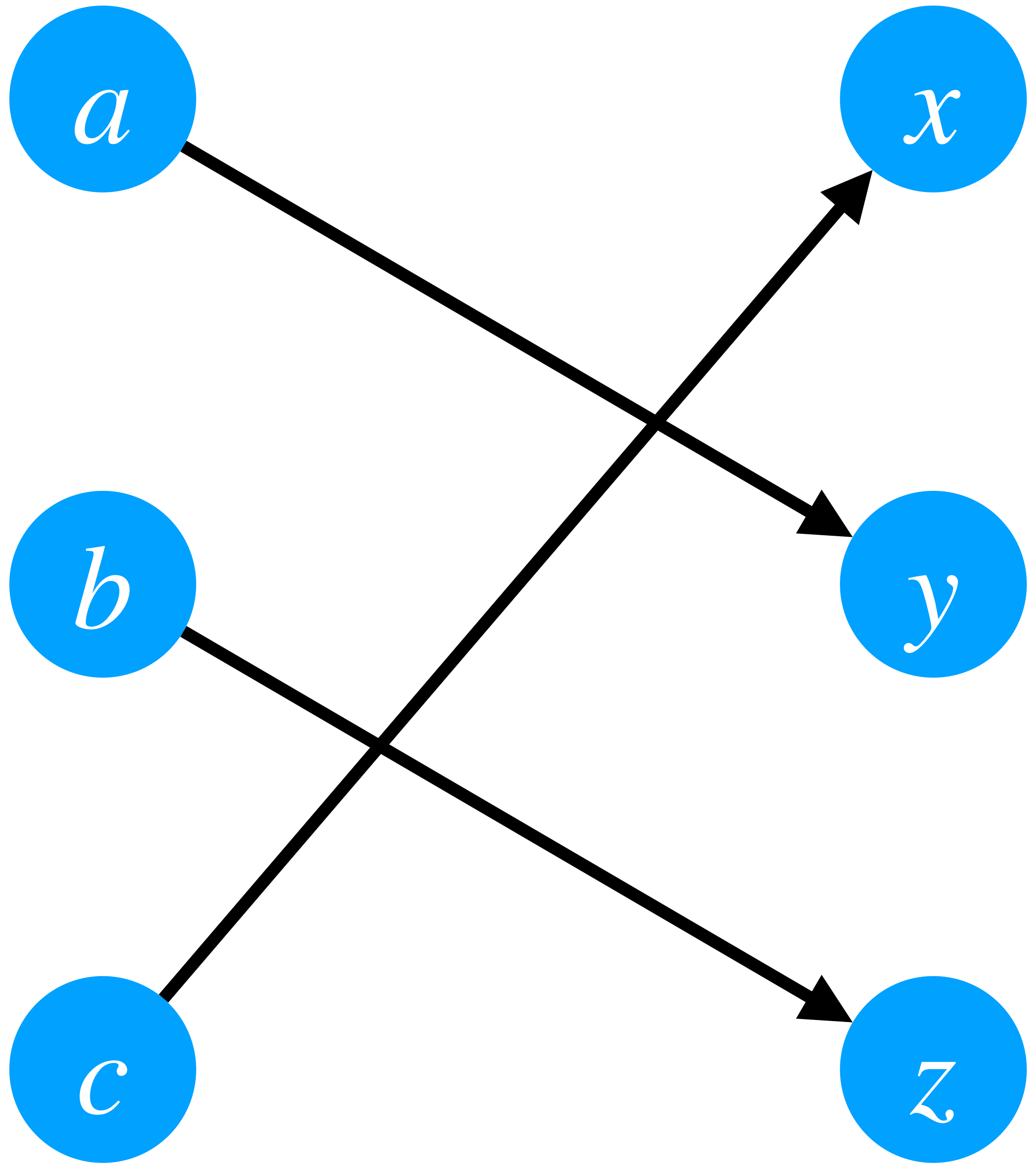
  No mechanism is stable and strategyproof for both sides.

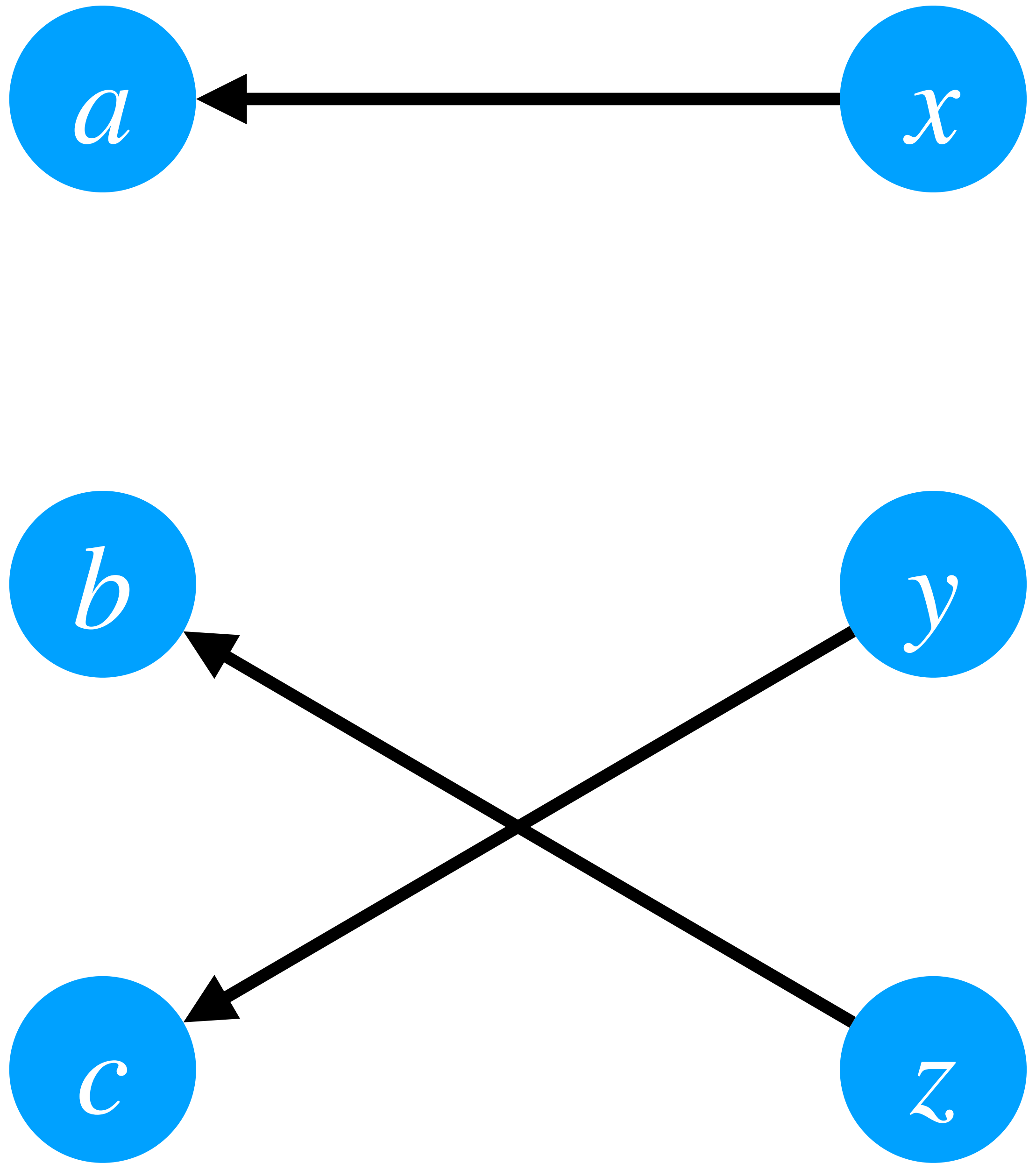
- proof idea:
  - instance with only two stable matchings
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a x \succ_a z$ $\quad$ $a \succ_x c \succ_x b$

$x \succ_b y \succ_b z$ $\quad$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad$ $a \succ_z b \succ_z c$





Strategic Behavior

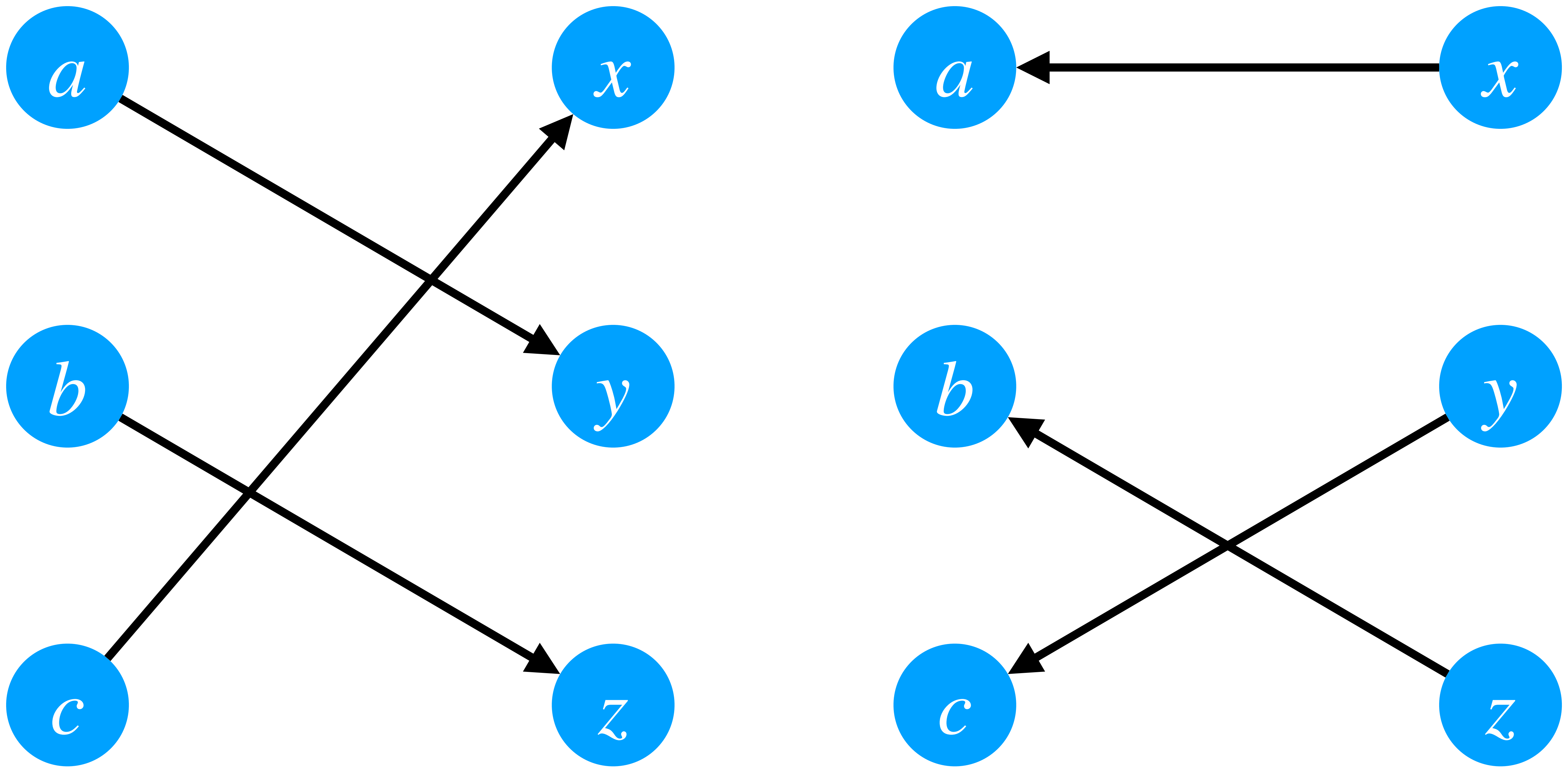
- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

$$y \succ_a x \succ_a z \qquad a \succ_x c \succ_x b$$
$$x \succ_b y \succ_b z \qquad c \succ_y a \succ_y b$$
$$x \succ_c y \succ_c z \qquad a \succ_z b \succ_z c$$

Theorem [Roth ’82]

No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings (why?)
  - for any of them, an agent can misreport and force the other (better) one



Strategic Behavior

- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

Theorem [Roth '82]

No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings (why?)
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a x \succ_a z$

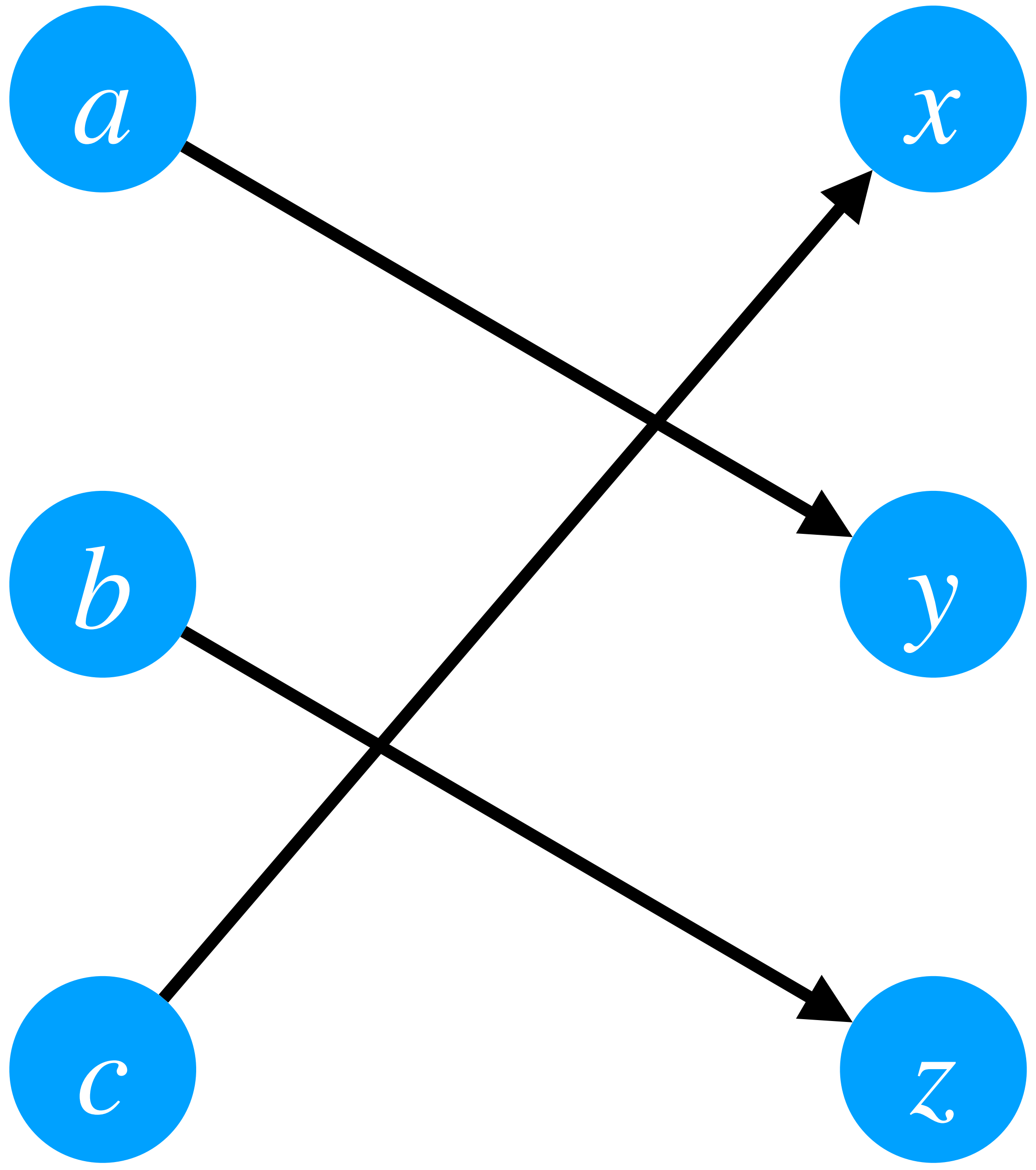
$x \succ_b y \succ_b z$

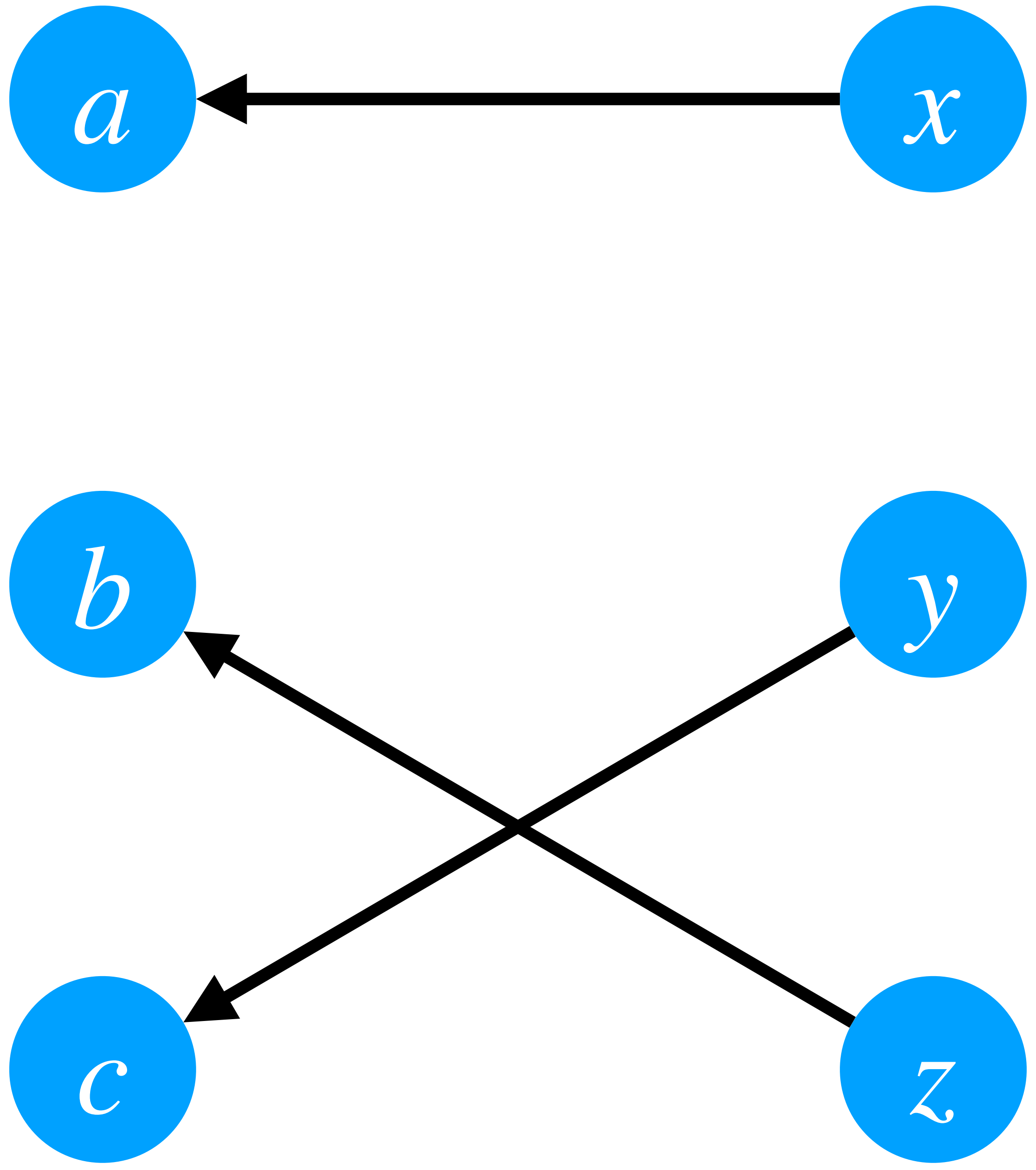
$x \succ_c y \succ_c z$

$a \succ_x c \succ_x b$ $a \succ_x b \succ_x c$

$c \succ_y a \succ_y b$

$a \succ_z b \succ_z c$





- Is DA strategyproof?
  yes (proposing side) and
  no (proposed side)

Theorem [Roth '82]

No mechanism is stable and strategyproof for both sides.

- proof idea:
  - instance with only two stable matchings (why?)
  - for any of them, an agent can misreport and force the other (better) one

$y \succ_a z \succ_a x$ $y \succ_a x \succ_a z$

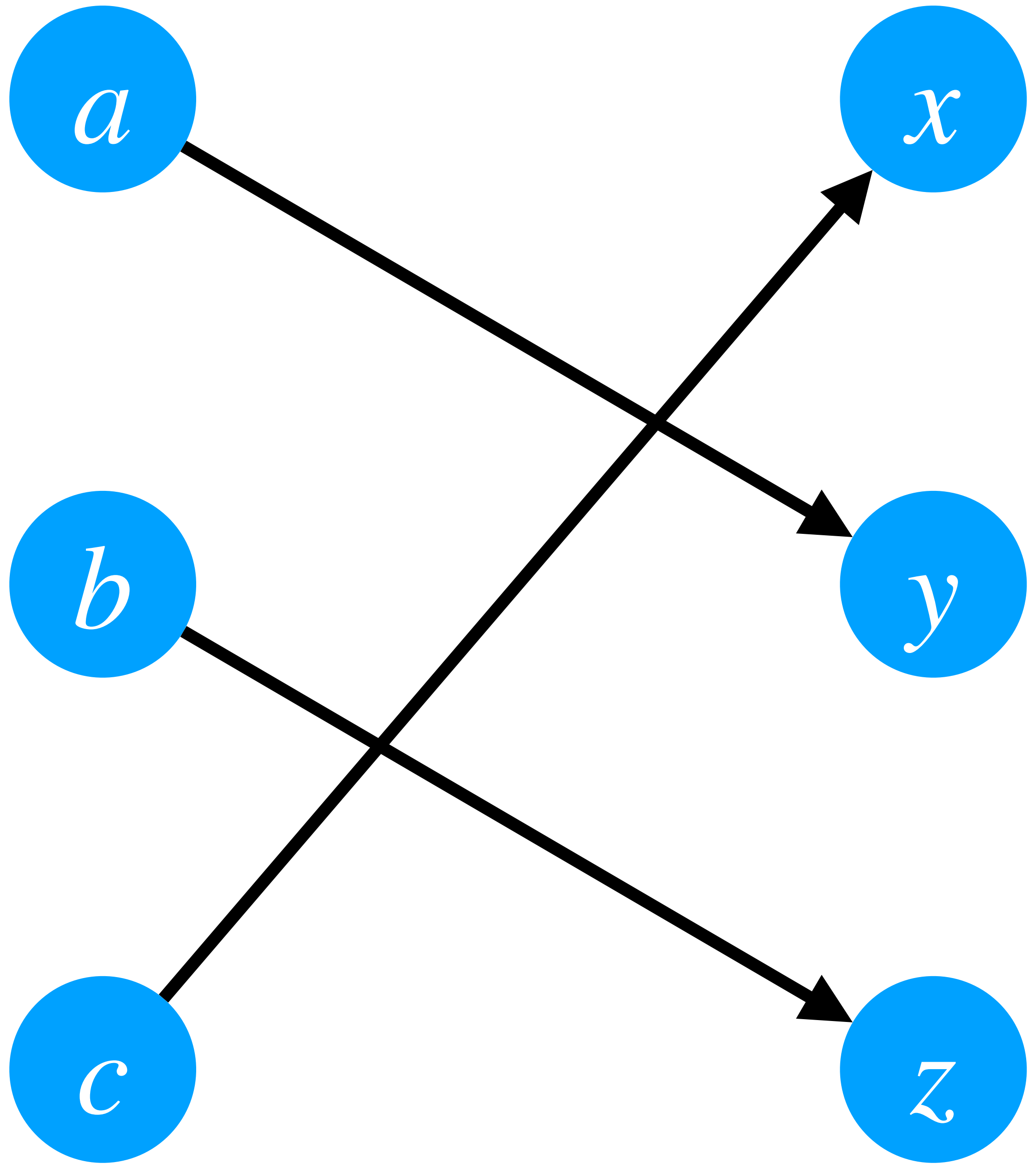
$x \succ_b y \succ_b z$

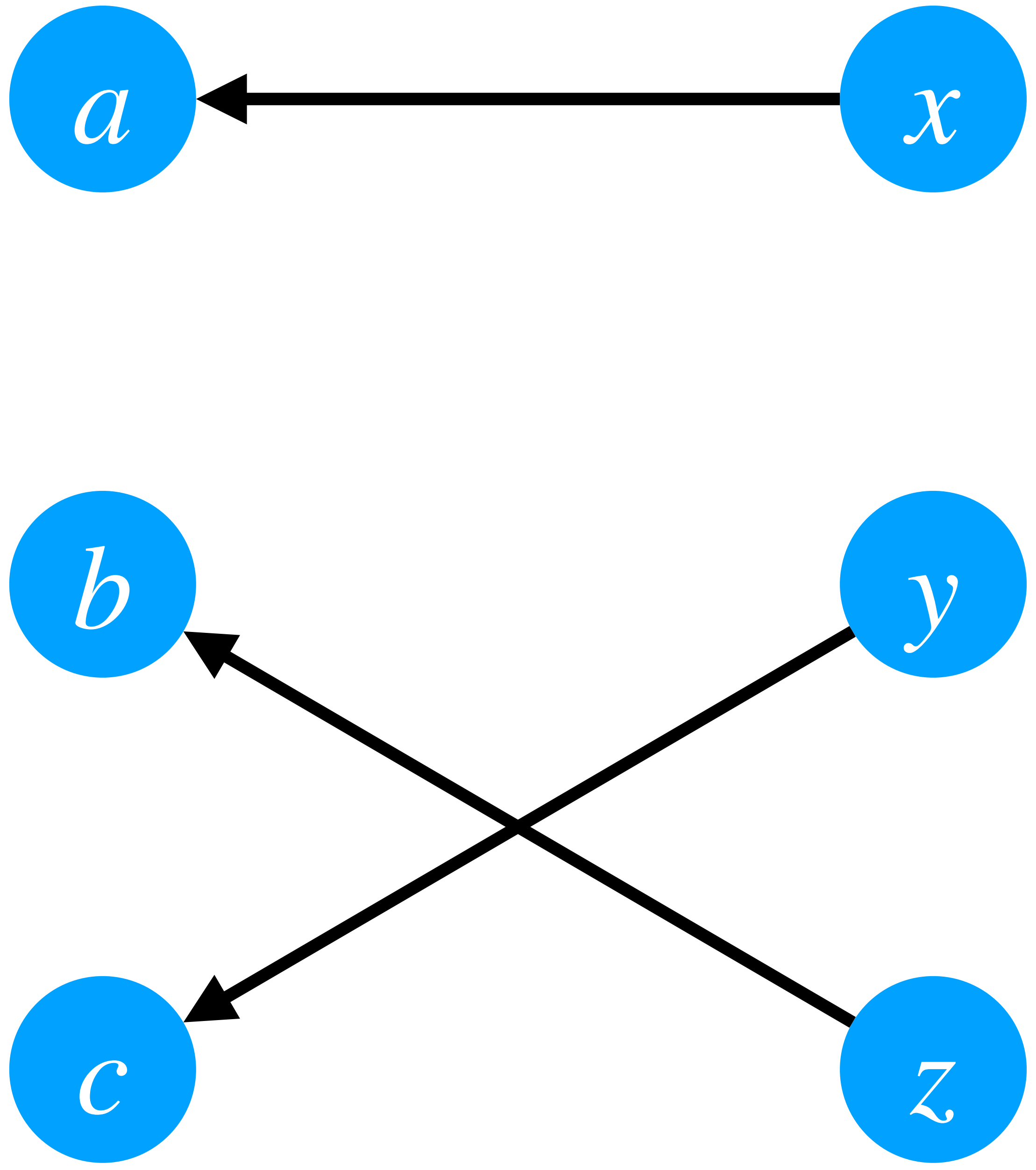
$x \succ_c y \succ_c z$

$a \succ_x c \succ_x b$

$c \succ_y a \succ_y b$

$a \succ_z b \succ_z c$





Efficiency

- a matching is **Pareto-optimal** for one side
  if no alternative matching is

Efficiency

- a matching is **Pareto-optimal** for one side
  if no alternative matching is
  - weakly better for everyone from this side

Efficiency

- a matching is **Pareto-optimal** for one side
  if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

- a matching is **Pareto-optimal** for one side
  if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

Theorem [Roth ’82]

No mechanism is stable and Pareto-optimal
for any side.

Efficiency

- a matching is **Pareto-optimal** for one side
  if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

Theorem [Roth '82]

No mechanism is stable and Pareto-optimal
for any side.

- proof idea:
  - instance where DA is not Pareto-optimal
    for the proposing side

- a matching is **Pareto-optimal** for one side
  if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

Theorem [Roth '82]

No mechanism is stable and Pareto-optimal
for any side.

- proof idea:
  - instance where DA is not Pareto-optimal
    for the proposing side
  - the outcome of DA Pareto-dominates all other
    stable matchings for the proposing side

Efficiency

- a matching is **Pareto-optimal** for one side if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

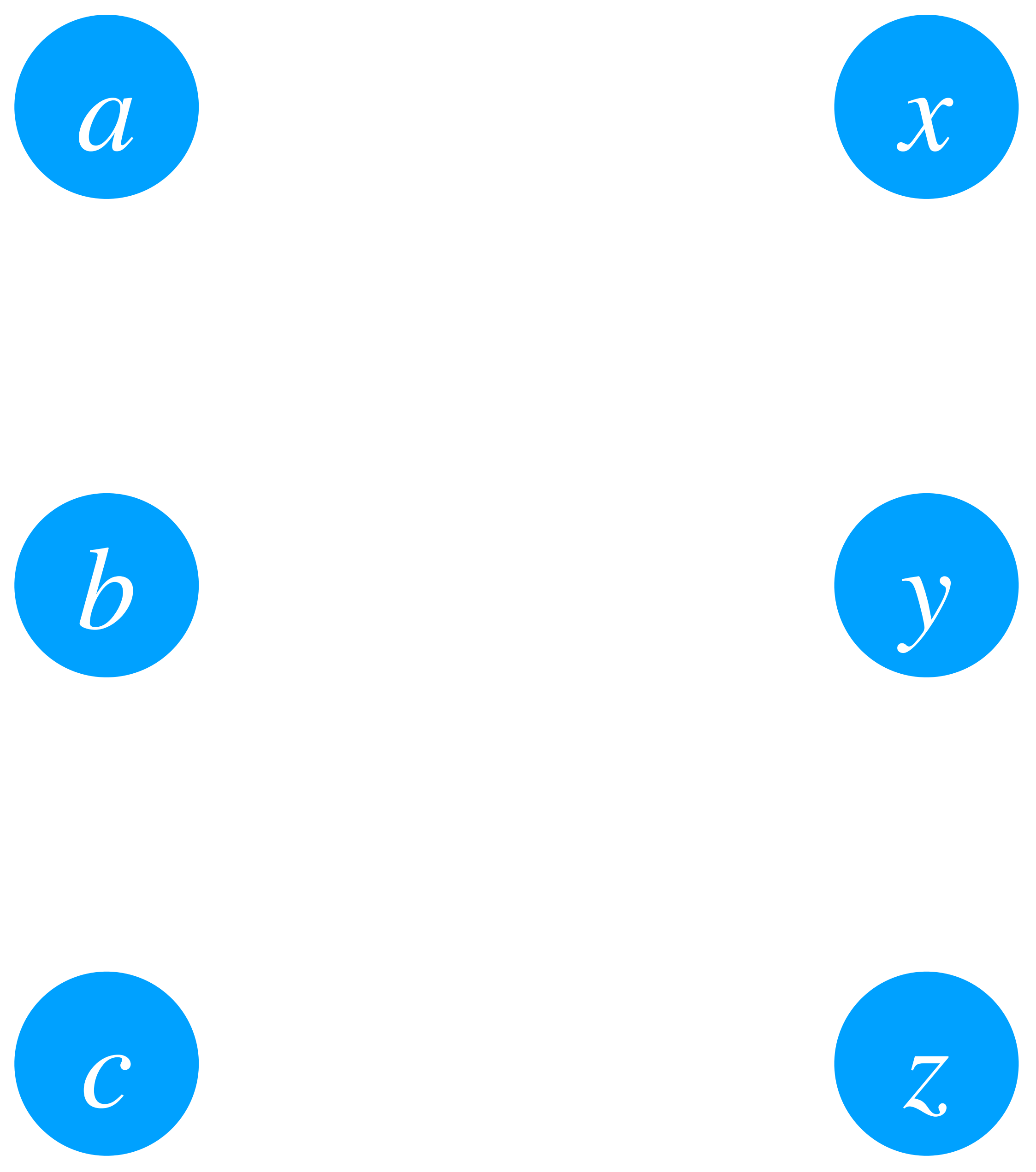
$y \succ_a x \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b y \succ_b z$ $\quad$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad$ $a \succ_z b \succ_z c$

Theorem [Roth '82]

No mechanism is stable and Pareto-optimal for any side.



- proof idea:
  - instance where DA is not Pareto-optimal for the proposing side
  - the outcome of DA Pareto-dominates all other stable matchings for the proposing side

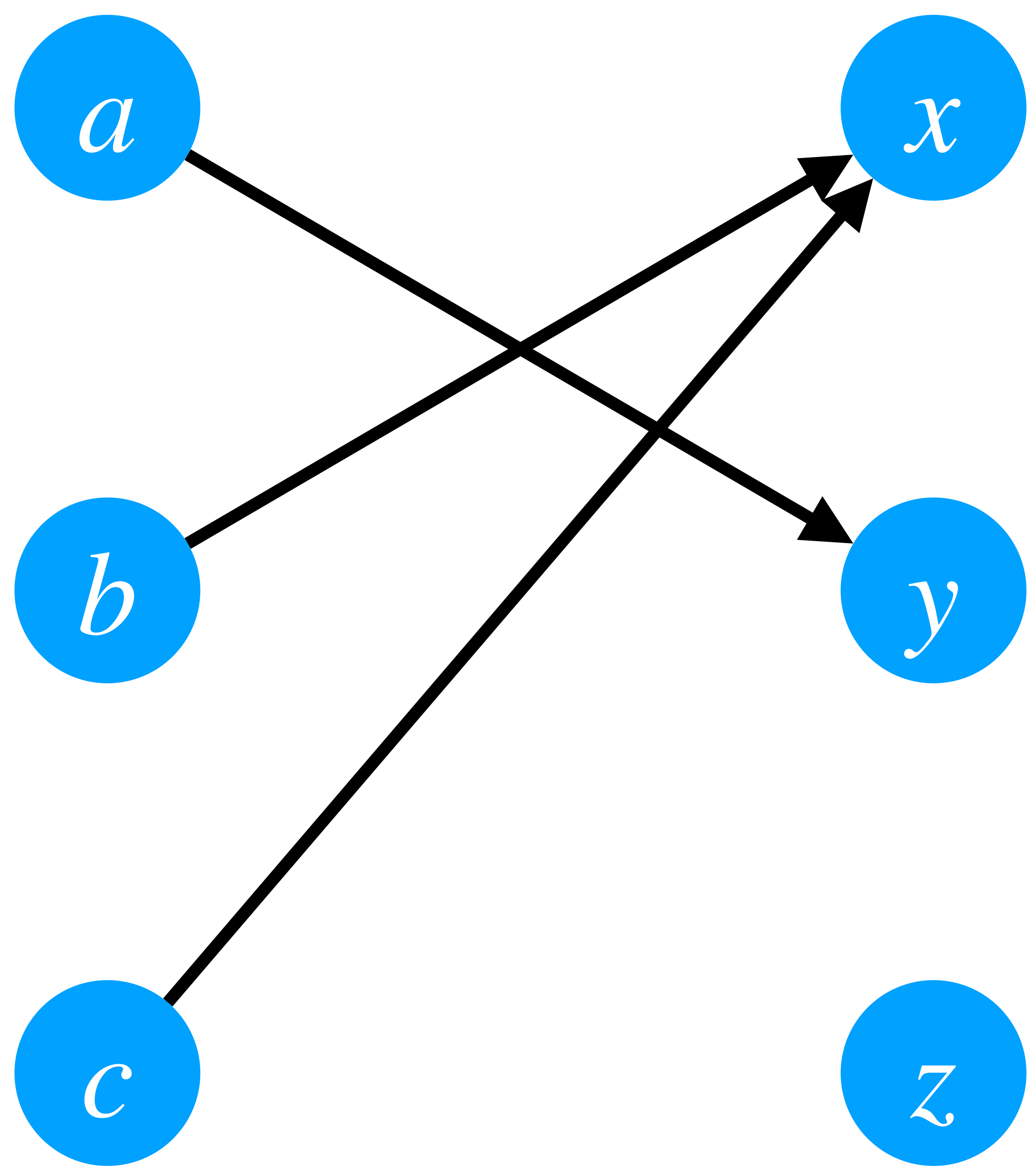
- a matching is **Pareto-optimal** for one side if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

$$y \succ_a x \succ_a z \qquad a \succ_x b \succ_x c$$
$$x \succ_b y \succ_b z \qquad c \succ_y a \succ_y b$$
$$x \succ_c y \succ_c z \qquad a \succ_z b \succ_z c$$

Theorem [Roth '82]

No mechanism is stable and Pareto-optimal for any side.

- proof idea:
  - instance where DA is not Pareto-optimal for the proposing side
  - the outcome of DA Pareto-dominates all other stable matchings for the proposing side



- a matching is **Pareto-optimal** for one side
  if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

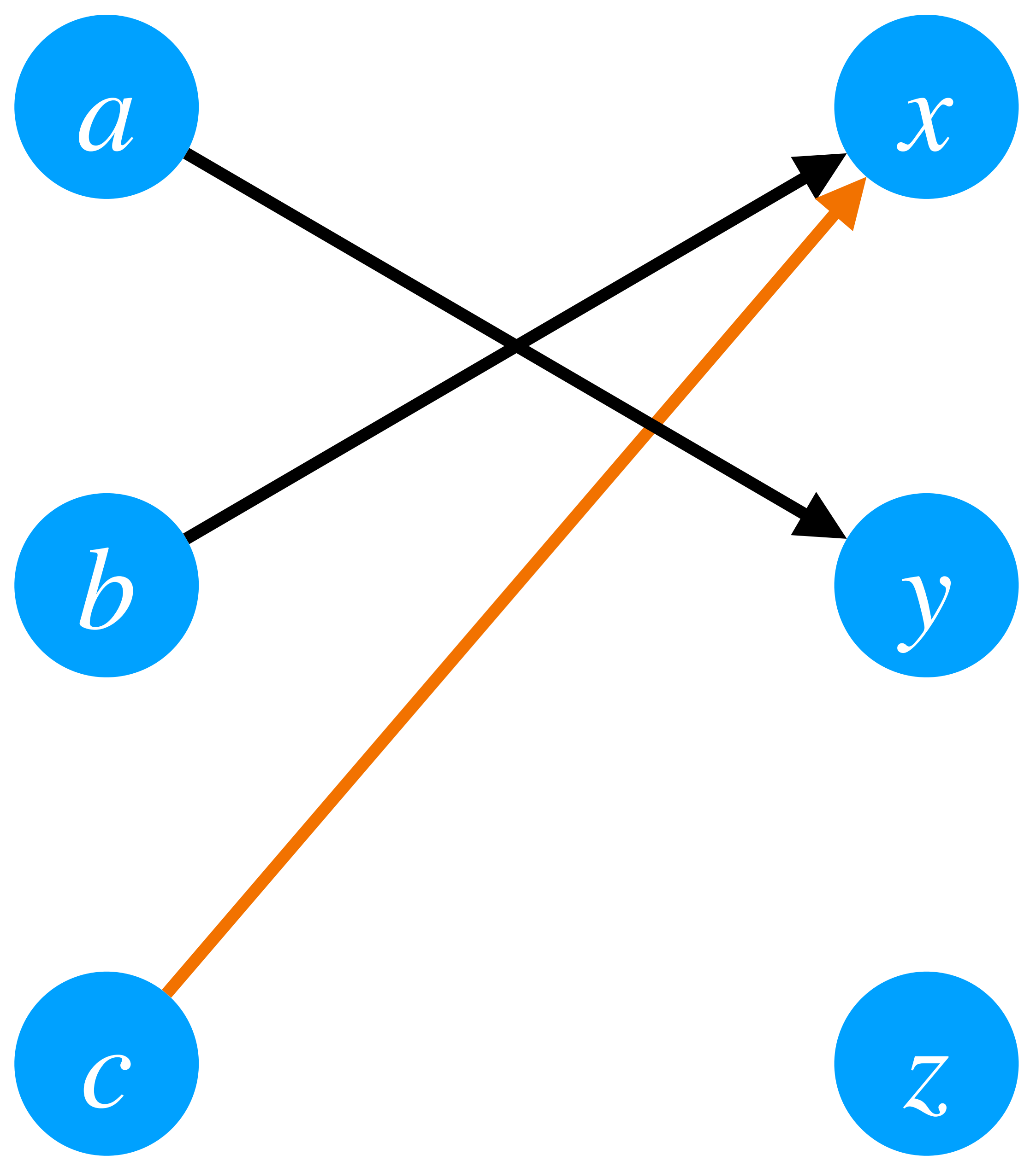
$y \succ_a x \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b y \succ_b z$ $\quad$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad$ $a \succ_z b \succ_z c$

- **Theorem** [Roth ’82]

  No mechanism is stable and Pareto-optimal
  for any side.



- proof idea:
  - instance where DA is not Pareto-optimal
    for the proposing side
  - the outcome of DA Pareto-dominates all other
    stable matchings for the proposing side

Efficiency

- a matching is **Pareto-optimal** for one side
  if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

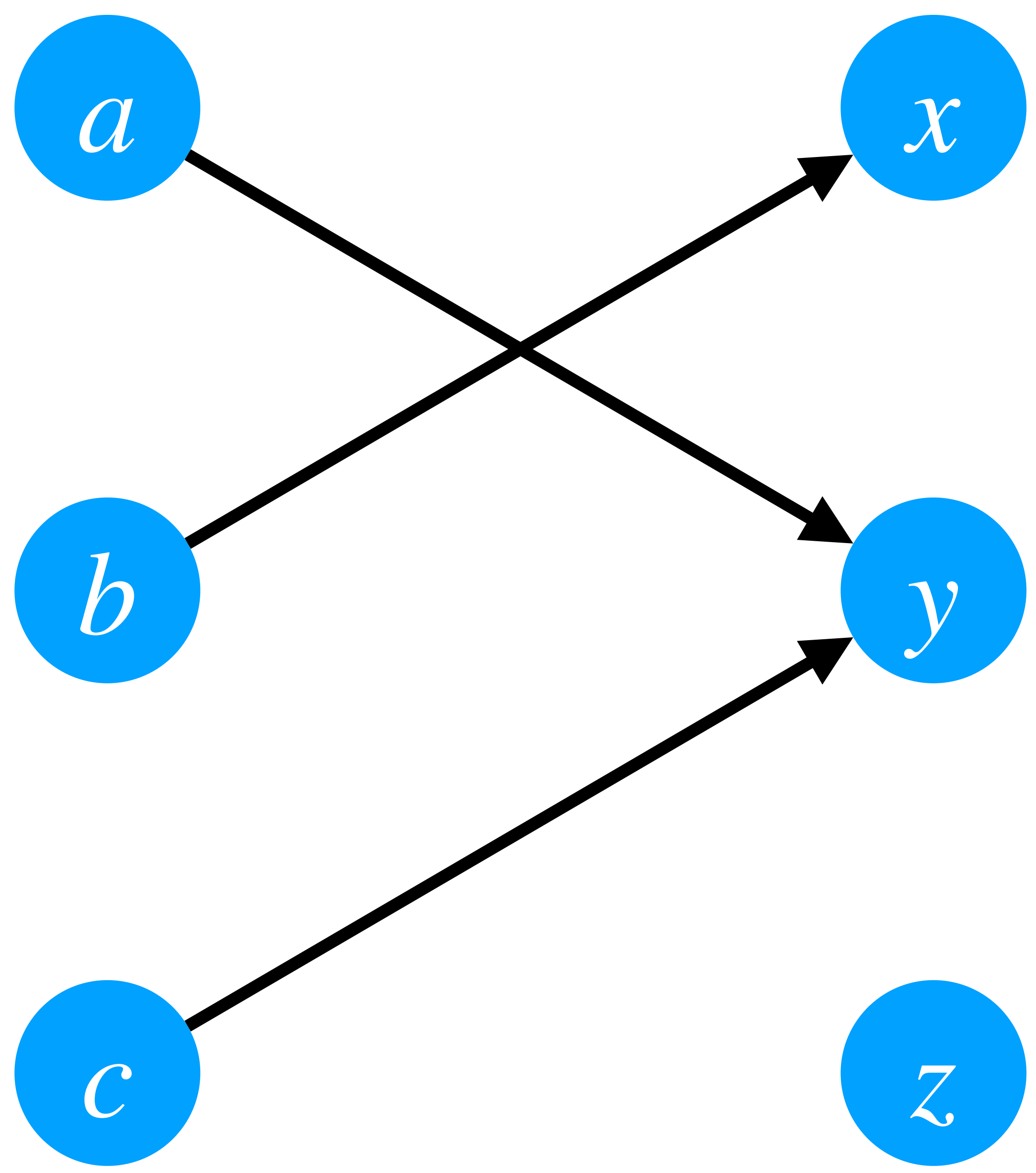
$y \succ_a x \succ_a z \qquad a \succ_x b \succ_x c$

$x \succ_b y \succ_b z \qquad c \succ_y a \succ_y b$

$x \succ_c y \succ_c z \qquad a \succ_z b \succ_z c$

- **Theorem** [Roth ’82]

  No mechanism is stable and Pareto-optimal
  for any side.



- proof idea:
  - instance where DA is not Pareto-optimal
    for the proposing side
  - the outcome of DA Pareto-dominates all other
    stable matchings for the proposing side

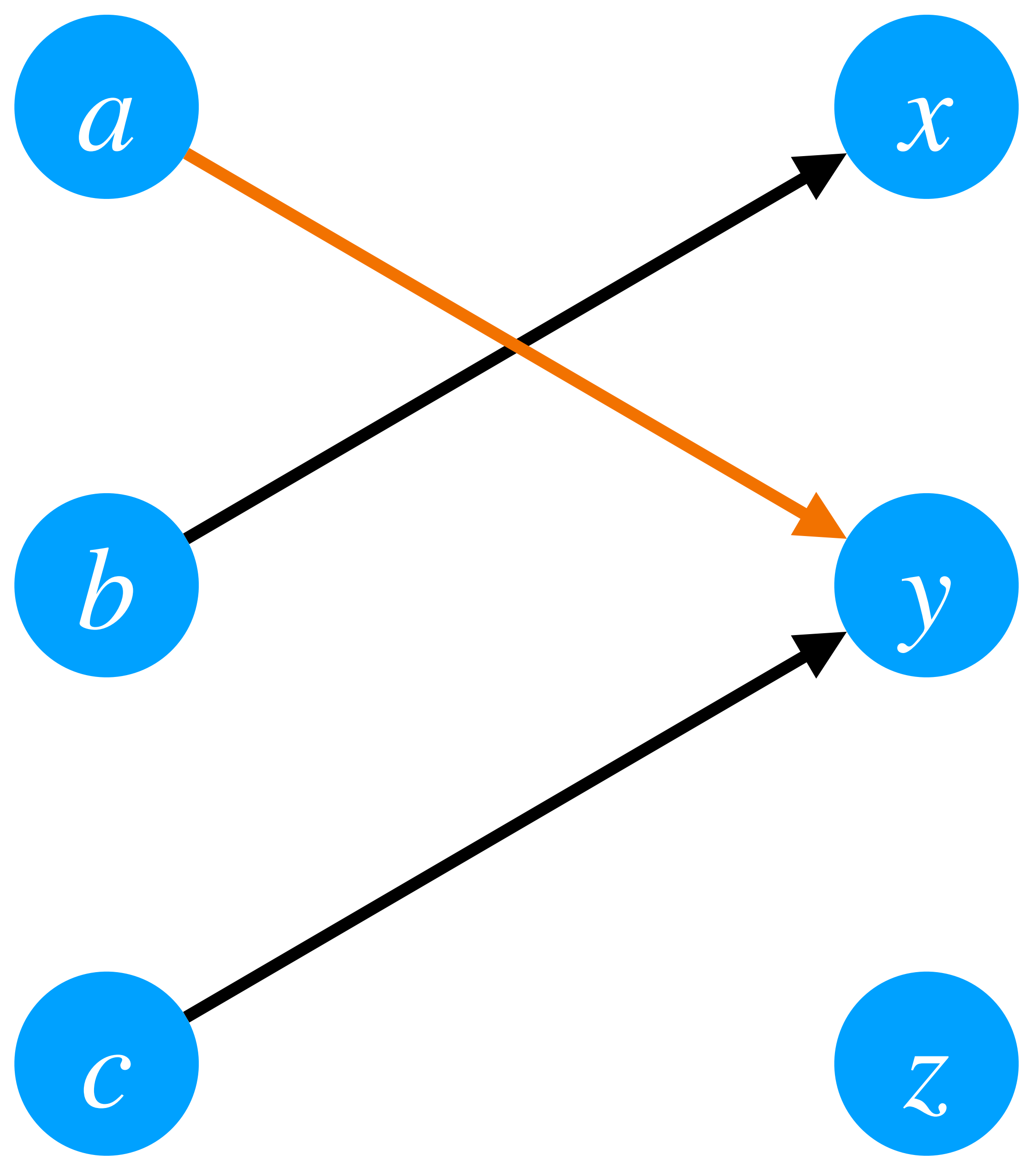
- a matching is **Pareto-optimal** for one side if no alternative matching is
    - weakly better for everyone from this side
    - strictly better for someone from this side

$$y \succ_a x \succ_a z \qquad a \succ_x b \succ_x c$$
$$x \succ_b y \succ_b z \qquad c \succ_y a \succ_y b$$
$$x \succ_c y \succ_c z \qquad a \succ_z b \succ_z c$$

Theorem [Roth '82]

No mechanism is stable and Pareto-optimal for any side.

- proof idea:
    - instance where DA is not Pareto-optimal for the proposing side
    - the outcome of DA Pareto-dominates all other stable matchings for the proposing side



- a matching is **Pareto-optimal** for one side
  if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

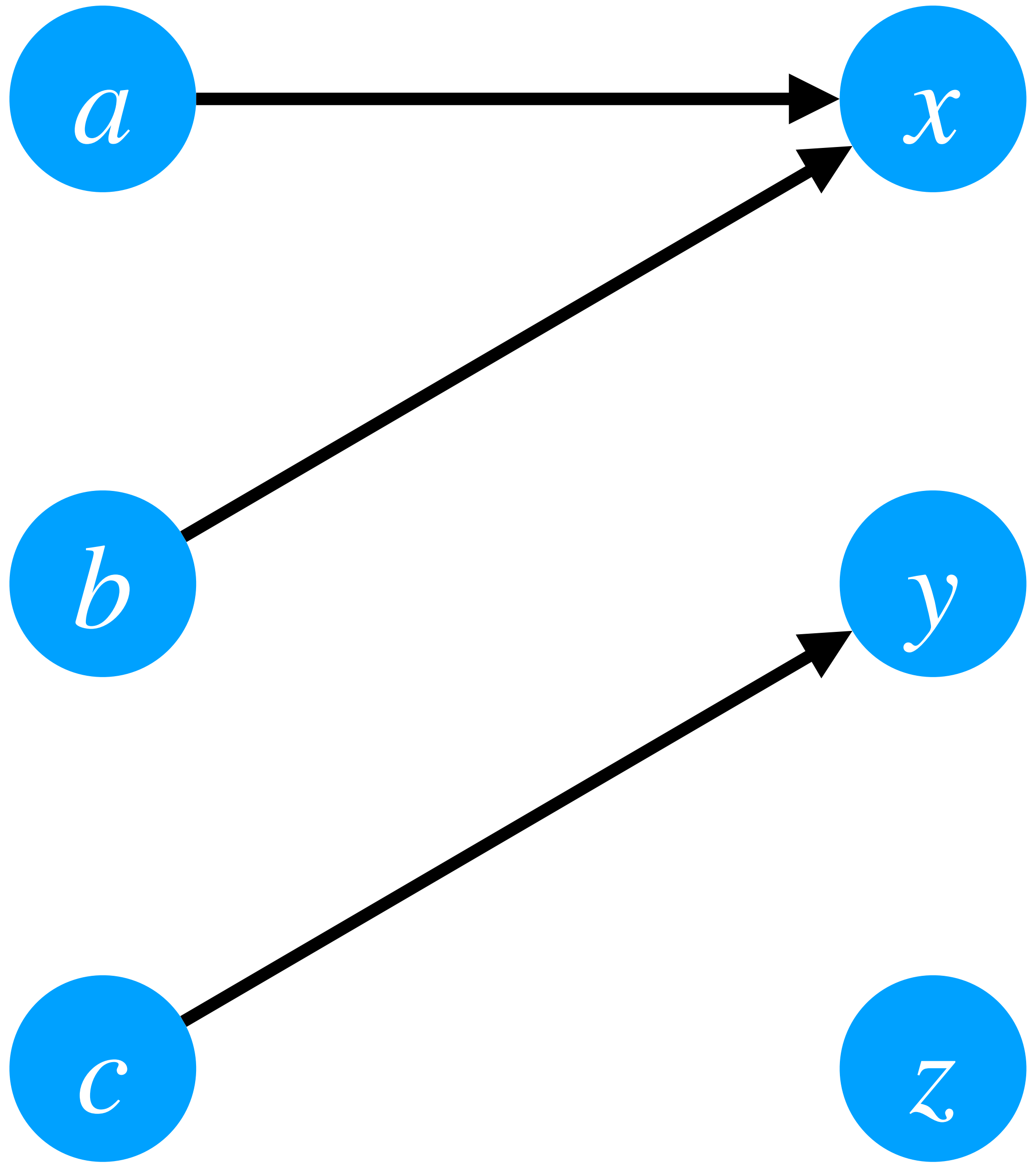
$y \succ_a x \succ_a z$ $\quad a \succ_x b \succ_x c$

$x \succ_b y \succ_b z$ $\quad c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad a \succ_z b \succ_z c$

Theorem [Roth ’82]

No mechanism is stable and Pareto-optimal for any side.



- proof idea:
  - instance where DA is not Pareto-optimal for the proposing side
  - the outcome of DA Pareto-dominates all other stable matchings for the proposing side

Efficiency

- a matching is **Pareto-optimal** for one side if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

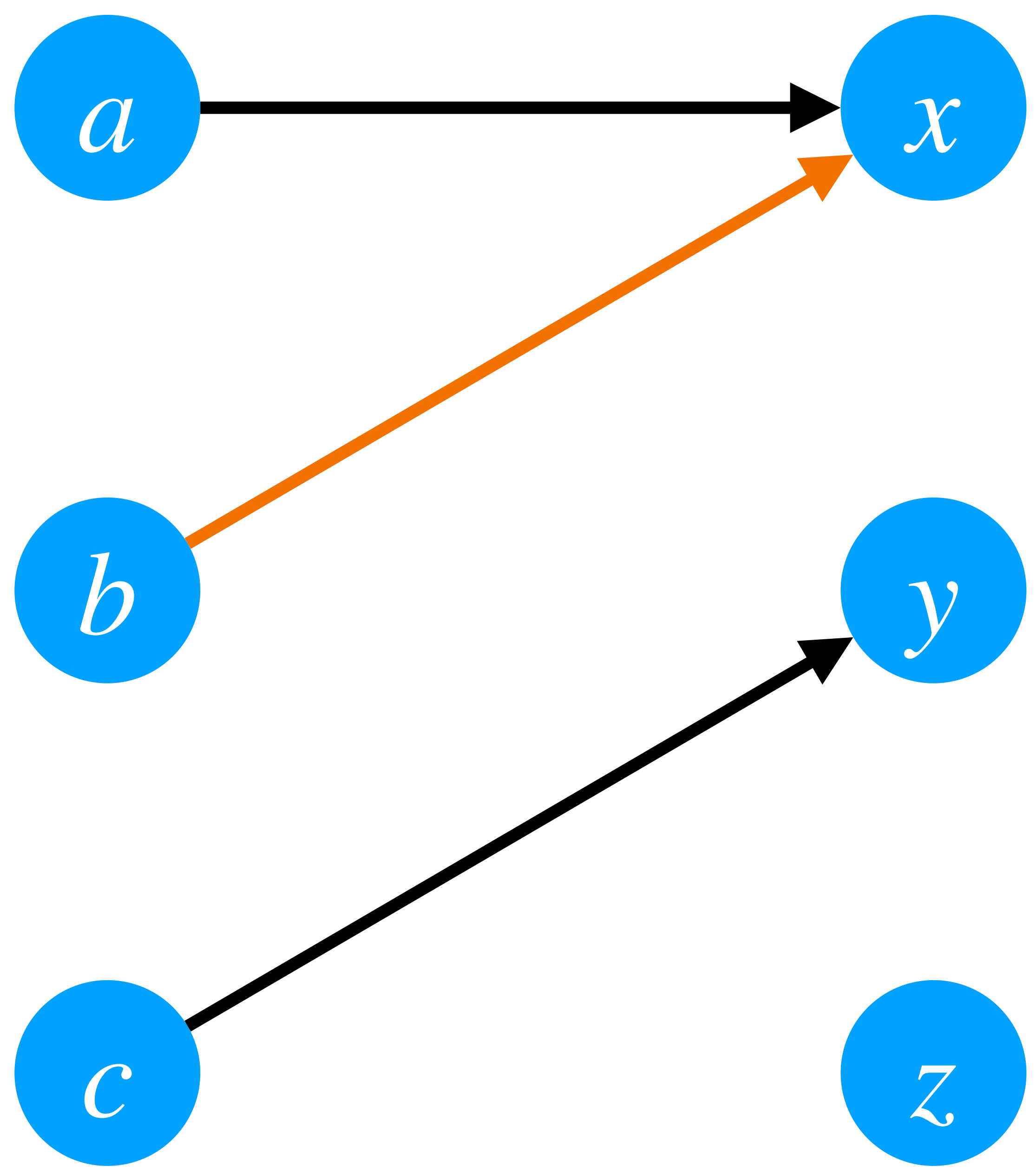
$y \succ_a x \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b y \succ_b z$ $\quad$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad$ $a \succ_z b \succ_z c$

Theorem [Roth '82]

No mechanism is stable and Pareto-optimal for any side.



- proof idea:
  - instance where DA is not Pareto-optimal for the proposing side
  - the outcome of DA Pareto-dominates all other stable matchings for the proposing side

Efficiency

- a matching is **Pareto-optimal** for one side if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

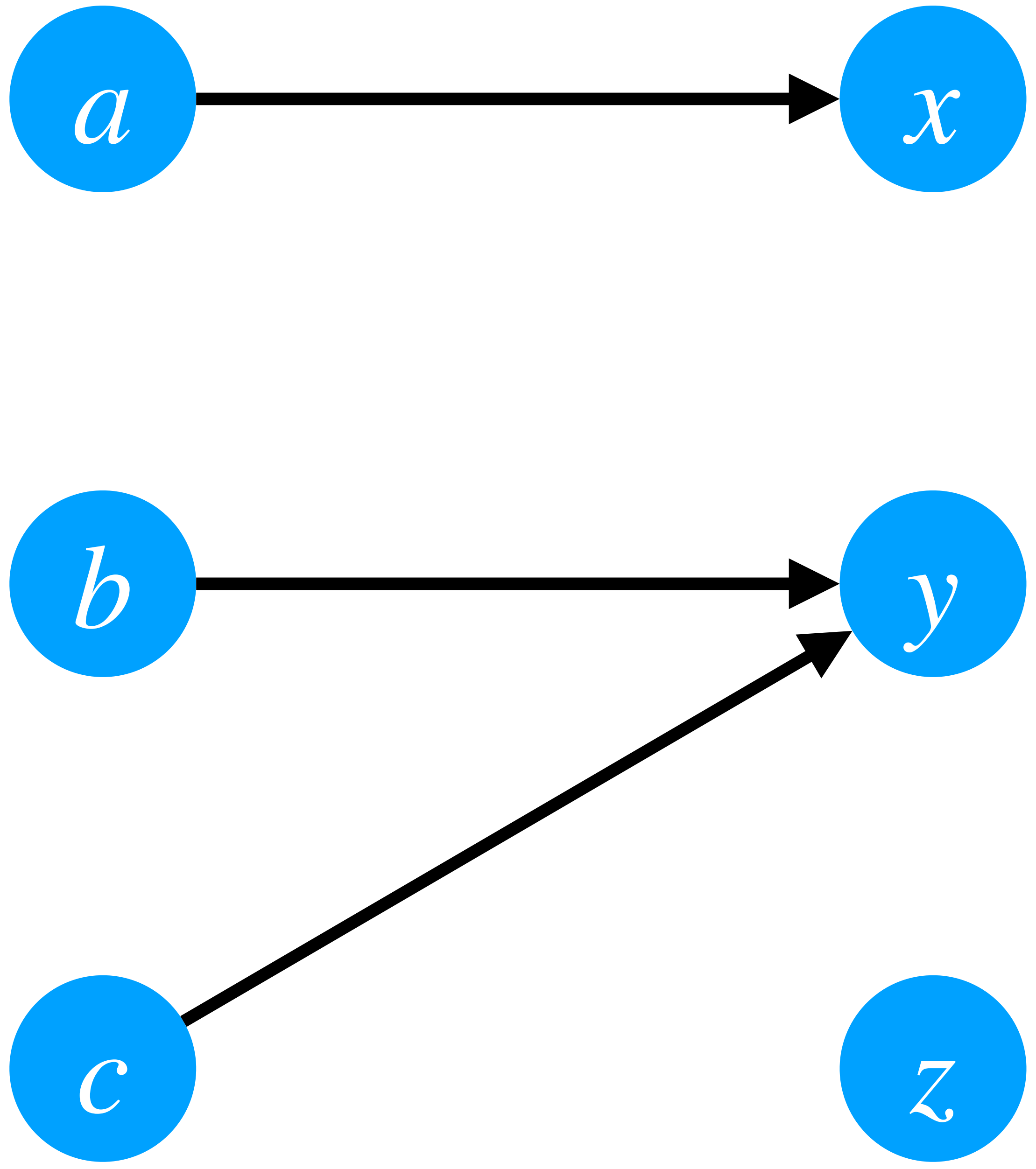
$y \succ_a x \succ_a z$ $\quad a \succ_x b \succ_x c$

$x \succ_b y \succ_b z$ $\quad c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad a \succ_z b \succ_z c$

Theorem [Roth '82]

No mechanism is stable and Pareto-optimal for any side.



- proof idea:
  - instance where DA is not Pareto-optimal for the proposing side
  - the outcome of DA Pareto-dominates all other stable matchings for the proposing side

Efficiency

- a matching is **Pareto-optimal** for one side if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

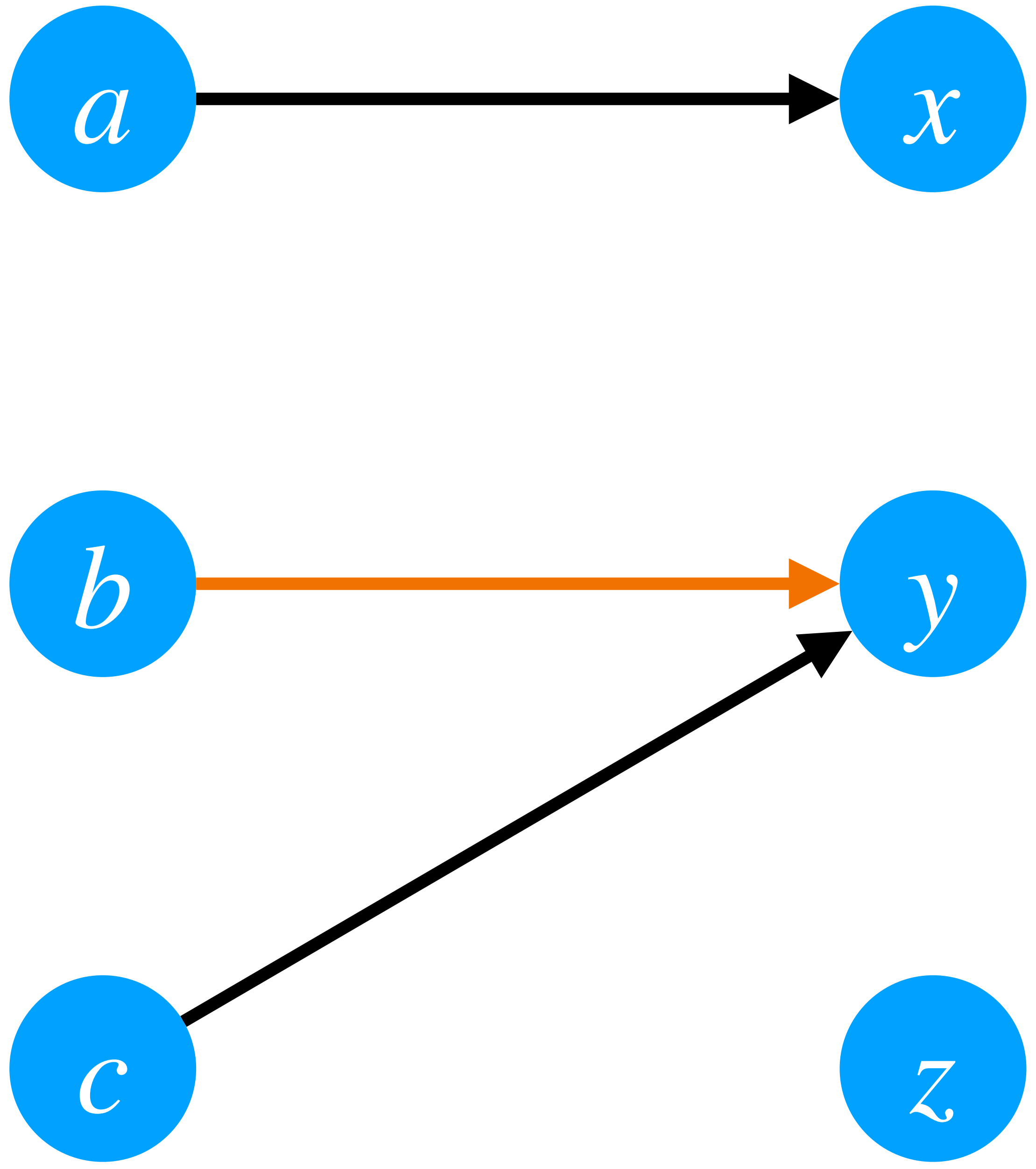
$y \succ_a x \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b y \succ_b z$ $\quad$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad$ $a \succ_z b \succ_z c$

Theorem [Roth ’82]

No mechanism is stable and Pareto-optimal for any side.



- proof idea:
  - instance where DA is not Pareto-optimal for the proposing side
  - the outcome of DA Pareto-dominates all other stable matchings for the proposing side

Efficiency

- a matching is **Pareto-optimal** for one side
  if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

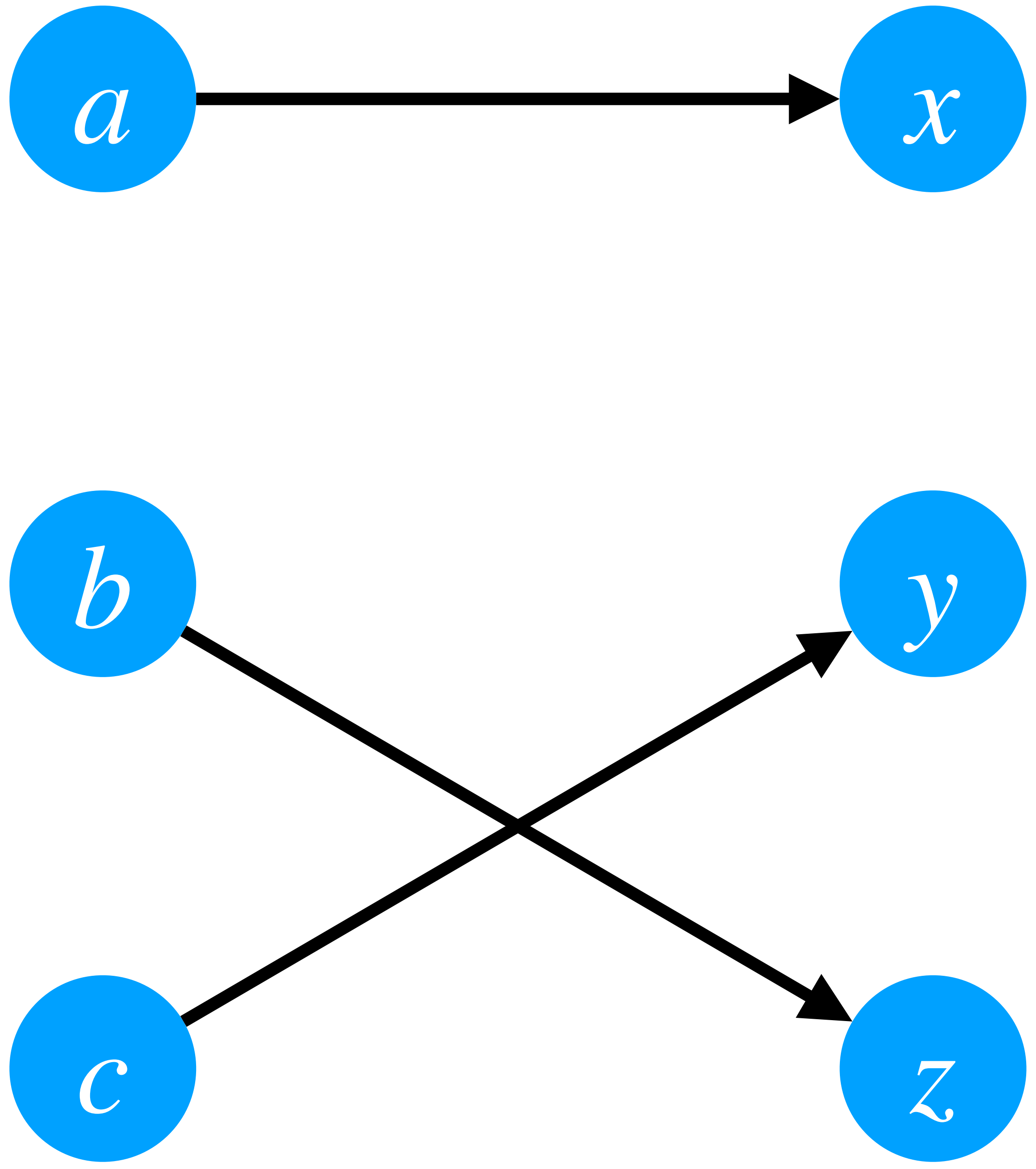
$y \succ_a x \succ_a z$ $\quad$ $a \succ_x b \succ_x c$

$x \succ_b y \succ_b z$ $\quad$ $c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\quad$ $a \succ_z b \succ_z c$

Theorem [Roth ’82]

No mechanism is stable and Pareto-optimal
for any side.



- proof idea:
  - instance where DA is not Pareto-optimal
    for the proposing side
  - the outcome of DA Pareto-dominates all other
    stable matchings for the proposing side

Efficiency

- a matching is **Pareto-optimal** for one side if no alternative matching is
  - weakly better for everyone from this side
  - strictly better for someone from this side

Theorem [Roth '82]

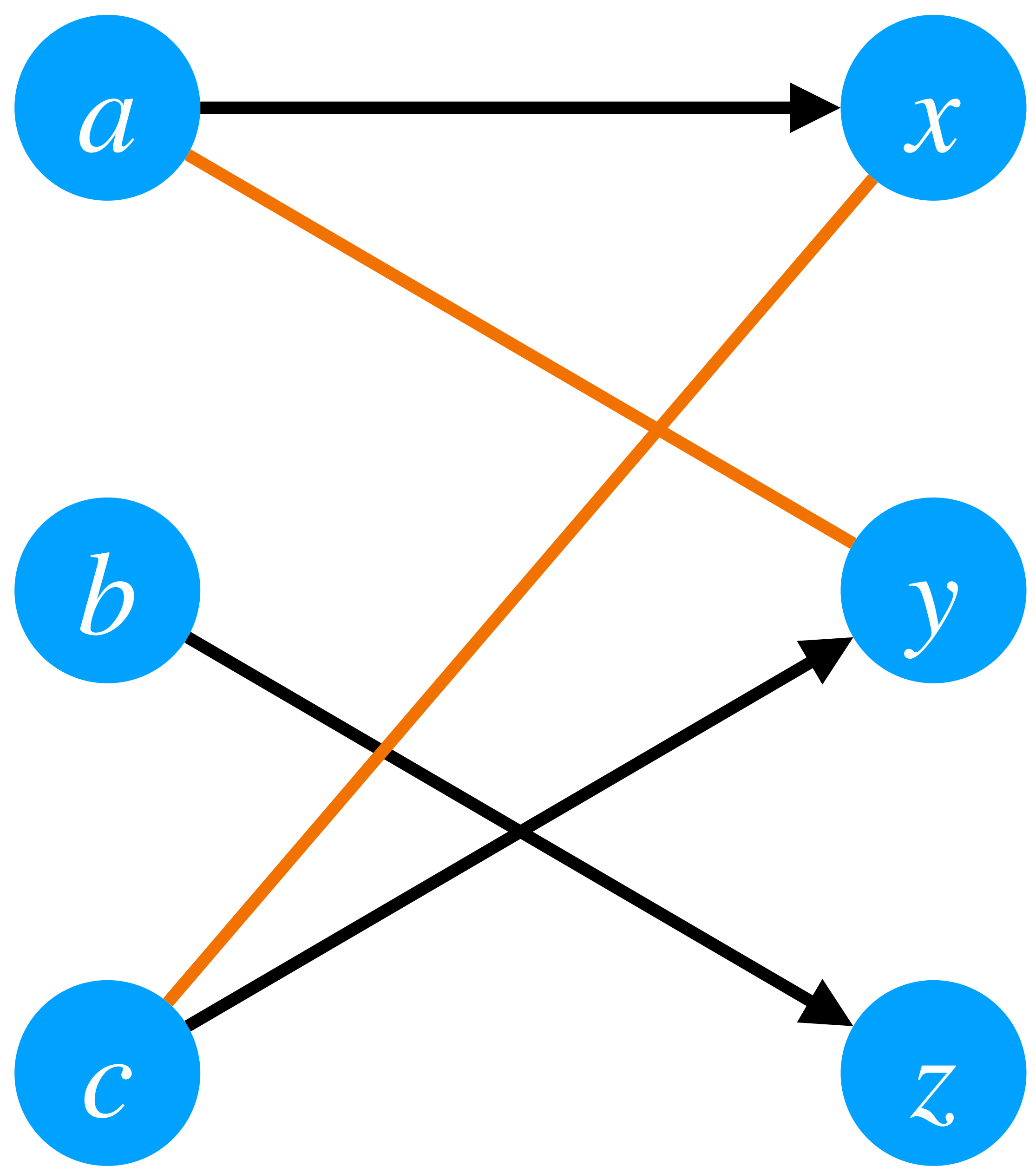
No mechanism is stable and Pareto-optimal for any side.

- proof idea:
  - instance where DA is not Pareto-optimal for the proposing side
  - the outcome of DA Pareto-dominates all other stable matchings for the proposing side

$y \succ_a x \succ_a z$ $\qquad a \succ_x b \succ_x c$

$x \succ_b y \succ_b z$ $\qquad c \succ_y a \succ_y b$

$x \succ_c y \succ_c z$ $\qquad a \succ_z b \succ_z c$



Lattice of Stable Matchings

- **partial order** $\leq$ over stable matchings $\mathscr{M}$ defined by $M \leq M' \Leftrightarrow$ all applicants weakly prefer M' over M

Lattice of Stable Matchings

- **partial order** $\leq$ over stable matchings $\mathscr{M}$ defined by $M \leq M' \Leftrightarrow$ all applicants weakly prefer M' over M

- $(\mathscr{M}, \leq)$ is a (distributive) **lattice**
  see Gale, Sotomayor '85 for a simple proof

Lattice of Stable Matchings

- **partial order** $\leq$ over stable matchings $\mathscr{M}$ defined by $M \leq M' \Leftrightarrow$ all applicants weakly prefer M' over M
- $(\mathscr{M}, \leq)$ is a (distributive) **lattice**

  see Gale, Sotomayor ’85 for a simple proof

  - in $M \vee M'$, each applicant is matched to their most-preferred company among M, M'

A *lattice* is a partially ordered set $(L, \leq)$ s.t. any pair of elements $a, b \in L$ have a least upper bound $a \vee b$ in L and a greatest lower bound $a \wedge b$ in L.

Lattice of Stable Matchings

- **partial order** $\leq$ over stable matchings $\mathscr{M}$ defined by $M \leq M' \Leftrightarrow$ all applicants weakly prefer M' over M
- $(\mathscr{M}, \leq)$ is a (distributive) **lattice**
  see Gale, Sotomayor '85 for a simple proof
  - in $M \vee M'$, each applicant is matched to their most-preferred company among M, M'
  - in $M \wedge M'$, each applicant is matched to their least-preferred company among M, M'

A *lattice* is a partially ordered set $(L, \leq)$ s.t. any pair of elements $a, b \in L$ have a least upper bound $a \vee b$ in L and a greatest lower bound $a \wedge b$ in L.

Lattice of Stable Matchings

- **partial order** $\leq$ over stable matchings $\mathscr{M}$ defined by $M \leq M' \Leftrightarrow$ all applicants weakly prefer M' over M
- $(\mathscr{M}, \leq)$ is a (distributive) **lattice**

  see Gale, Sotomayor '85 for a simple proof

  - in $M \vee M'$, each applicant is matched to their most-preferred company among M, M'
  - in $M \wedge M'$, each applicant is matched to their least-preferred company among M, M'

$x \succ_a y \succ_a z \succ_a w$ $\quad$ $c \succ_x d \succ_x b \succ_x a$

$y \succ_b x \succ_b w \succ_b z$ $\quad$ $d \succ_y c \succ_y a \succ_y b$

$z \succ_c w \succ_c x \succ_c y$ $\quad$ $a \succ_z b \succ_z d \succ_z c$

$w \succ_d z \succ_d y \succ_d x$ $\quad$ $b \succ_w a \succ_w c \succ_w d$

Lattice of Stable Matchings

- **partial order** $\leq$ over stable matchings $\mathscr{M}$ defined by $M \leq M' \Leftrightarrow$ all applicants weakly prefer M' over M
- $(\mathscr{M}, \leq)$ is a (distributive) **lattice**

  see Gale, Sotomayor '85 for a simple proof

  - in $M \vee M'$, each applicant is matched to their most-preferred company among M, M'
  - in $M \wedge M'$, each applicant is matched to their least-preferred company among M, M'

$x \succ_a y \succ_a z \succ_a w$ $\quad$ $c \succ_x d \succ_x b \succ_x a$

$y \succ_b x \succ_b w \succ_b z$ $\quad$ $d \succ_y c \succ_y a \succ_y b$

$z \succ_c w \succ_c x \succ_c y$ $\quad$ $a \succ_z b \succ_z d \succ_z c$

$w \succ_d z \succ_d y \succ_d x$ $\quad$ $b \succ_w a \succ_w c \succ_w d$

(ax, by, cz, dw)

Lattice of Stable Matchings

- **partial order** $\leq$ over stable matchings $\mathscr{M}$ defined by $M \leq M' \Leftrightarrow$ all applicants weakly prefer M' over M
- $(\mathscr{M}, \leq)$ is a (distributive) **lattice**

  see Gale, Sotomayor '85 for a simple proof

  - in $M \vee M'$, each applicant is matched to their most-preferred company among M, M'
  - in $M \wedge M'$, each applicant is matched to their least-preferred company among M, M'

$x \succ_a y \succ_a z \succ_a w$ $\quad$ $c \succ_x d \succ_x b \succ_x a$

$y \succ_b x \succ_b w \succ_b z$ $\quad$ $d \succ_y c \succ_y a \succ_y b$

$z \succ_c w \succ_c x \succ_c y$ $\quad$ $a \succ_z b \succ_z d \succ_z c$

$w \succ_d z \succ_d y \succ_d x$ $\quad$ $b \succ_w a \succ_w c \succ_w d$

(ax, by, cz, dw)

(az, bw, cx, dy)

Lattice of Stable Matchings

- **partial order** $\preceq$ over stable matchings $\mathcal{M}$ defined by $M \preceq M' \Leftrightarrow$ all applicants weakly prefer M' over M
- $(\mathcal{M}, \preceq)$ is a (distributive) **lattice**

  see Gale, Sotomayor '85 for a simple proof

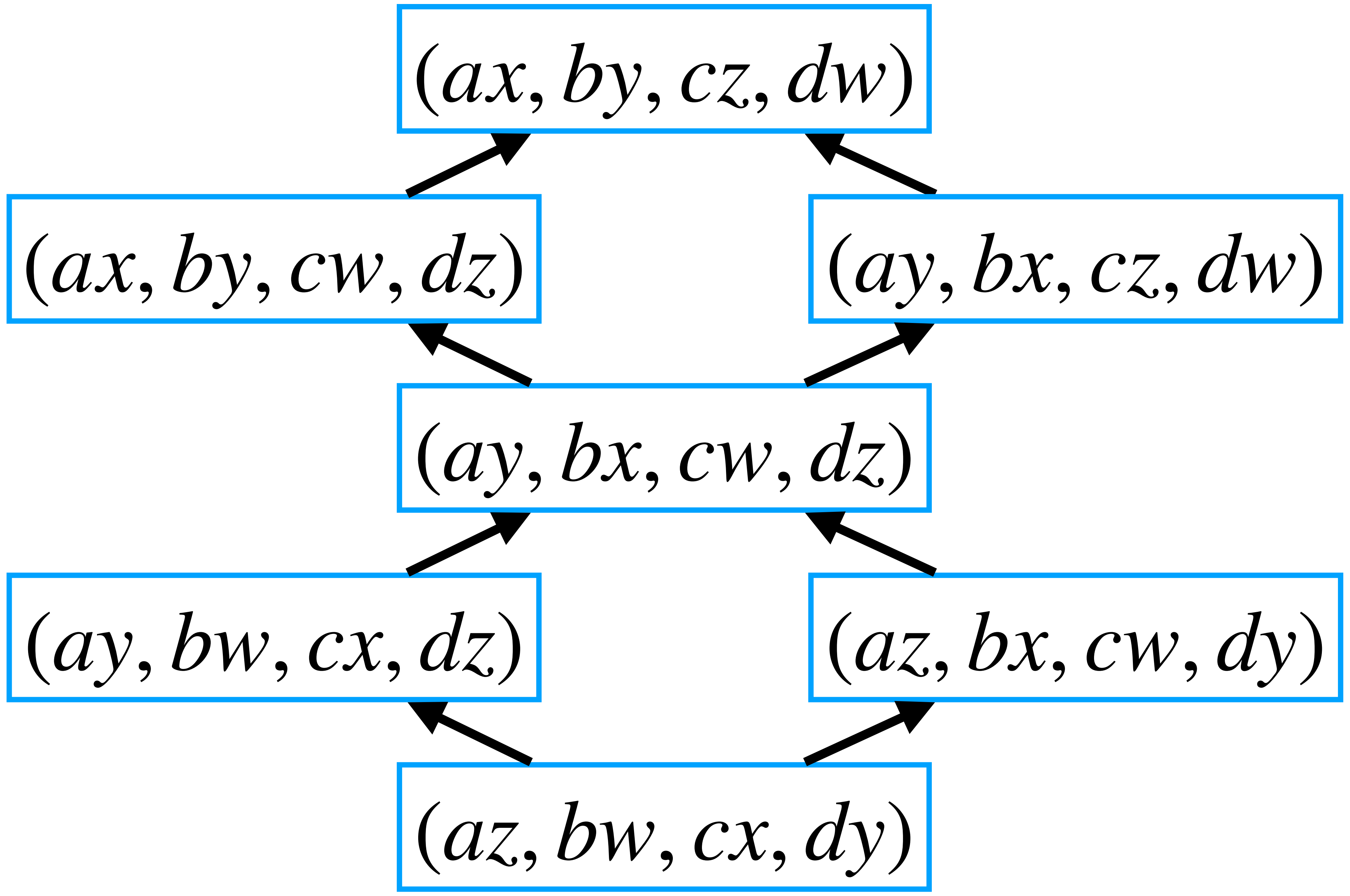
  - in $M \vee M'$, each applicant is matched to their most-preferred company among M, M'
  - in $M \wedge M'$, each applicant is matched to their least-preferred company among M, M'

$x \succ_a y \succ_a z \succ_a w$ $c \succ_x d \succ_x b \succ_x a$

$y \succ_b x \succ_b w \succ_b z$ $d \succ_y c \succ_y a \succ_y b$

$z \succ_c w \succ_c x \succ_c y$ $a \succ_z b \succ_z d \succ_z c$

$w \succ_d z \succ_d y \succ_d x$ $b \succ_w a \succ_w c \succ_w d$



Lattice of Stable Matchings

- **partial order** $\leq$ over stable matchings $\mathcal{M}$ defined by $M \leq M' \Leftrightarrow$ all applicants weakly prefer M' over M
- $(\mathcal{M}, \leq)$ is a (distributive) **lattice**

  see Gale, Sotomayor ’85 for a simple proof
  - in $M \vee M'$, each applicant is matched to their most-preferred company among M, M'
  - in $M \wedge M'$, each applicant is matched to their least-preferred company among M, M'
- $M \vee M'$, $M \wedge M'$ matching?

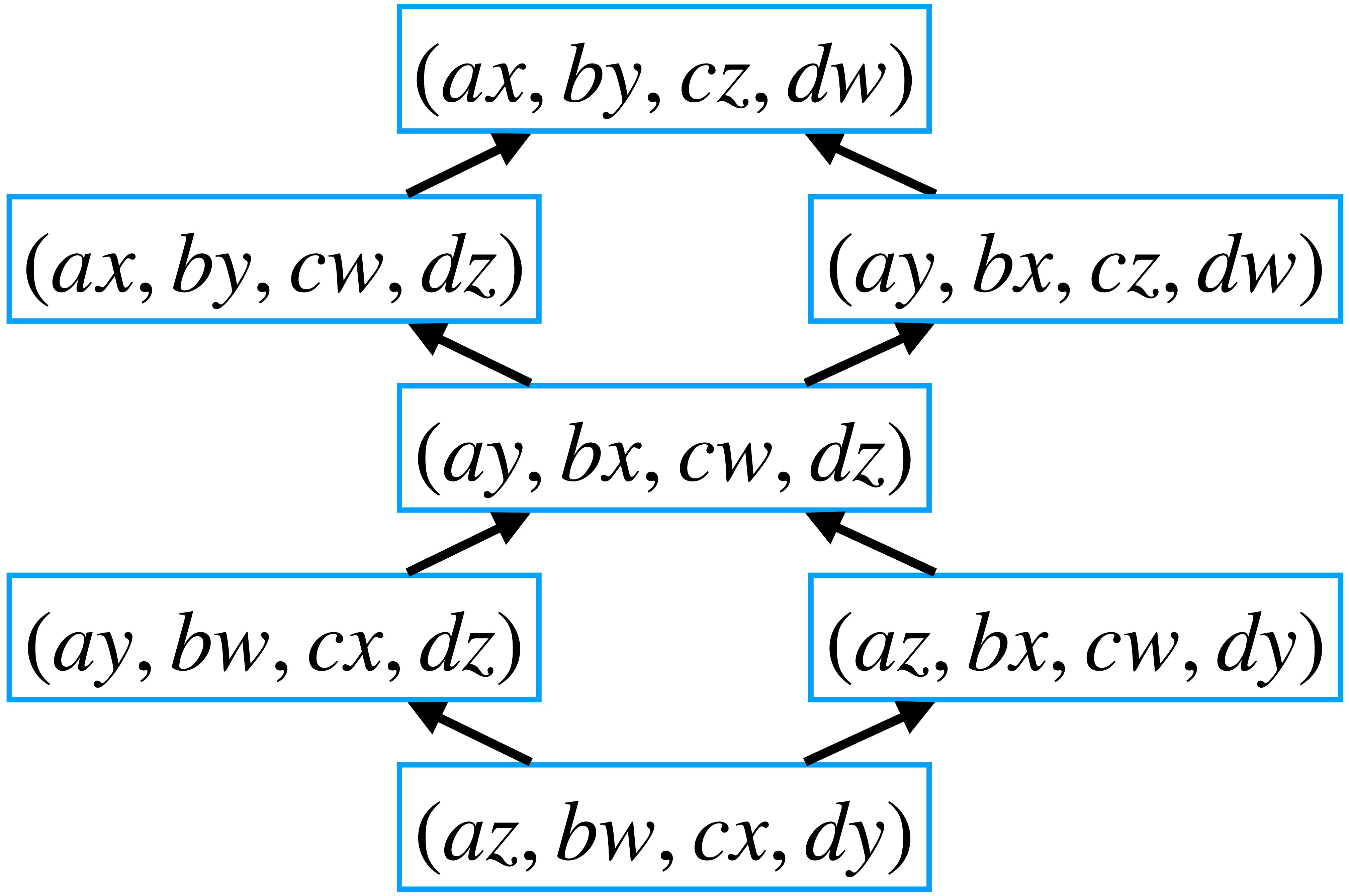
  if two applicants have the same top choice in M and M', one of them forms a blocking pair

$x \succ_a y \succ_a z \succ_a w$ $\quad$ $c \succ_x d \succ_x b \succ_x a$

$y \succ_b x \succ_b w \succ_b z$ $\quad$ $d \succ_y c \succ_y a \succ_y b$

$z \succ_c w \succ_c x \succ_c y$ $\quad$ $a \succ_z b \succ_z d \succ_z c$

$w \succ_d z \succ_d y \succ_d x$ $\quad$ $b \succ_w a \succ_w c \succ_w d$



Lattice of Stable Matchings

- **partial order** $\leq$ over stable matchings $\mathscr{M}$ defined by $M \leq M' \Leftrightarrow$ all applicants weakly prefer M' over M
- $(\mathscr{M}, \leq)$ is a (distributive) **lattice**

  see Gale, Sotomayor '85 for a simple proof

  - in $M \vee M'$, each applicant is matched to their most-preferred company among M, M'
  - in $M \wedge M'$, each applicant is matched to their least-preferred company among M, M'
- $M \vee M'$, $M \wedge M'$ matching?

  if two applicants have the same top choice in M and M', one of them forms a blocking pair
- $M \vee M'$, $M \wedge M'$ stable?

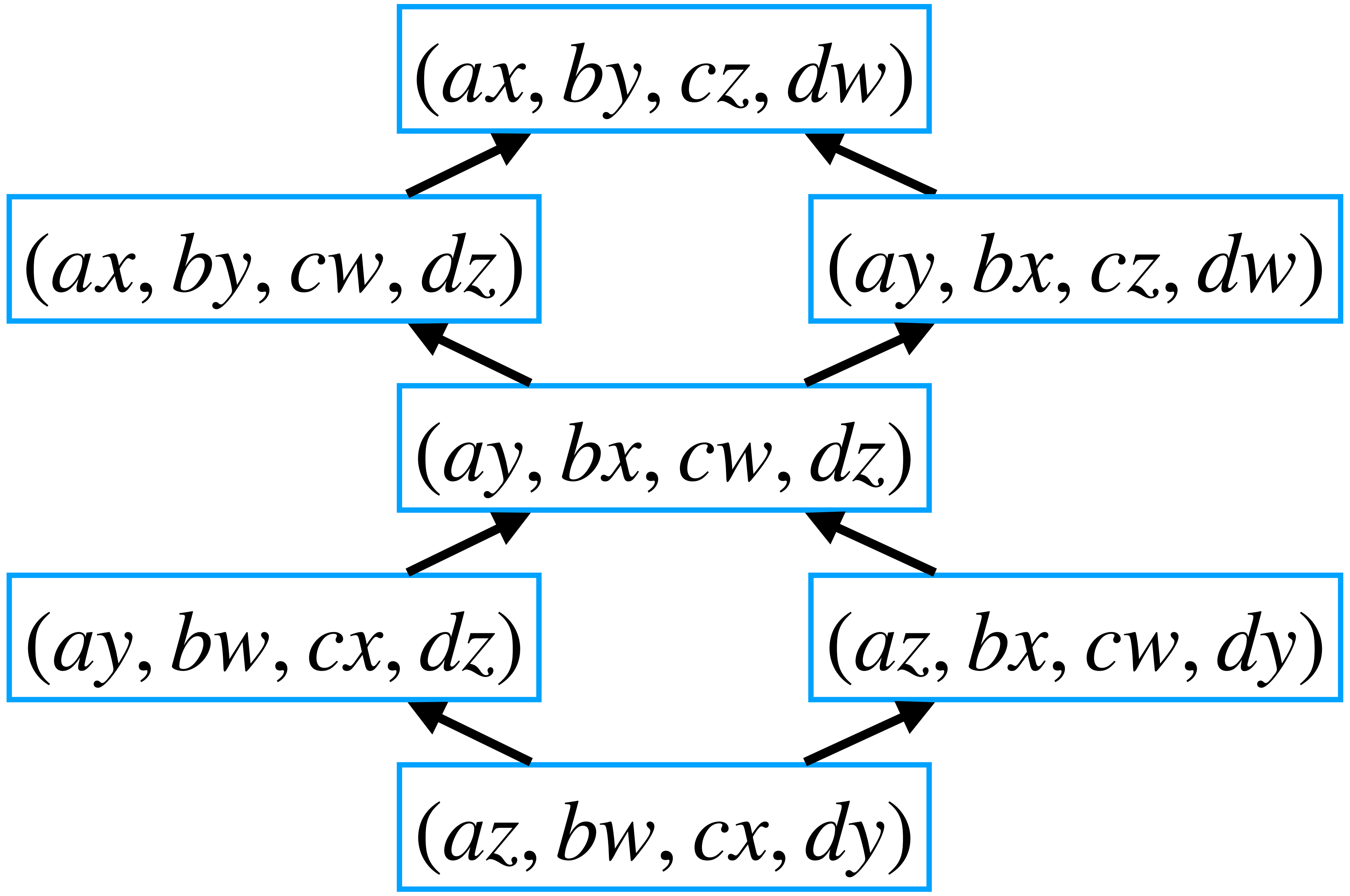
  blocking pairs would be blocking pairs in M or M'

$x \succ_a y \succ_a z \succ_a w$ $\quad$ $c \succ_x d \succ_x b \succ_x a$

$y \succ_b x \succ_b w \succ_b z$ $\quad$ $d \succ_y c \succ_y a \succ_y b$

$z \succ_c w \succ_c x \succ_c y$ $\quad$ $a \succ_z b \succ_z d \succ_z c$

$w \succ_d z \succ_d y \succ_d x$ $\quad$ $b \succ_w a \succ_w c \succ_w d$



Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality

Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality except when proposed side has homogeneous preferences

Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality except when proposed side has homogeneous preferences

- Are SP and Pareto-optimality for one side compatible?

Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality except when proposed side has homogeneous preferences
- Are SP and Pareto-optimality for one side compatible? **yes,** e.g. serial dictatorship

Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality except when proposed side has homogeneous preferences
- Are SP and Pareto-optimality for one side compatible? **yes,** e.g. serial dictatorship
- applications: assignment of doctors to hospitals, students to schools/colleges, etc

Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality except when proposed side has homogeneous preferences
- Are SP and Pareto-optimality for one side compatible? **yes,** e.g. serial dictatorship
- applications: assignment of doctors to hospitals, students to schools/colleges, etc
  - results from this lecture remain valid in such many-to-one matching markets

Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality except when proposed side has homogeneous preferences
- Are SP and Pareto-optimality for one side compatible? **yes,** e.g. serial dictatorship
- applications: assignment of doctors to hospitals, students to schools/colleges, etc
    - results from this lecture remain valid in such many-to-one matching markets
- SP for proposed side is often overlooked, partly because these preferences are often exogenous

Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality except when proposed side has homogeneous preferences
- Are SP and Pareto-optimality for one side compatible? **yes,** e.g. serial dictatorship
- applications: assignment of doctors to hospitals, students to schools/colleges, etc
    - results from this lecture remain valid in such many-to-one matching markets
- SP for proposed side is often overlooked, partly because these preferences are often exogenous
- real-life applications pose several challenges

Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality except when proposed side has homogeneous preferences
- Are SP and Pareto-optimality for one side compatible? **yes,** e.g. serial dictatorship
- applications: assignment of doctors to hospitals, students to schools/colleges, etc
  - results from this lecture remain valid in such many-to-one matching markets
- SP for proposed side is often overlooked, partly because these preferences are often exogenous
- real-life applications pose several challenges
  - diversity/fairness considerations [Hafalir et al. ’13]

Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality except when proposed side has homogeneous preferences
- Are SP and Pareto-optimality for one side compatible? **yes,** e.g. serial dictatorship
- applications: assignment of doctors to hospitals, students to schools/colleges, etc
  - results from this lecture remain valid in such many-to-one matching markets
- SP for proposed side is often overlooked, partly because these preferences are often exogenous
- real-life applications pose several challenges
  - diversity/fairness considerations [Hafalir et al. ’13]
  - joint applications (couples/families) [Nguyen, Vohra ’18]

Takeaways and Applications

- Deferred Acceptance: stability, strategy-proofness for proposing side, computational efficiency, but not Pareto-optimality except when proposed side has homogeneous preferences
- Are SP and Pareto-optimality for one side compatible? **yes,** e.g. serial dictatorship
- applications: assignment of doctors to hospitals, students to schools/colleges, etc
  - results from this lecture remain valid in such many-to-one matching markets
- SP for proposed side is often overlooked, partly because these preferences are often exogenous
- real-life applications pose several challenges
  - diversity/fairness considerations [Hafalir et al. '13]
  - joint applications (couples/families) [Nguyen, Vohra '18]
  - Chilean school admission system gives interesting examples! [Correa et al. '21]

References

- Correa, J., Epstein, N., Epstein, R., Escobar, J., Rios, I., Aramayo, N., Bahamondes, B., Bonet, C., Castillo, M., Cristi, A., Epstein, B., Subiabre, F. (2022). School Choice in Chile. *Operations Research*, *70*(2), 1066-1087.
- Gale, D., Shapley, L. S. (1962). College admissions and the stability of marriage. *The American mathematical monthly*, *69*(1), 9-15.
- Gale, D., Sotomayor, M. (1985). Some remarks on the stable matching problem. *Discrete Applied Mathematics*, *11*(3), 223-232.
- Hafalir, I. E., Yenmez, M. B., Yildirim, M. A. (2013). Effective affirmative action in school choice. Theoretical Economics, 8(2), 325-363.
- Nguyen, T., Vohra, R. (2018). Near-feasible stable matchings with couples. *American Economic Review*, *108*(11), 3154-3169.
- Roth, A. E. (1982). The economics of matching: Stability and incentives. *Mathematics of operations research*, *7*(4), 617-628.