Proof Complexity of Practical Integer Programming

Noah Fleming

Based on

- On the Power and Limitations of Branch and Cut

  Fleming, Göös, Impagliazzo, Pitassi, Robere, Wigderson

Outline

- Using Proof Complexity for Algorithm Analysis
- Proof Complexity of Integer Programming
  - Cutting Planes
- Branch and Cut and Stabbing Planes
- Stabbing Planes vs. Cutting Planes

Algorithm Analysis from Proofs

Idea: Formalize the techniques used in a class of algorithms A as a proof system P

Algorithm Analysis from Proofs

Idea: Formalize the techniques used in a class of algorithms A as a proof system P

▷ Hides practical details of algorithms

Algorithm Analysis from Proofs

Idea: Formalize the techniques used in a class of algorithms A as a proof system P

- Hides practical details of algorithms
- Lower bounds on P-proofs → lower bounds on runtime of A

Algorithm Analysis from Proofs

Idea: Formalize the techniques used in a class of algorithms A as a proof system P

- e.g. Algorithms for SAT
  - CDCL and Resolution

Algorithm Analysis from Proofs

Idea: Formalize the techniques used in a class of algorithms A as a proof system P

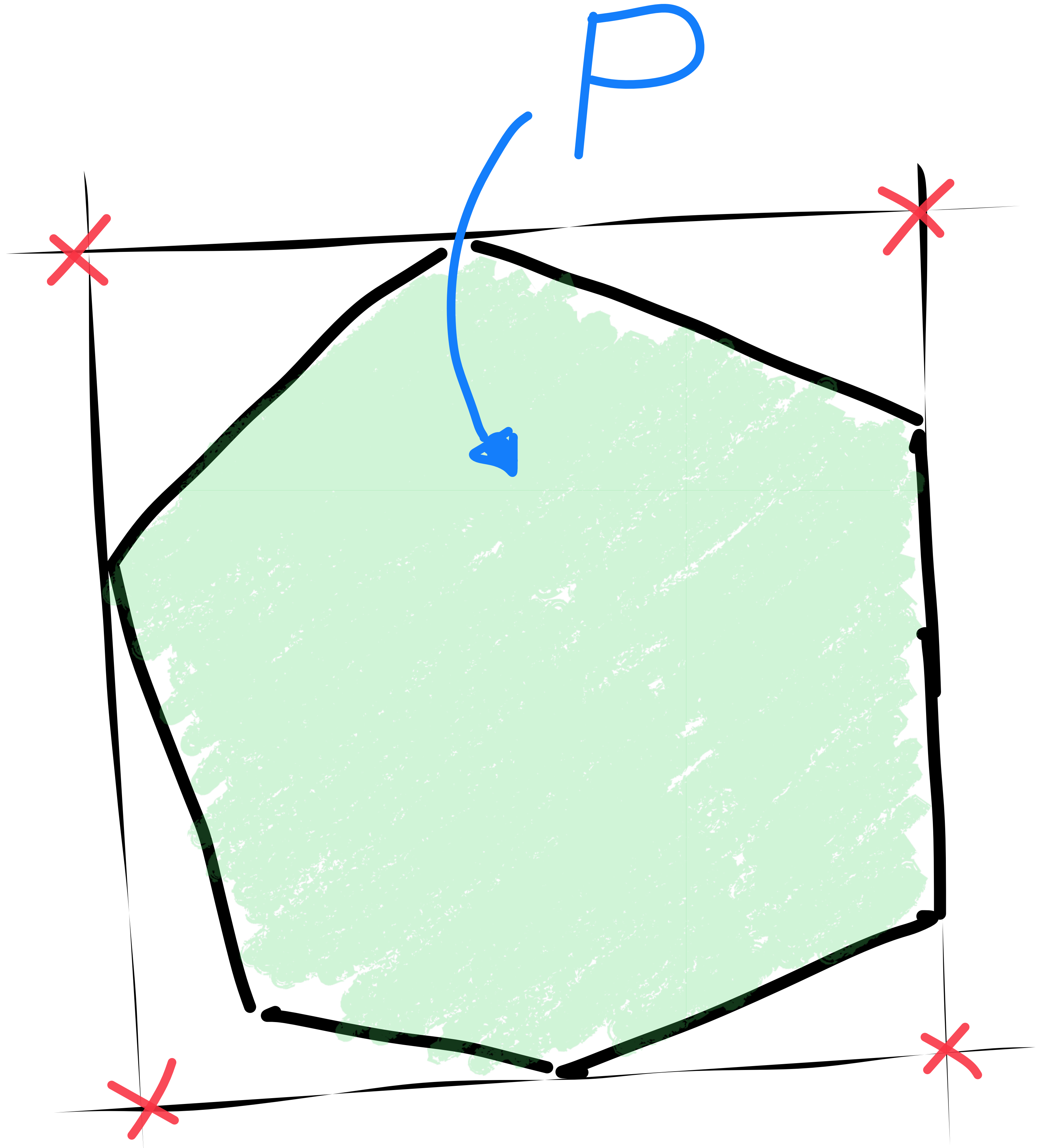
- e.g. Algorithms for SAT
  - CDCL and Resolution
- e.g. Algorithms for Integer Programming
  - Chvátal-Gomory Cutting Planes and Cutting Planes

Integer Programming

Integer-programming: Given $Ax \geq b$ find $x \in \mathbb{Z}^n, Ax \geq b$

Integer Programming

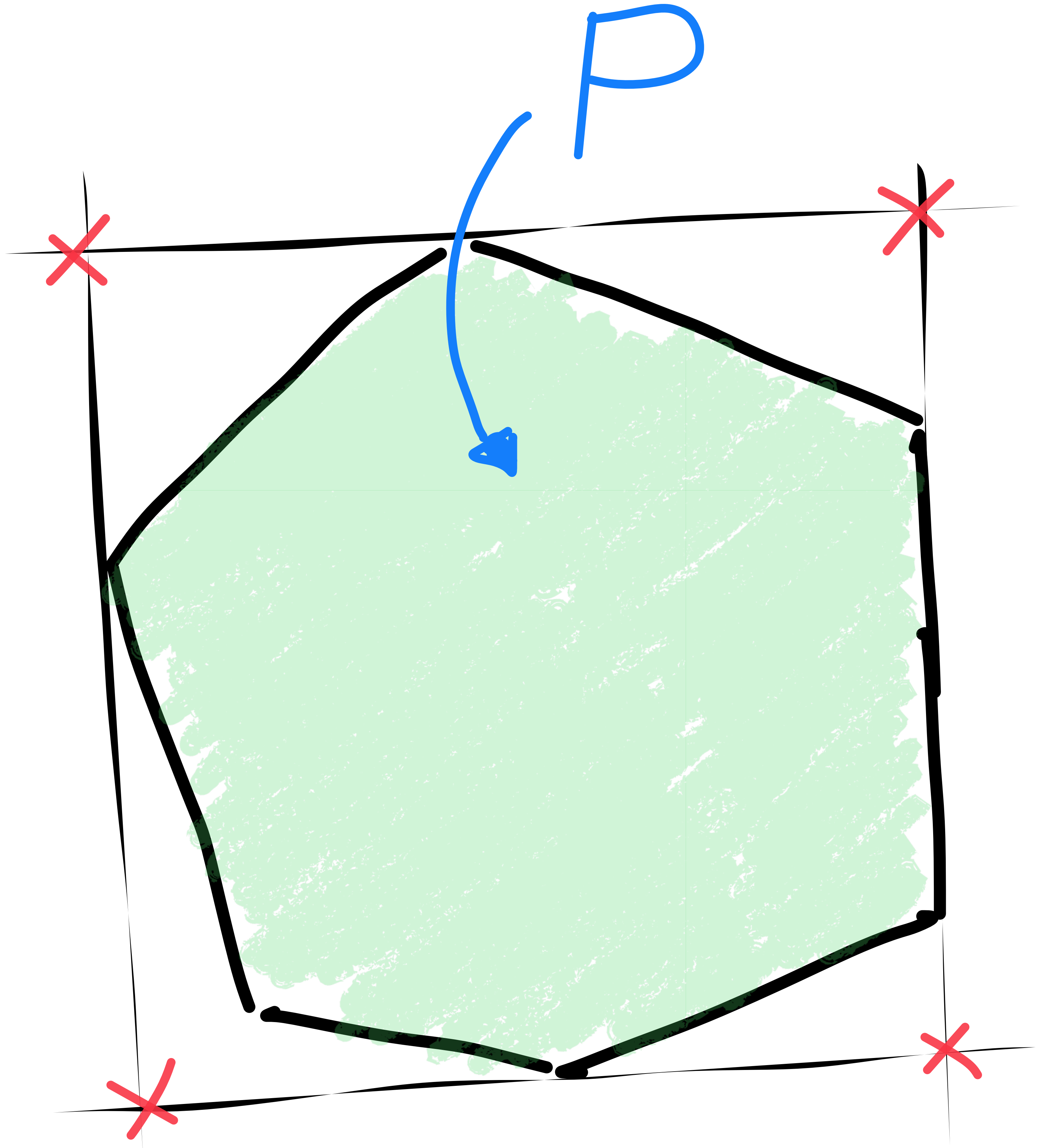
Integer-programming: Given $P = \{x : Ax \geq b\}$ find $x \in P \cap \mathbb{Z}^n$



Integer Programming

Integer-programming: Given $P = \{x : Ax \geq b\}$ find $x \in P \cap \mathbb{Z}^n$

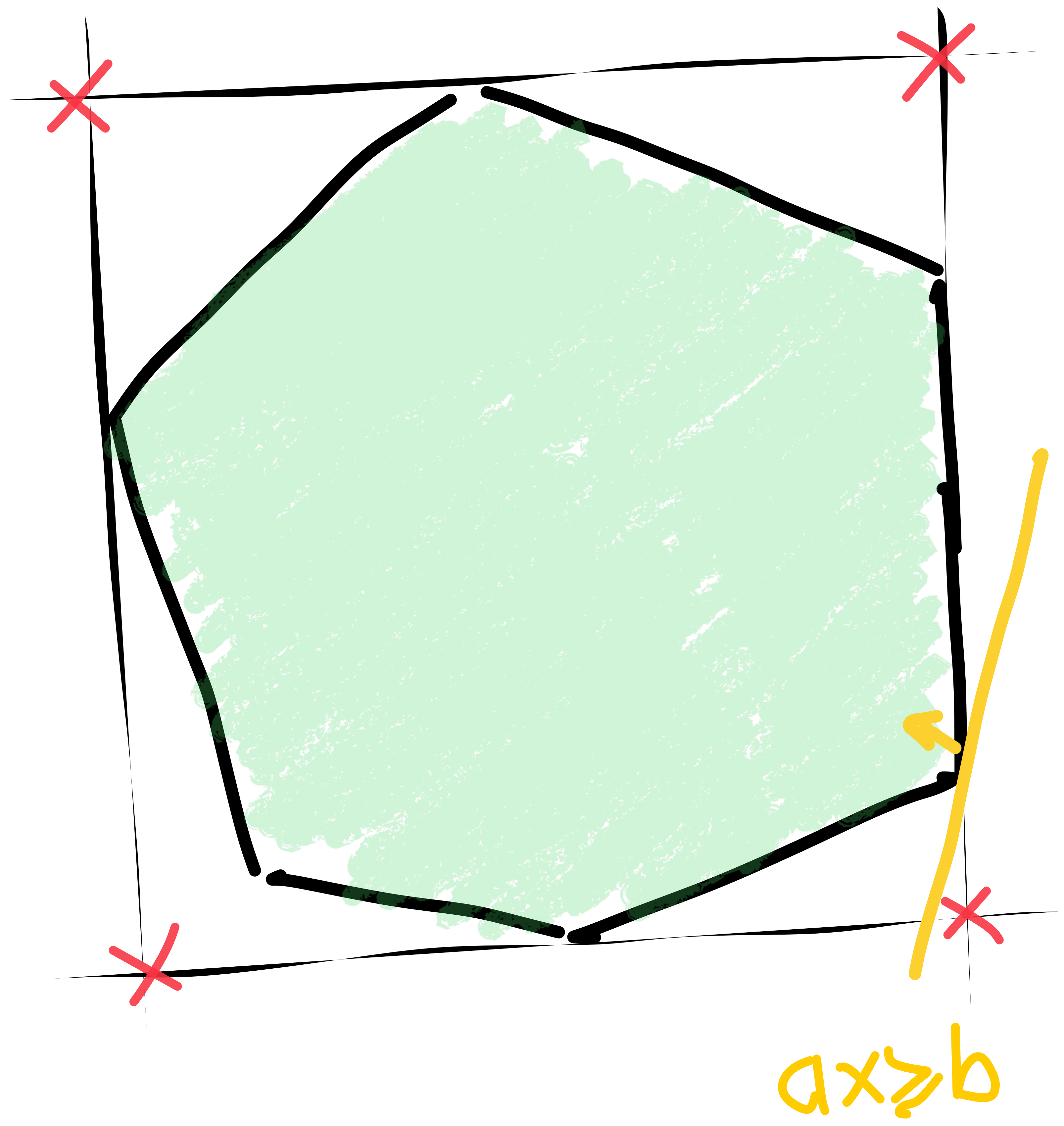
A Classic approach: Chvátal-Gomory Cutting Planes



Chvátal-Gomory Cutting Planes

Integer-programming: Given $P=\{x: Ax\geq b\}$ find $x\in P\cap\mathbb{Z}^n$

A Classic approach: Chvátal-Gomory Cutting Planes



CG-Cut: If $ax\geq b$ is valid for P

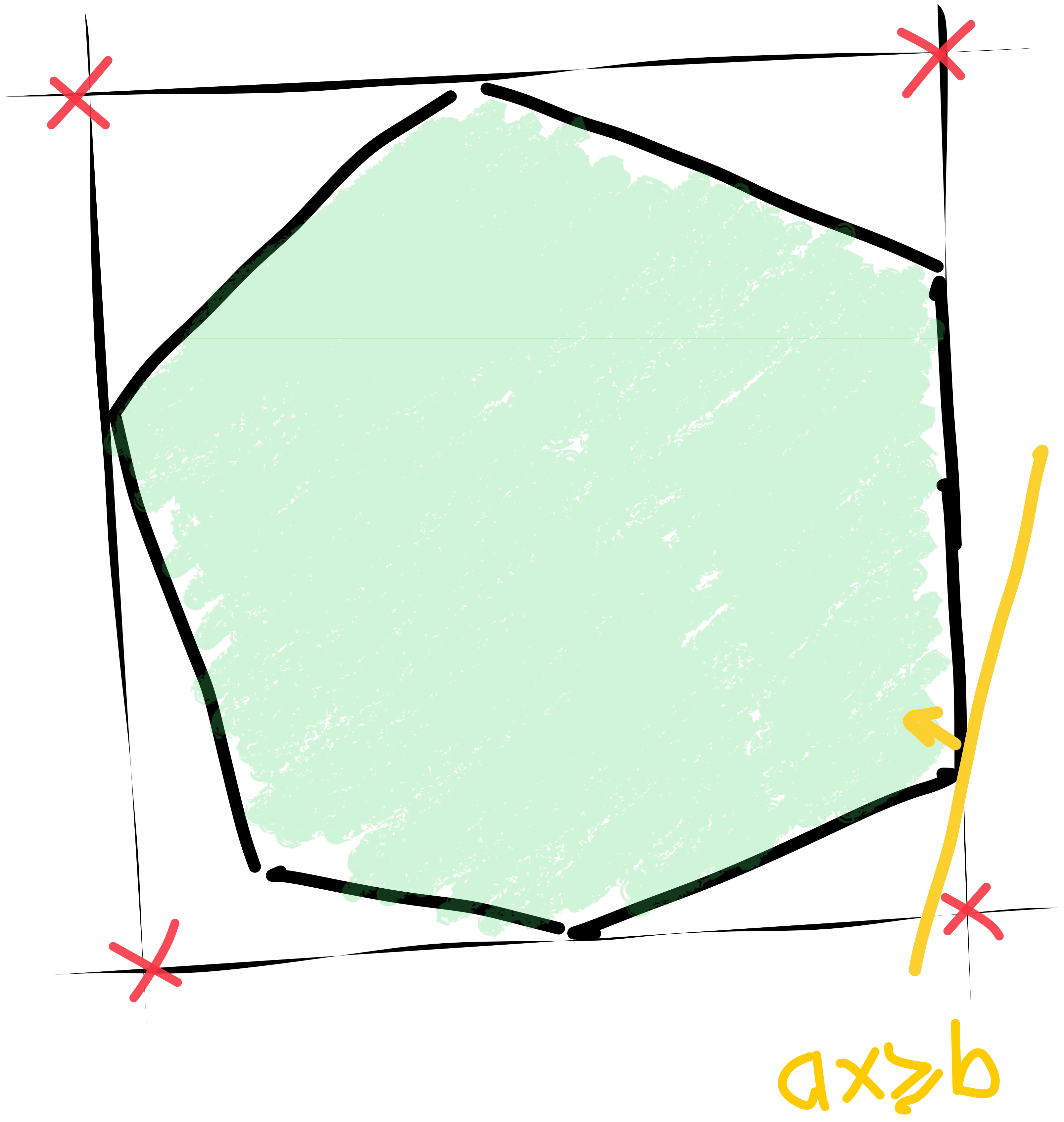
Chvátal-Gomory Cutting Planes

Integer-programming: Given $P = \{x : Ax \geq b\}$ find $x \in P \cap \mathbb{Z}^n$

A Classic approach: Chvátal-Gomory Cutting Planes

$a \in \mathbb{Z}^n, b \in \mathbb{R}$

CG-Cut: If $ax \geq b$ is valid for P



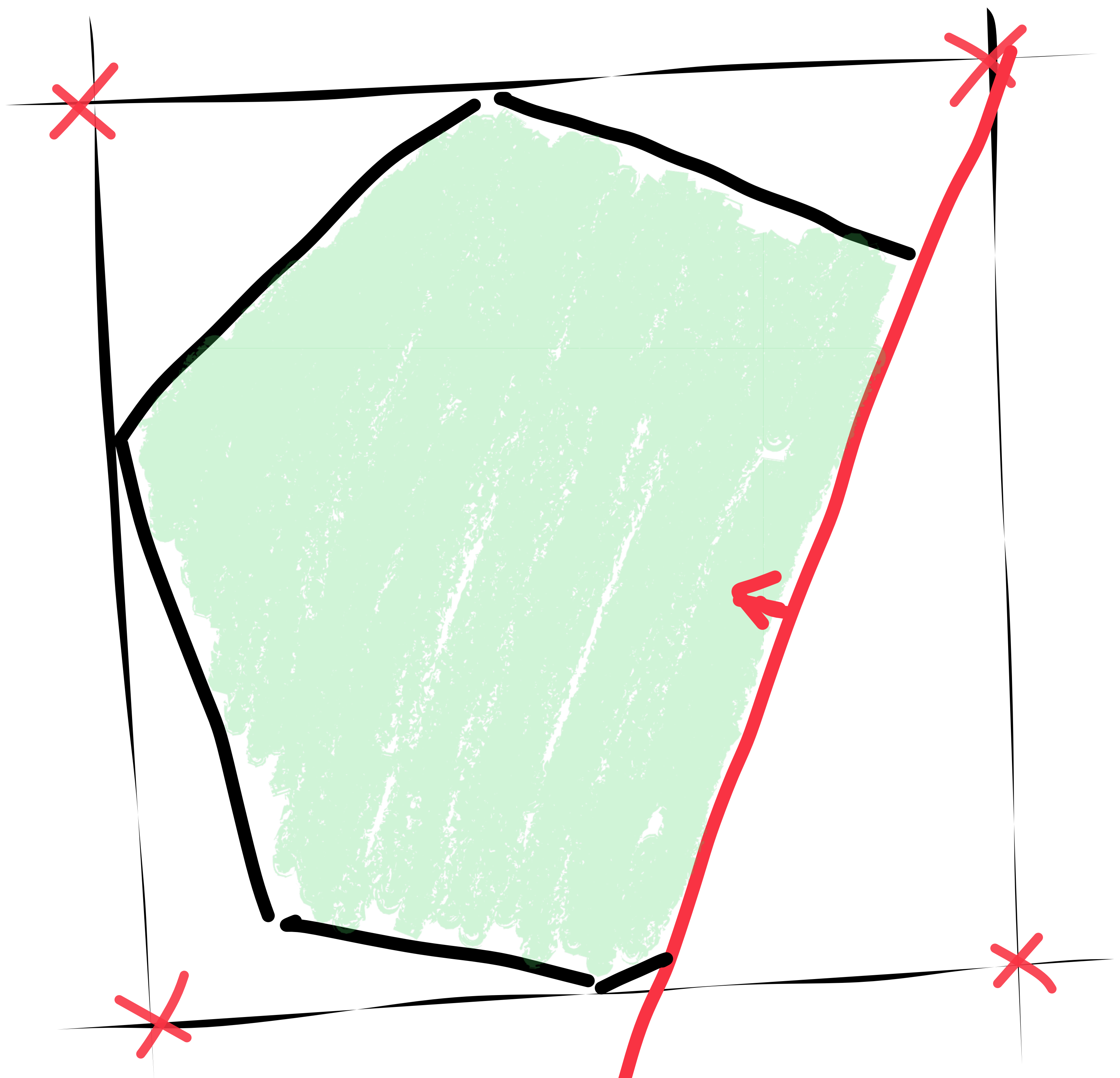
Chvátal-Gomory Cutting Planes

Integer-programming: Given $P = \{x : Ax \geq b\}$ find $x \in P \cap \mathbb{Z}^n$

A Classic approach: Chvátal-Gomory Cutting Planes

$a \in \mathbb{Z}^n, b \in \mathbb{R}$

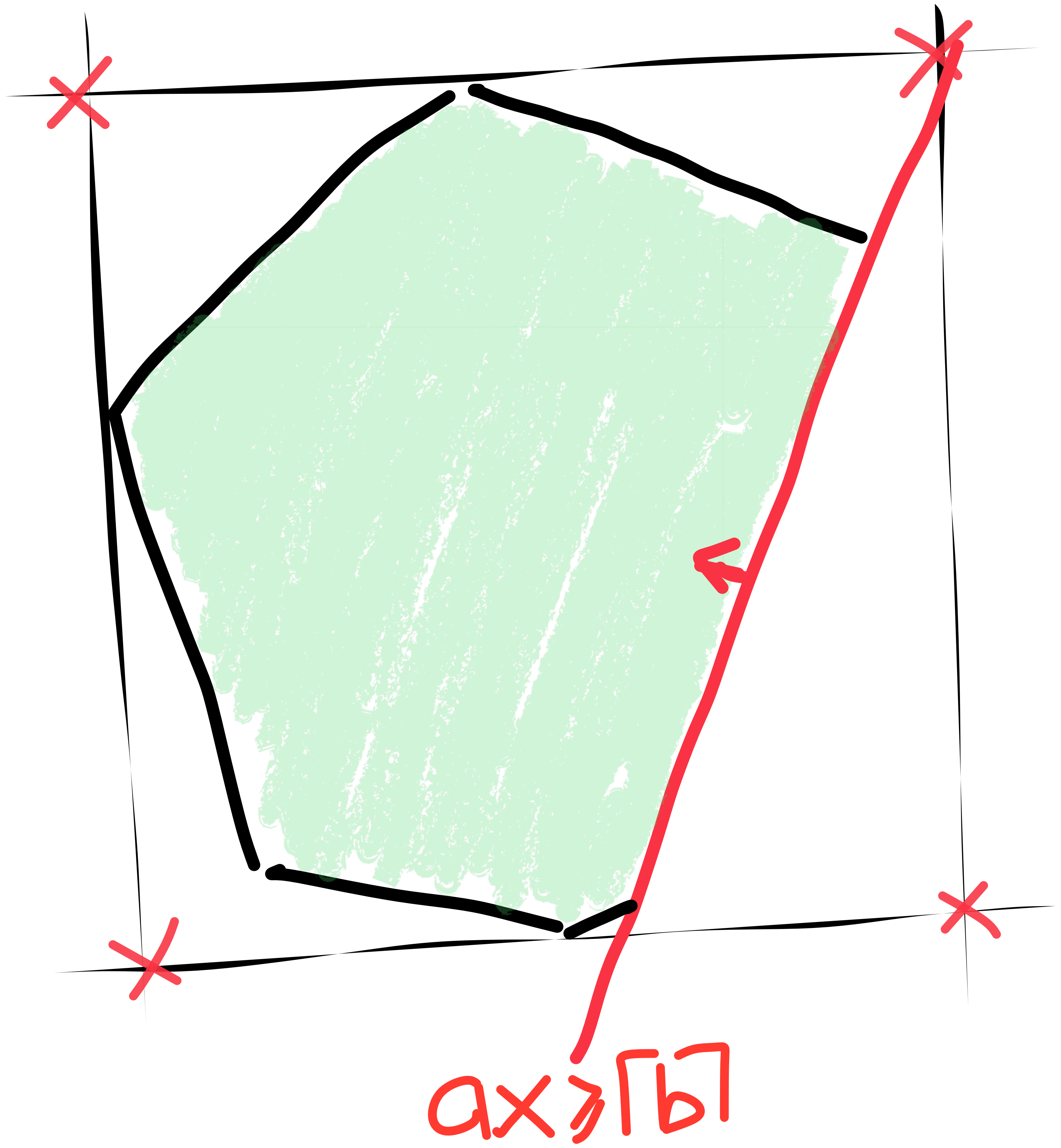
CG-Cut: If $ax \geq b$ is valid for P then $ax \geq \lceil b \rceil$ is a CG-cut



Chvátal-Gomory Cutting Planes

Integer-programming: Given $P=\{x: Ax \geq b\}$ find $x \in P \cap \mathbb{Z}^n$

A Classic approach: Chvátal-Gomory Cutting Planes



$a \in \mathbb{Z}^n, b \in \mathbb{R}$

CG-Cut: If $ax \geq b$ is valid for P

then $ax \geq \lceil b \rceil$ is a CG-cut

→ Preserves integer solutions to P

Chvátal-Gomory Cutting Planes

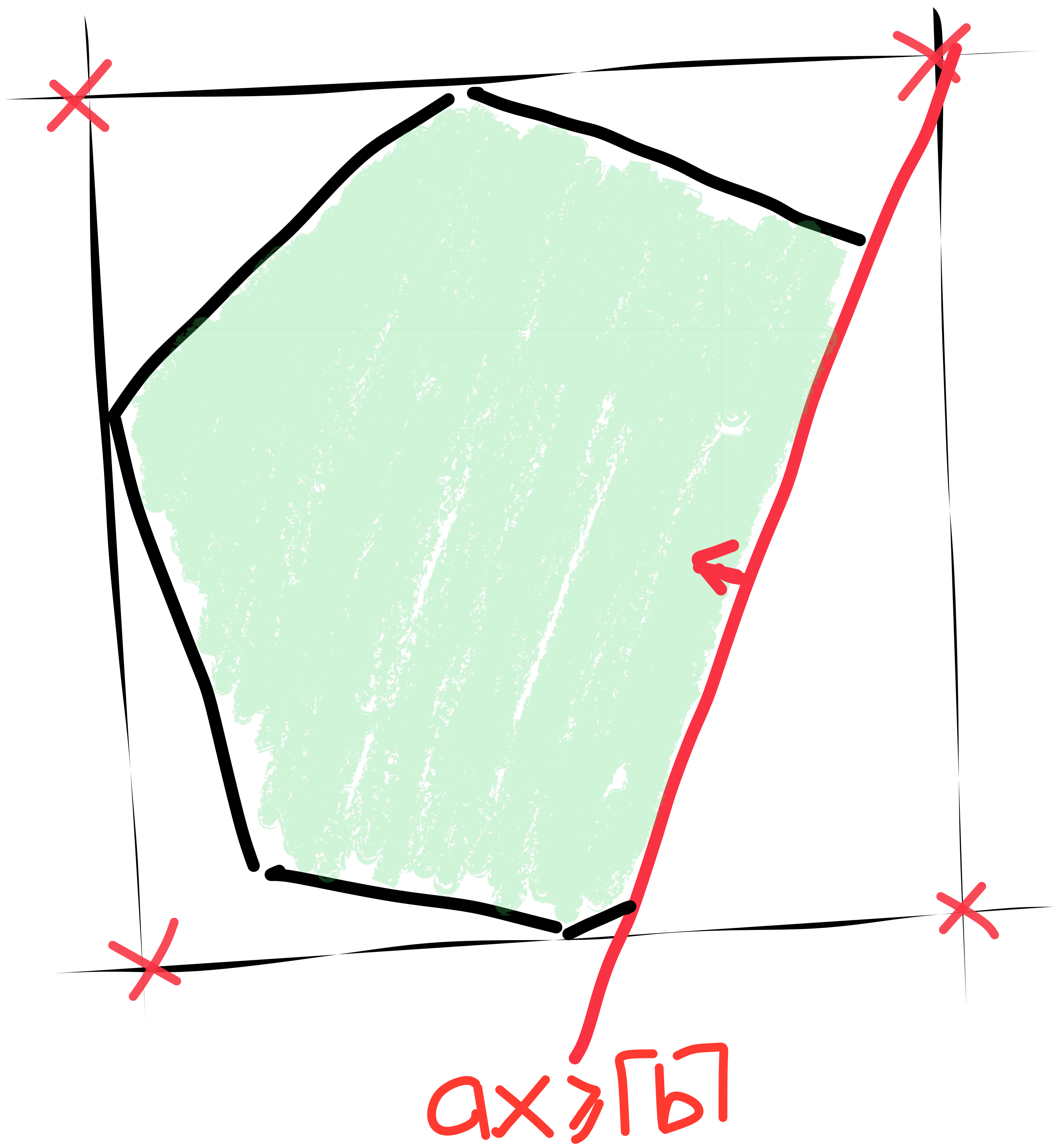
Integer-programming: Given $P=\{x: Ax\geq b\}$ find $x\in P\cap\mathbb{Z}^n$

CG-Cutting Planes

Heuristically add CG-cuts to P

CG-Cut: If $ax\geq b$ is valid for P then $ax\geq\lceil b\rceil$ is a CG-cut

→ Preserves integer solutions to P



Chvátal-Gomory Cutting Planes

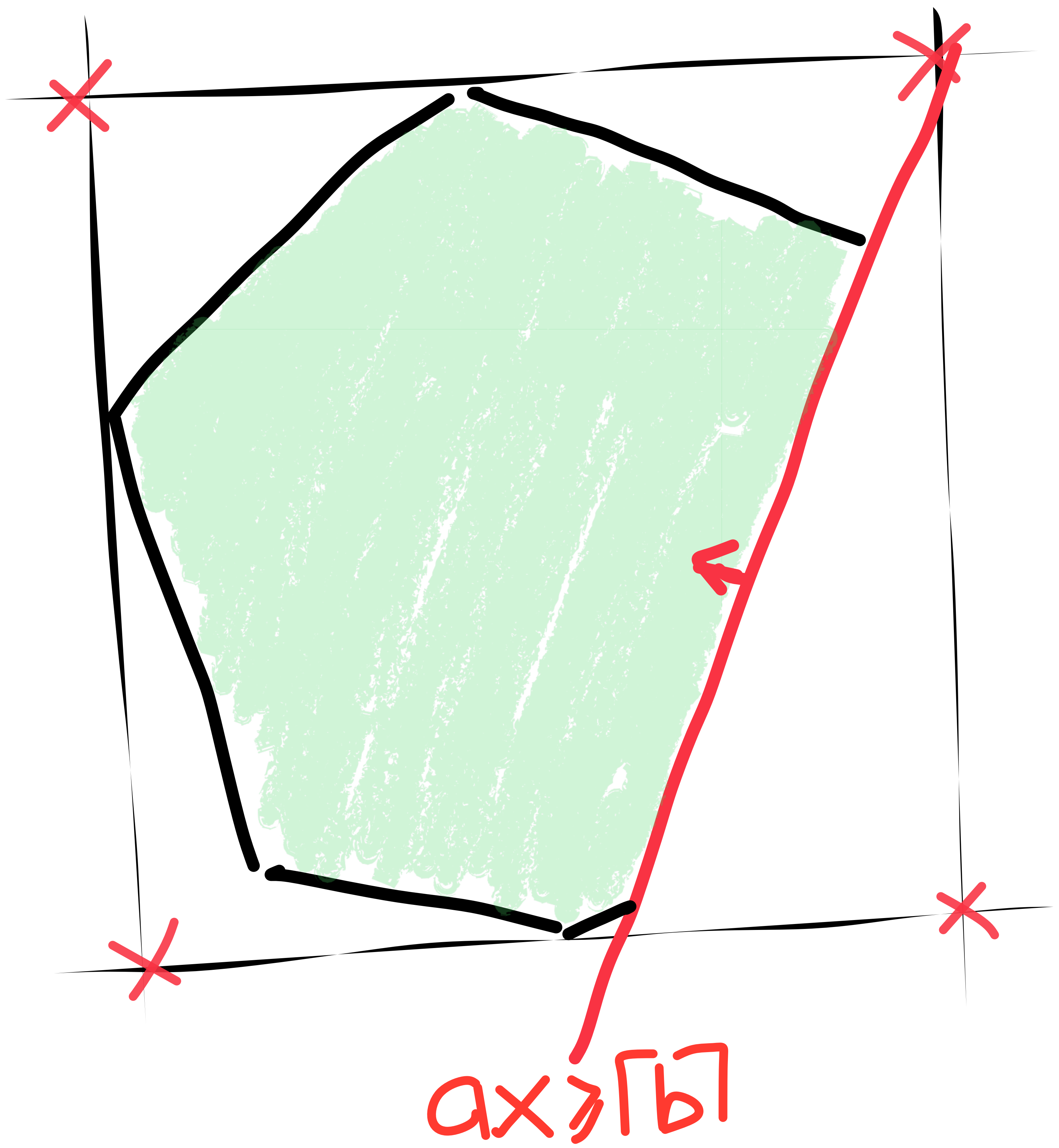
Integer-programming: Given $P = \{x : Ax \geq b\}$ find $x \in P \cap \mathbb{Z}^n$

CG-Cutting Planes

Heuristically add CG-cuts to P

CG-Cut: If $ax \geq b$ is valid for P
then $ax \geq \lceil b \rceil$ is a CG-cut

→ Preserves integer solutions to P



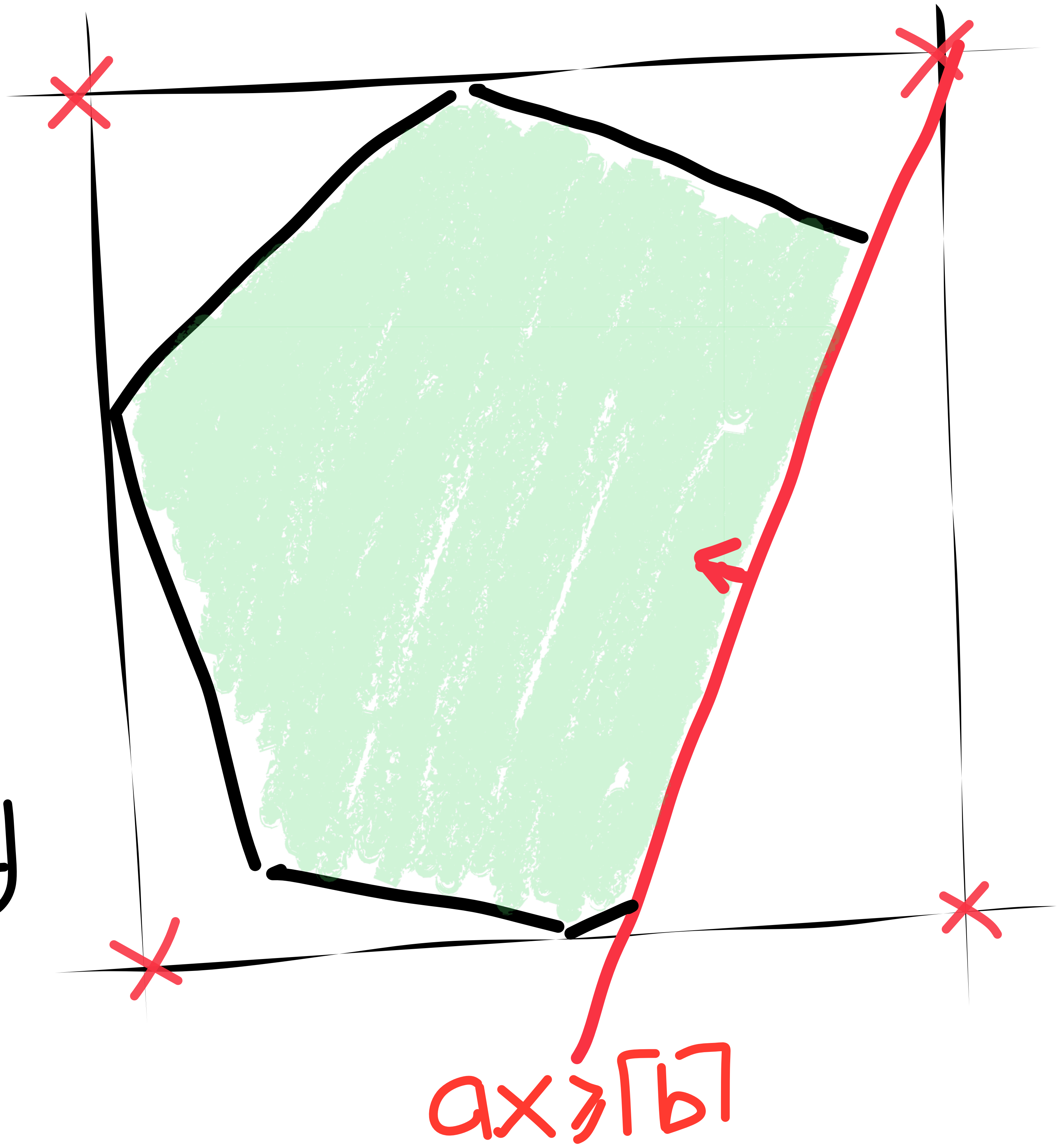
Chvátal-Gomory Cutting Planes

Integer-programming: Given $P = \{x : Ax \geq b\}$ find $x \in P \cap \mathbb{Z}^n$

CG-Cutting Planes

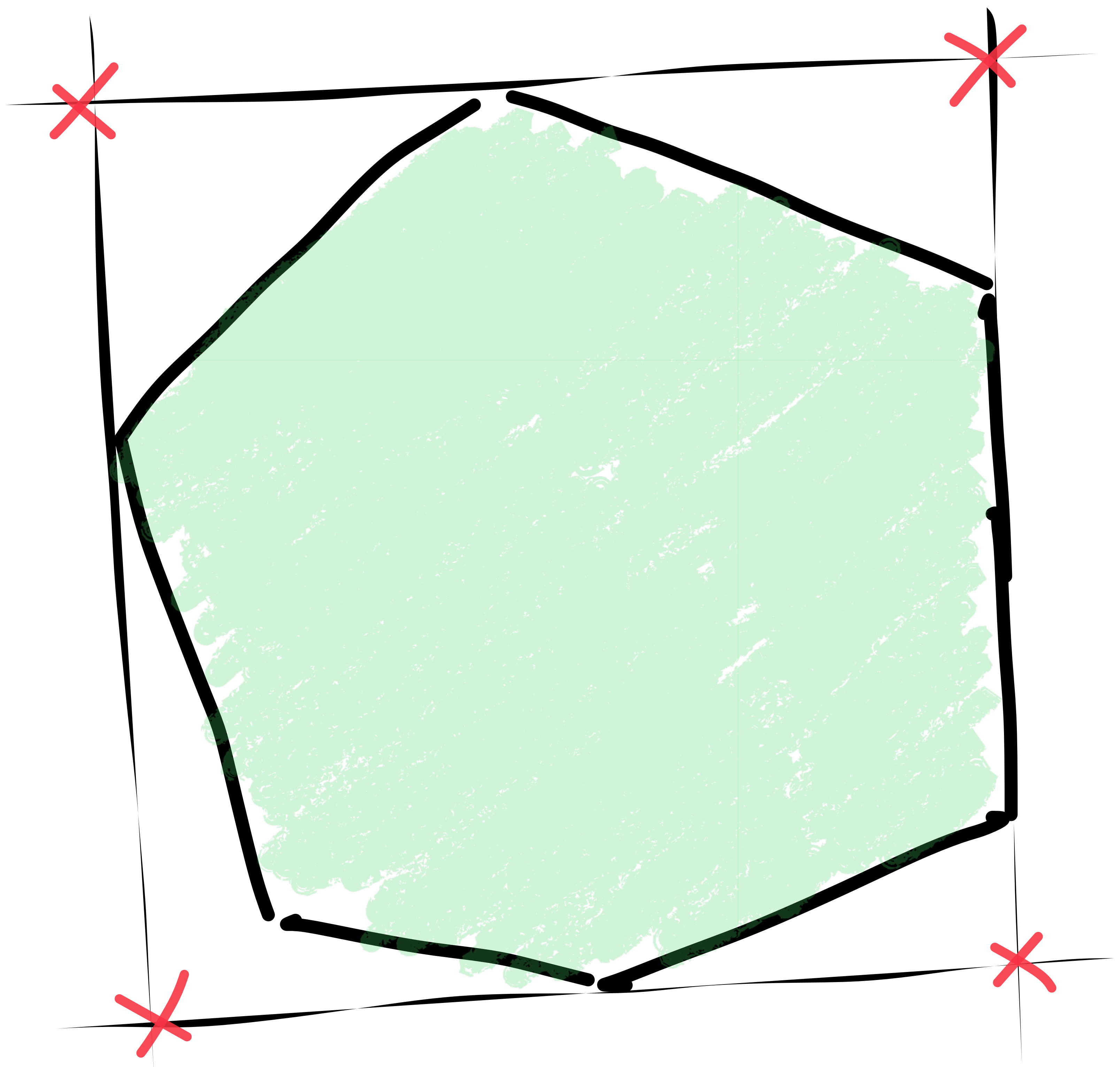
Heuristically add CG-Cuts to P until:

- an integer solution is found
- the empty polytope is deduced



Cutting Planes [CCT87]

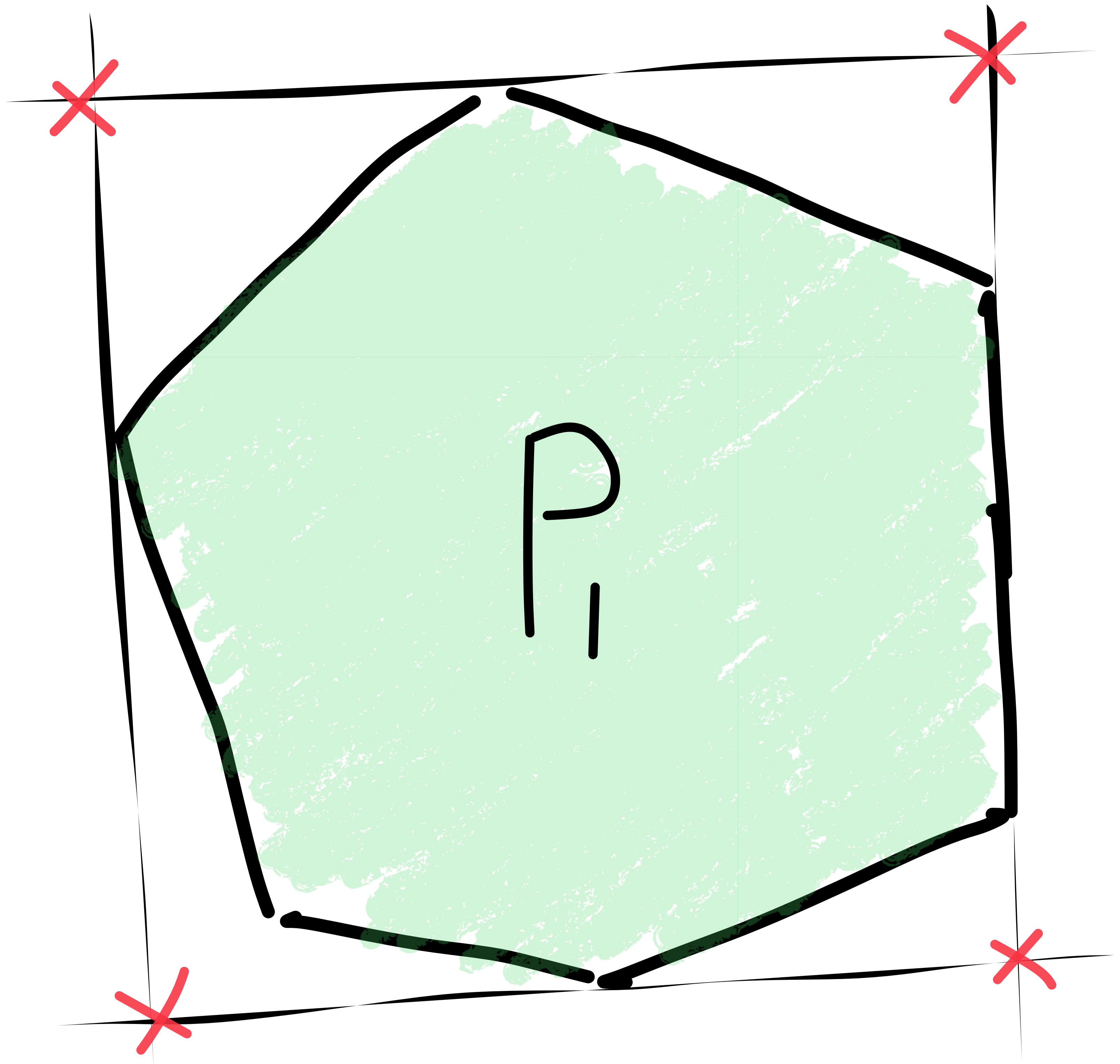
Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$



Cutting Planes

Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$

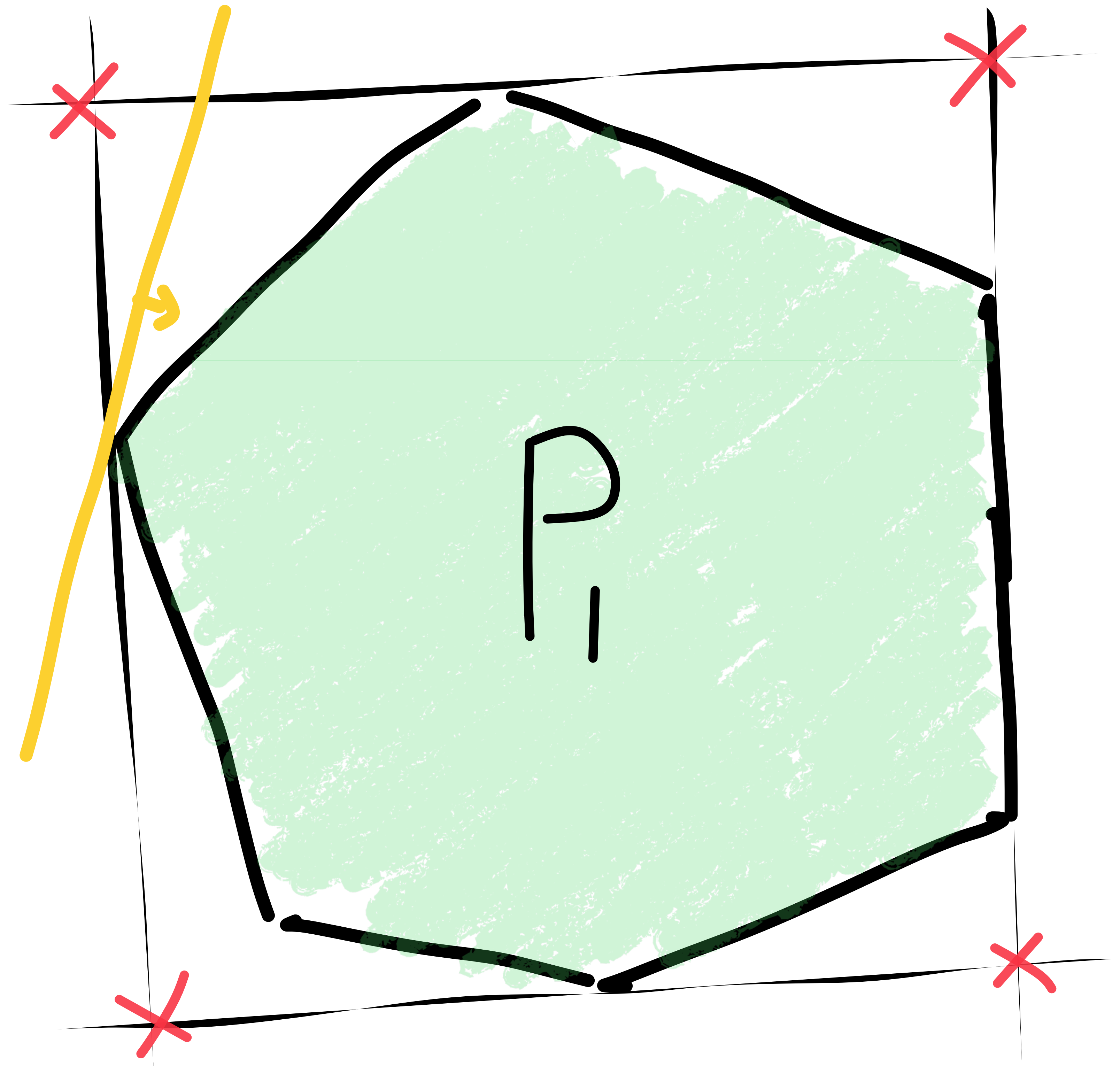
A CP proof that $P \cap \mathbb{Z}^n = \emptyset$ is a sequence of polytopes $P = P_1, \ldots, P_s = \emptyset$ s.t. P_{i+1} is deduced from P_i by a CG-cut



Cutting Planes

Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$

A CP proof that $P \cap \mathbb{Z}^n = \emptyset$ is a sequence of polytopes $P = P_1, \dots, P_s = \emptyset$ s.t. P_{i+1} is deduced from P_i by a CG-cut

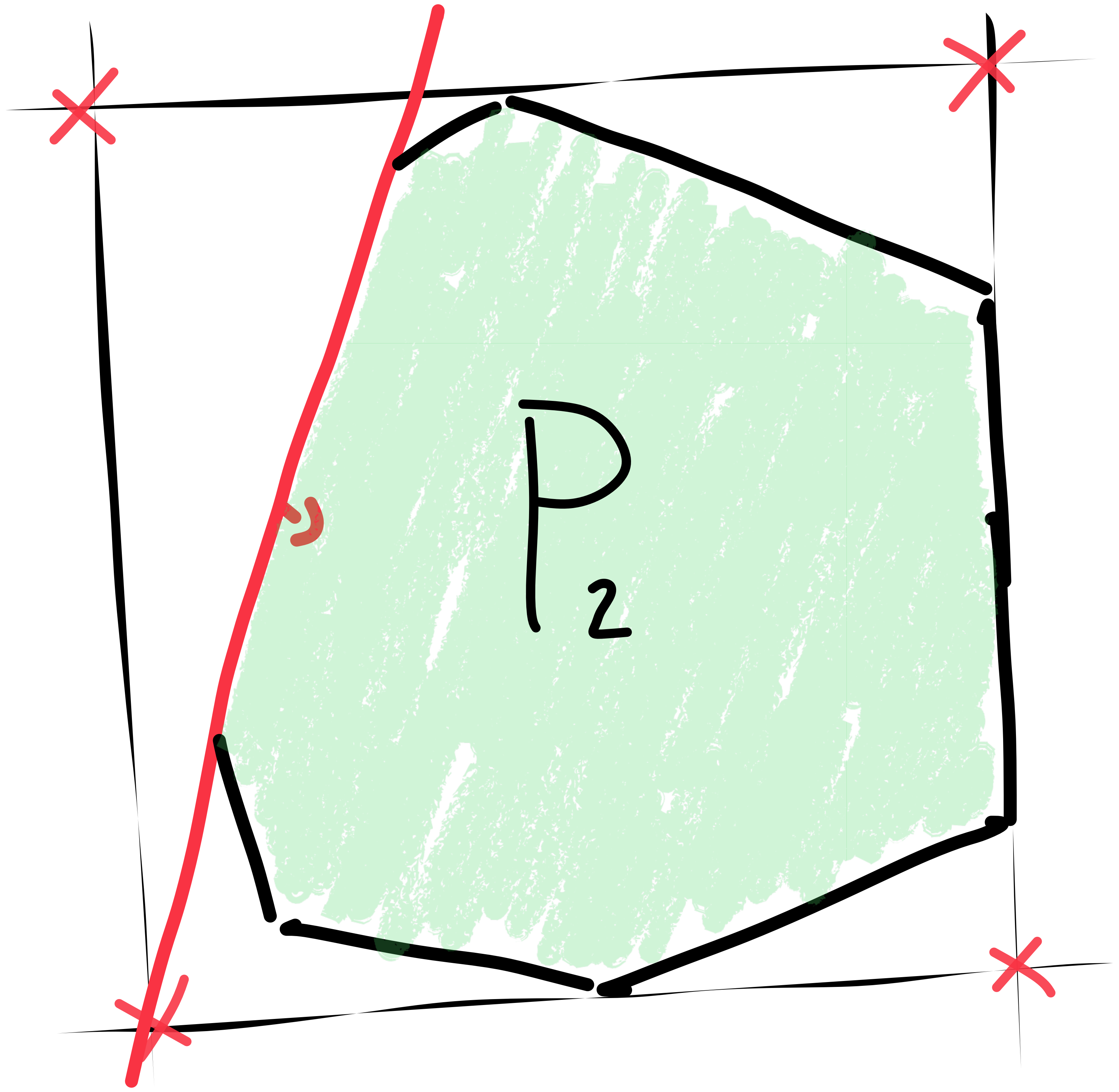


Cutting Planes

Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$

A CP proof that $P \cap \mathbb{Z}^n = \emptyset$ is a sequence of polytopes $P = P_1, \dots, P_s = \emptyset$

s.t. P_{i+1} is deduced from P_i by a CG-cut

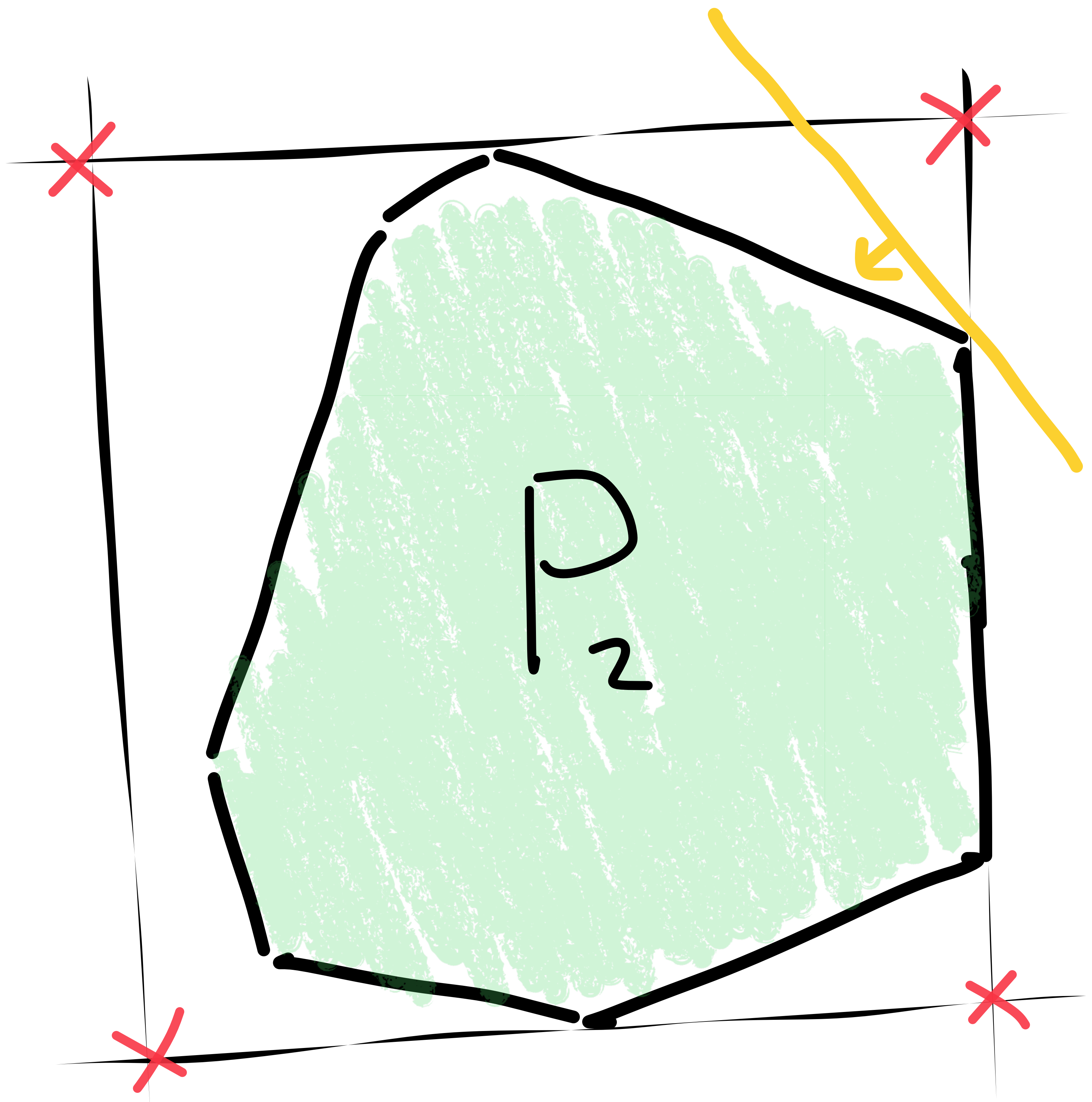


Cutting Planes

Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$

A CP proof that $P \cap \mathbb{Z}^n = \emptyset$ is a sequence of polytopes $P = P_1, \ldots, P_s = \emptyset$

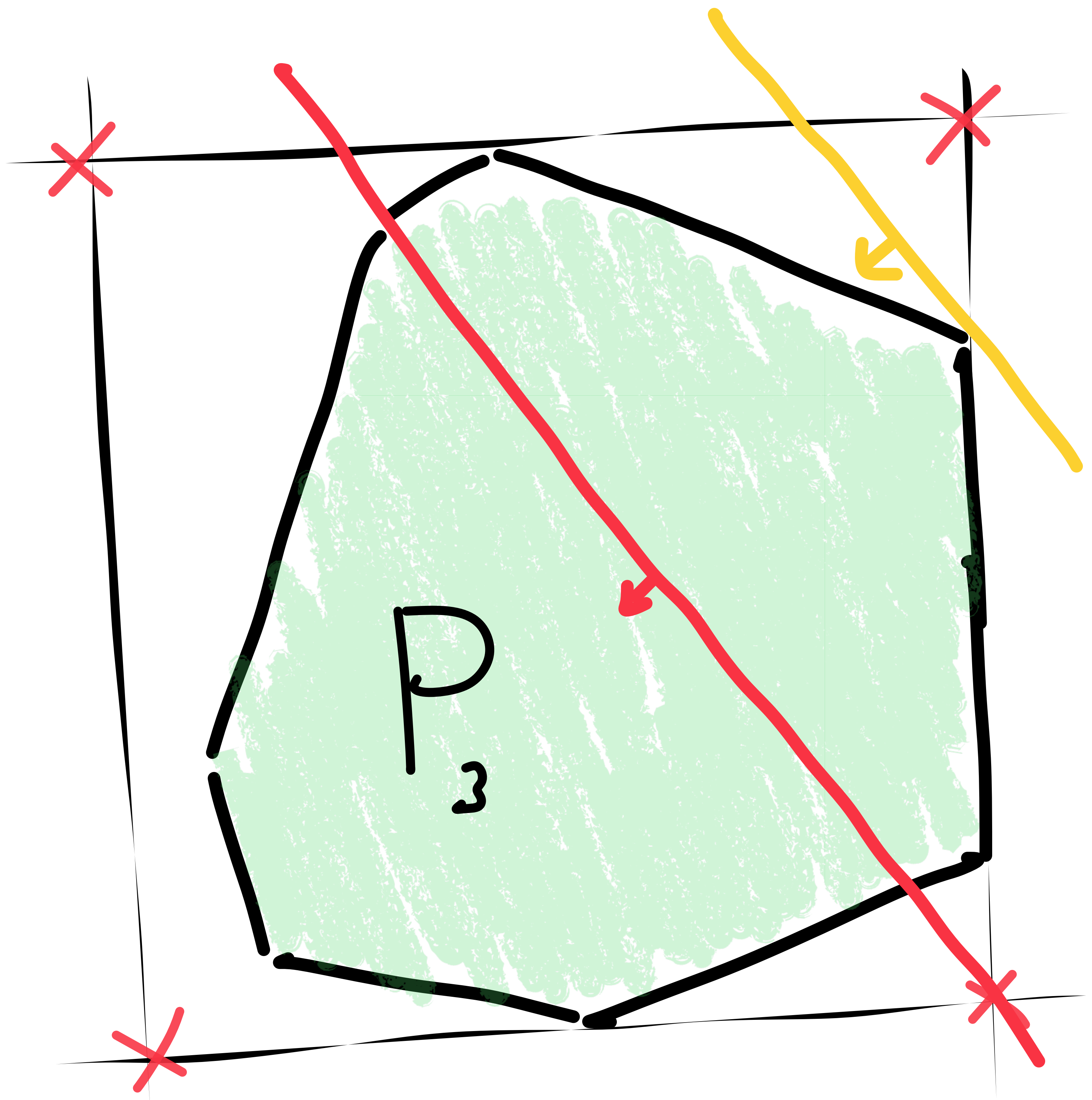
s.t. P_{i+1} is deduced from P_i by a CG-cut



Cutting Planes

Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$

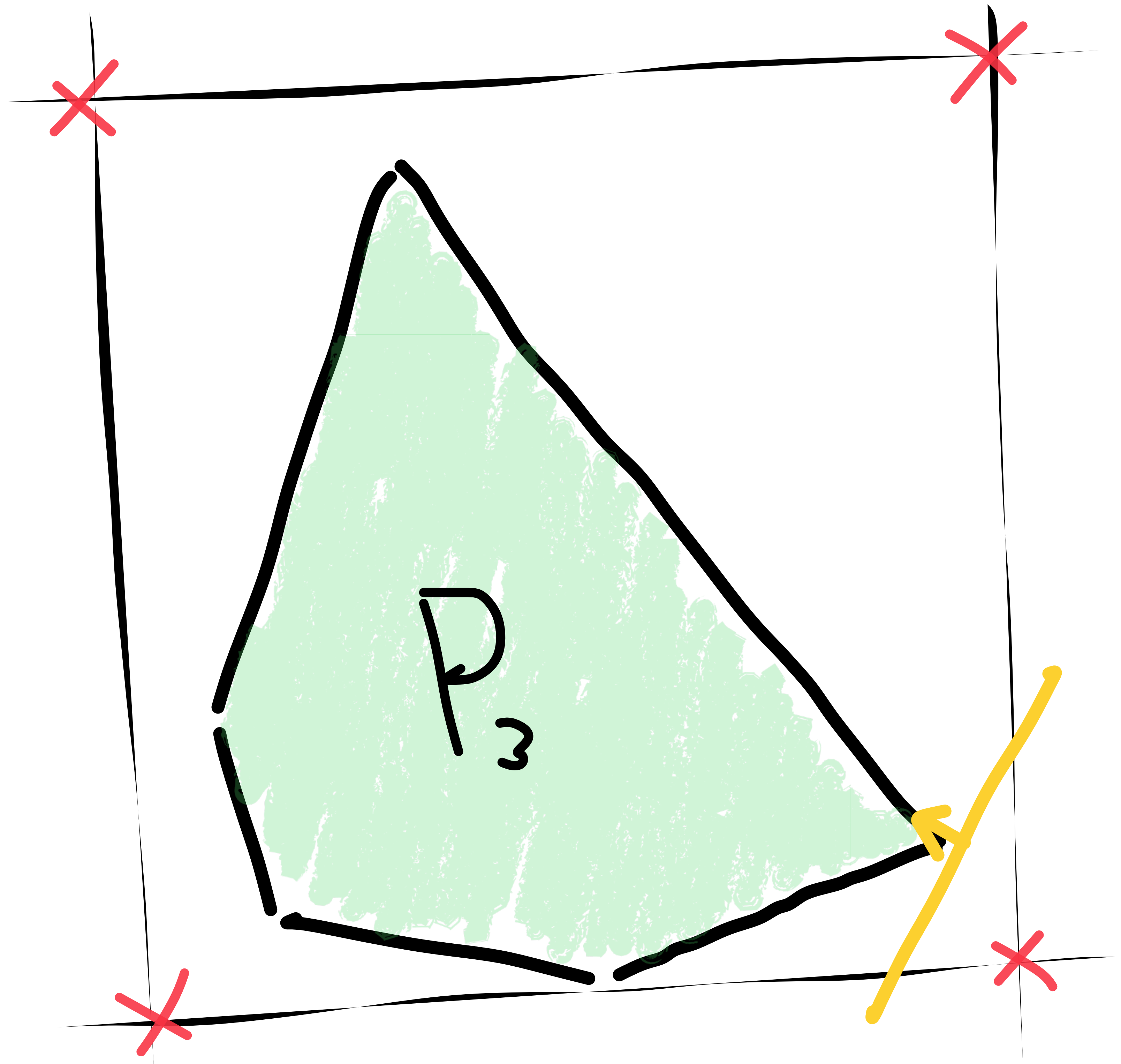
A CP proof that $P \cap \mathbb{Z}^n = \emptyset$ is a sequence of polytopes $P = P_1, \ldots, P_s = \emptyset$ s.t. P_{i+1} is deduced from P_i by a CG-cut



Cutting Planes

Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$

A CP proof that $P \cap \mathbb{Z}^n = \emptyset$ is a sequence of polytopes $P = P_1, \ldots, P_s = \emptyset$ s.t. P_{i+1} is deduced from P_i by a CG-cut

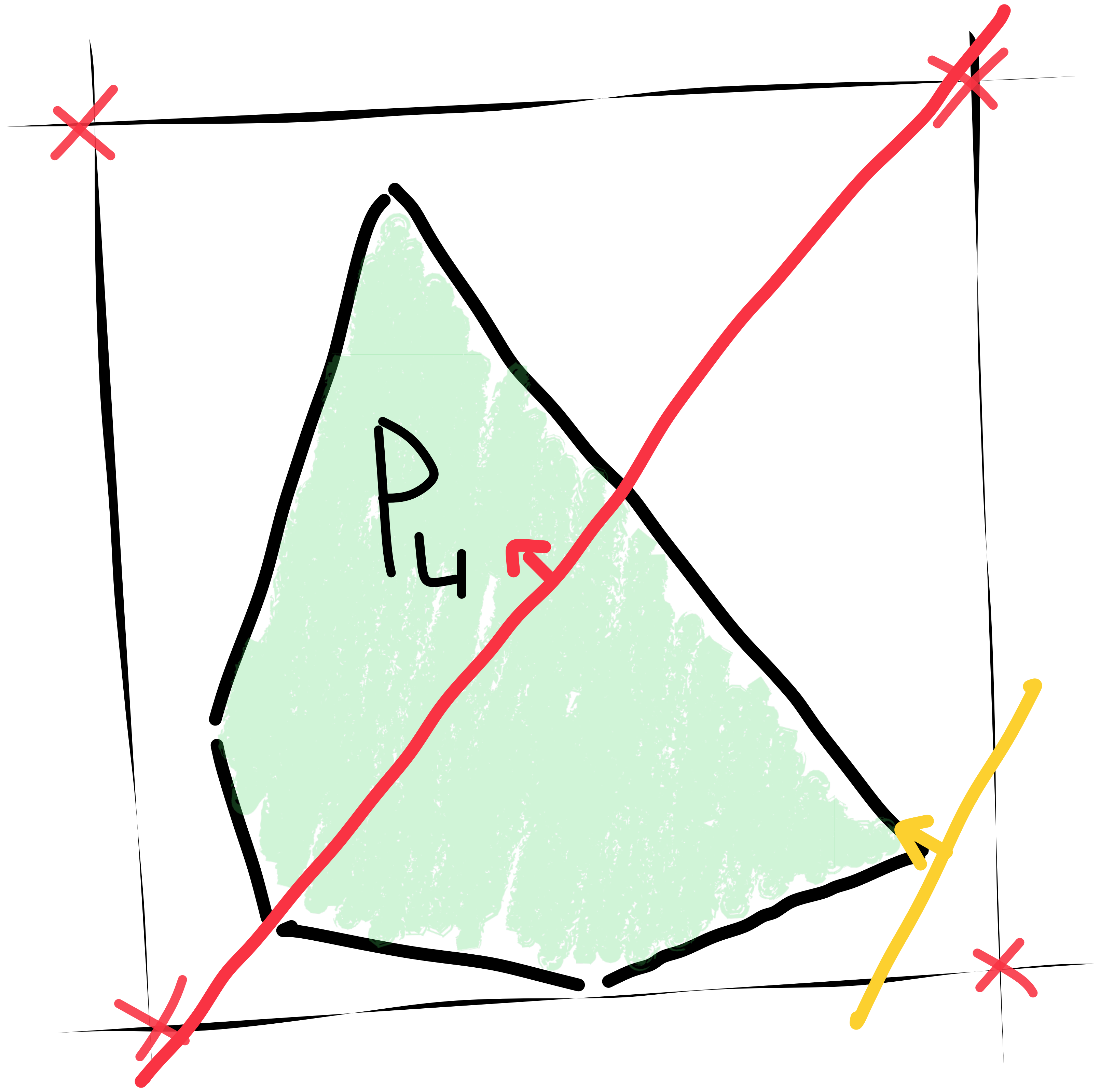


Cutting Planes

Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$

A CP proof that $P \cap \mathbb{Z}^n = \emptyset$ is a sequence of polytopes $P = P_1, \ldots, P_s = \emptyset$

s.t. P_{i+1} is deduced from P_i by a CG-cut

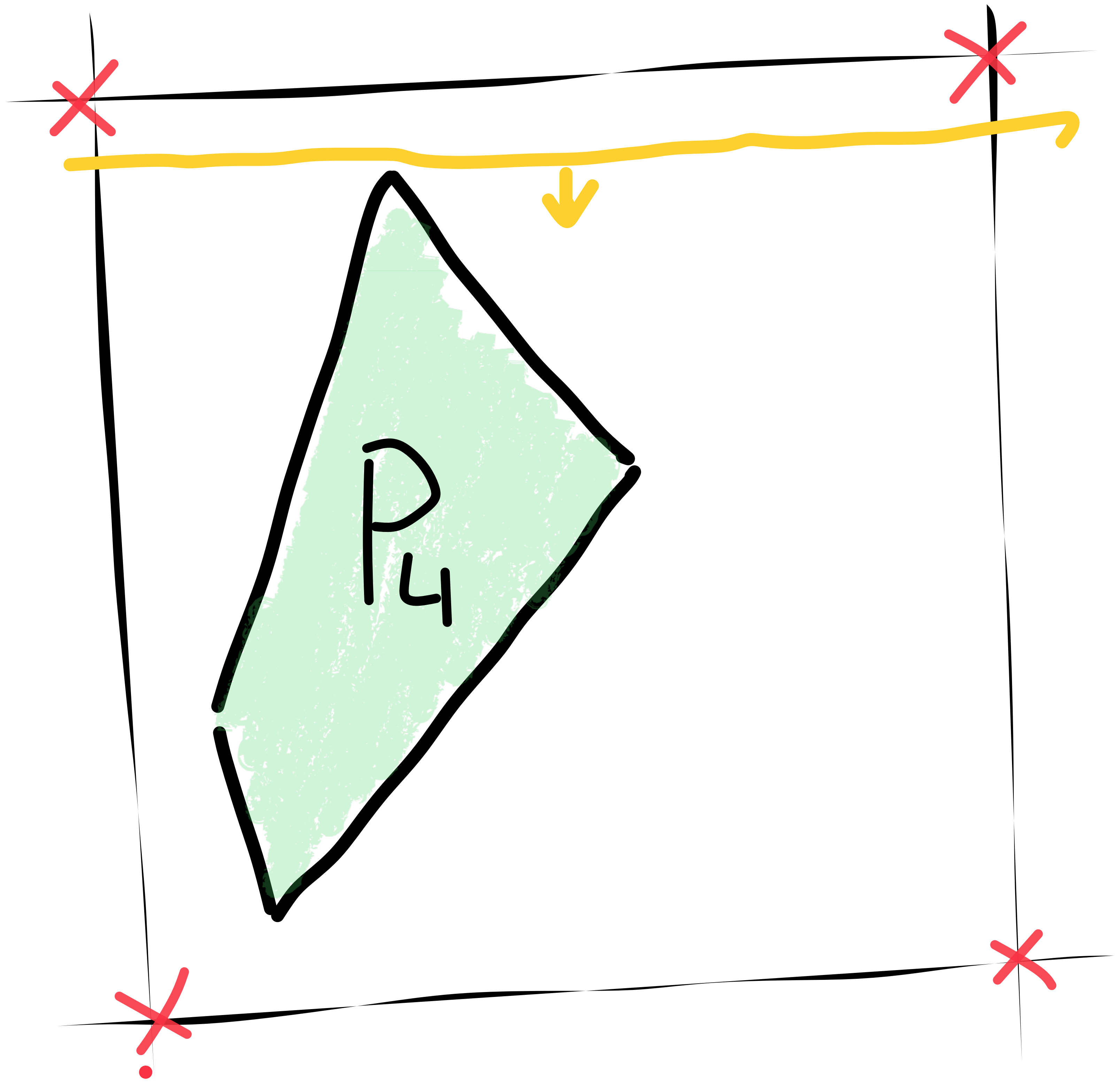


Cutting Planes

Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$

A CP proof that $P \cap \mathbb{Z}^n = \emptyset$ is a sequence of polytopes $P = P_1, \ldots, P_s = \emptyset$

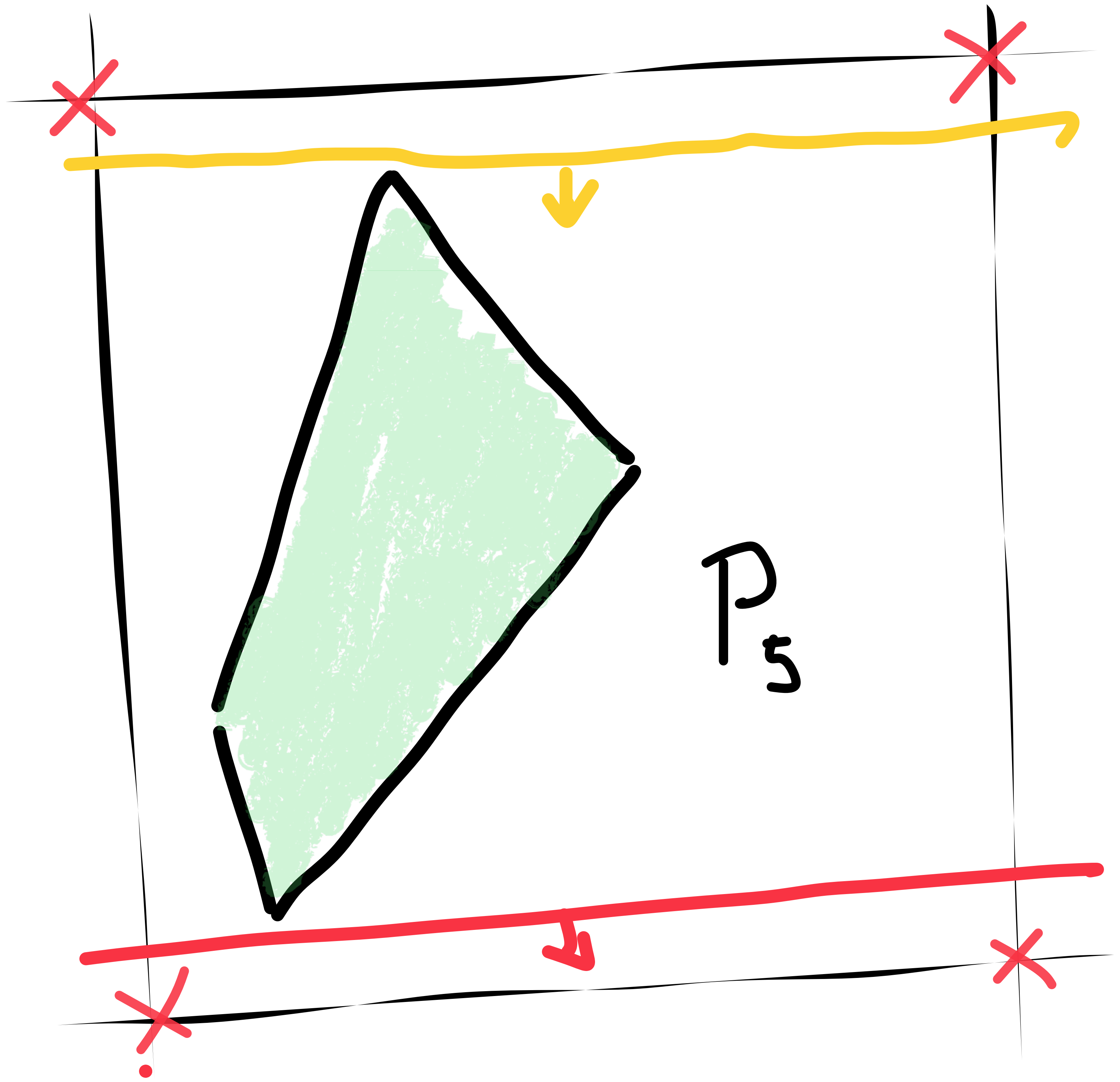
s.t. P_{i+1} is deduced from P_i by a CG-cut



Cutting Planes

Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$

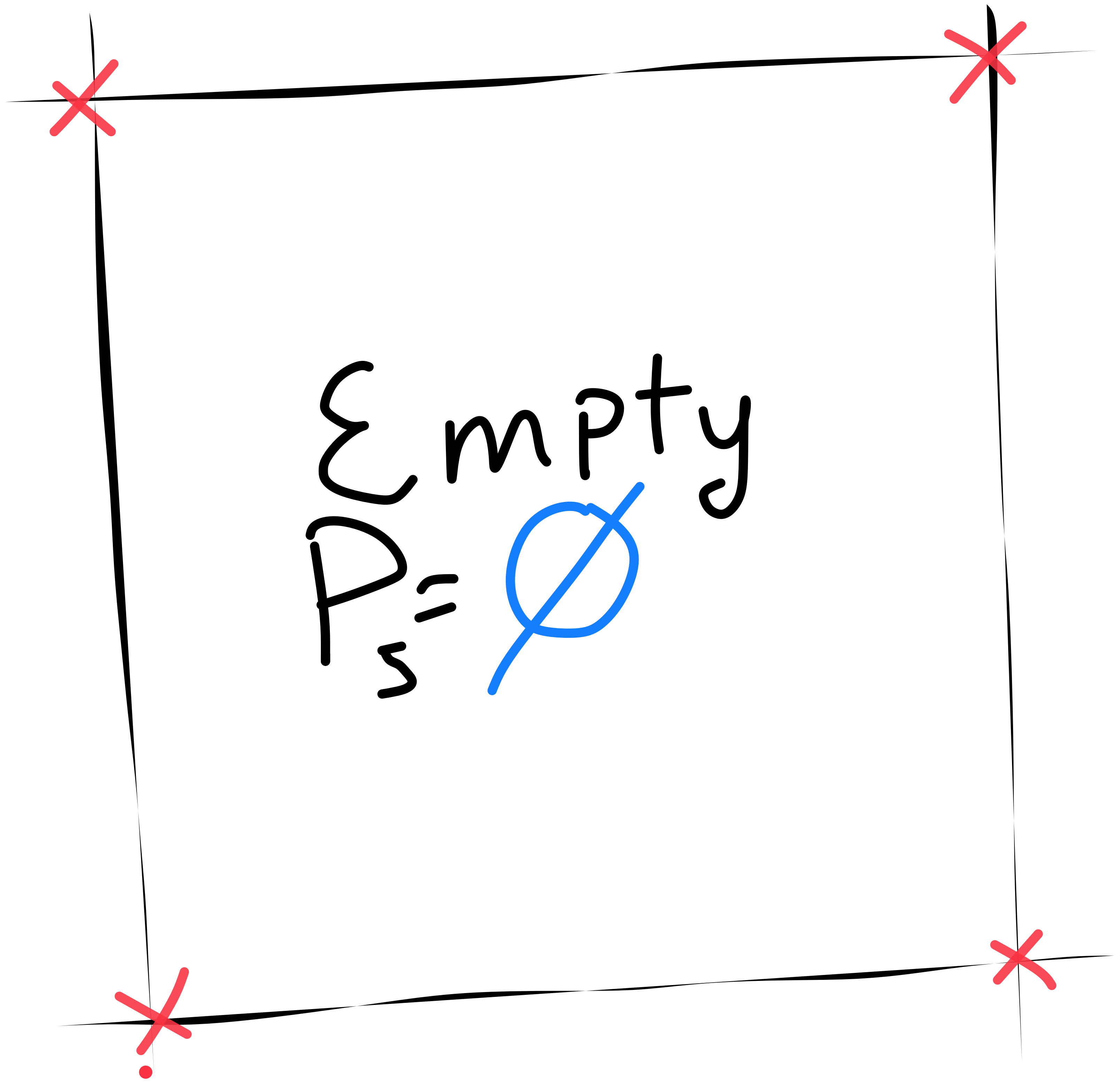
A CP proof that $P \cap \mathbb{Z}^n = \emptyset$ is a sequence of polytopes $P = P_1, \dots, P_s = \emptyset$ s.t. P_{i+1} is deduced from P_i by a CG-cut



Cutting Planes

Let $P = \{Ax \geq b\}$ be such that $P \cap \mathbb{Z}^n = \emptyset$

A CP proof that $P \cap \mathbb{Z}^n = \emptyset$ is a sequence of polytopes $P = P_1, \dots, P_s = \emptyset$ s.t. P_{i+1} is deduced from P_i by a CG-cut

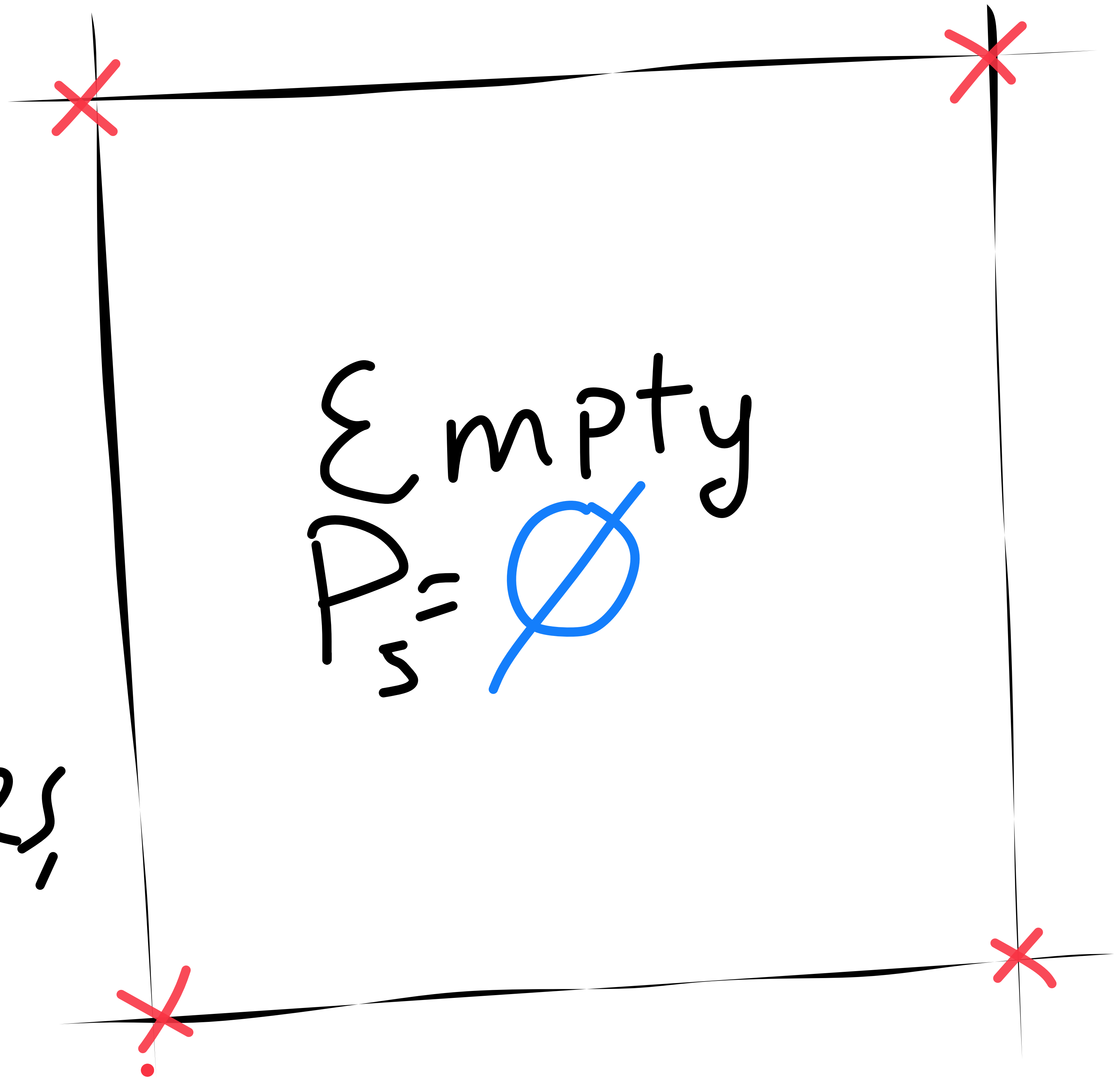


Cutting Planes

Let $P=\{Ax\geq b\}$ be such that $P\cap\mathbb{Z}^n=\emptyset$

A CP proof that $P\cap\mathbb{Z}^n=\emptyset$ is a sequence of polytopes $P=P_1,\dots,P_s=\emptyset$

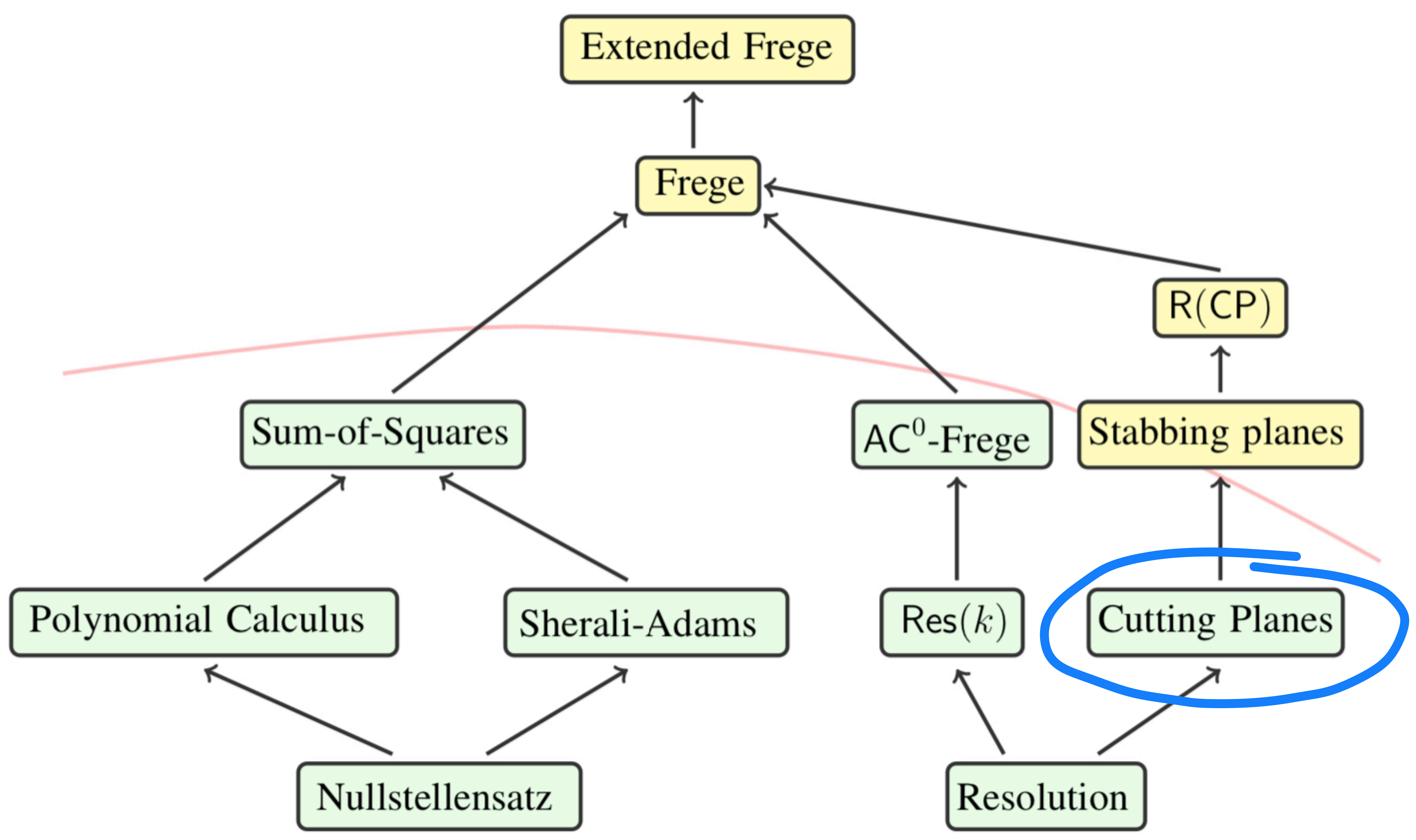
s.t. P_{i+1} is deduced from P_i by a CG-cut



Proof Size: the number of polytopes, s

Cutting Planes

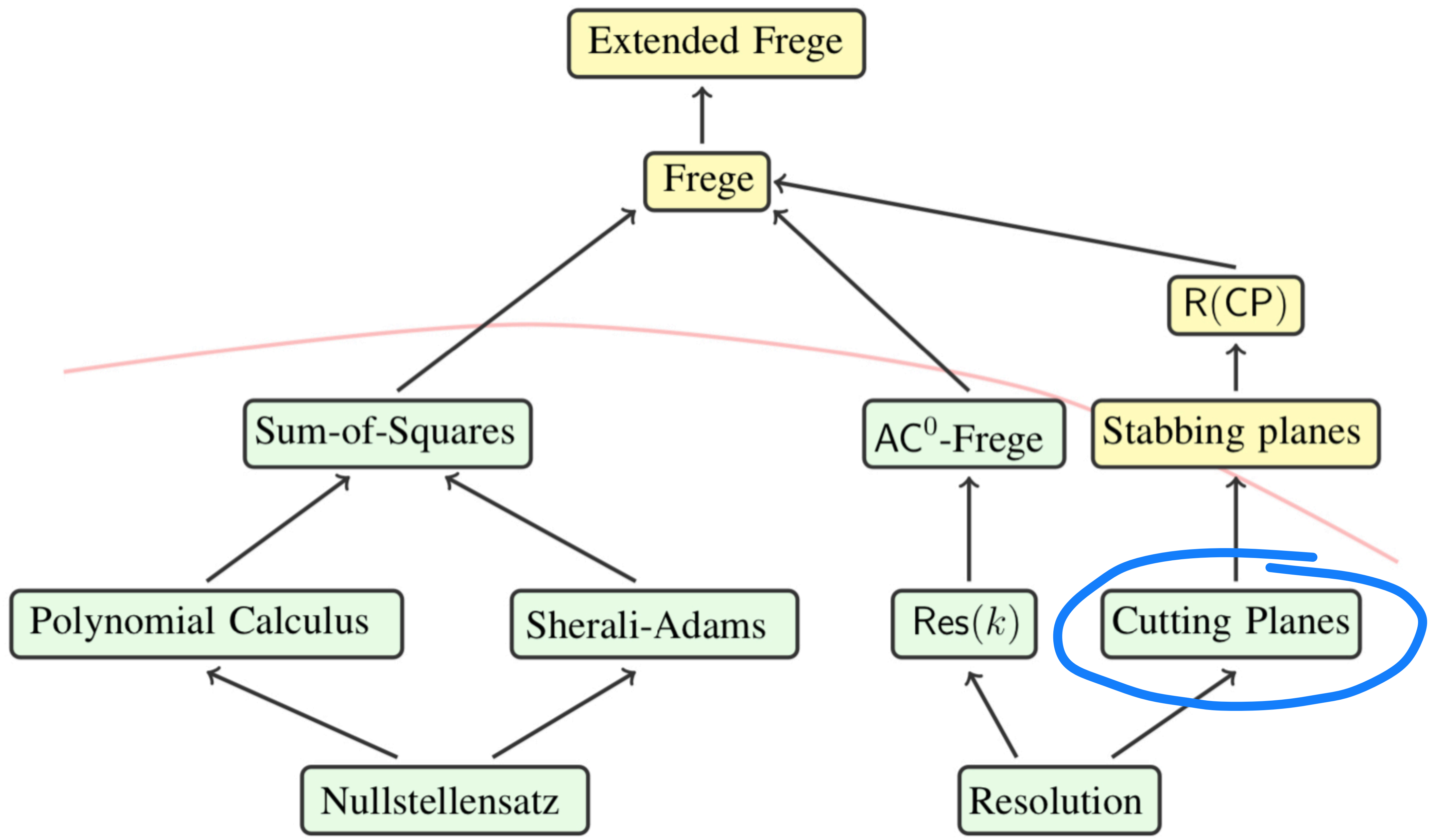
Powerful & Well-studied algebraic proof system



Cutting Planes

Powerful & Well-studied algebraic proof system

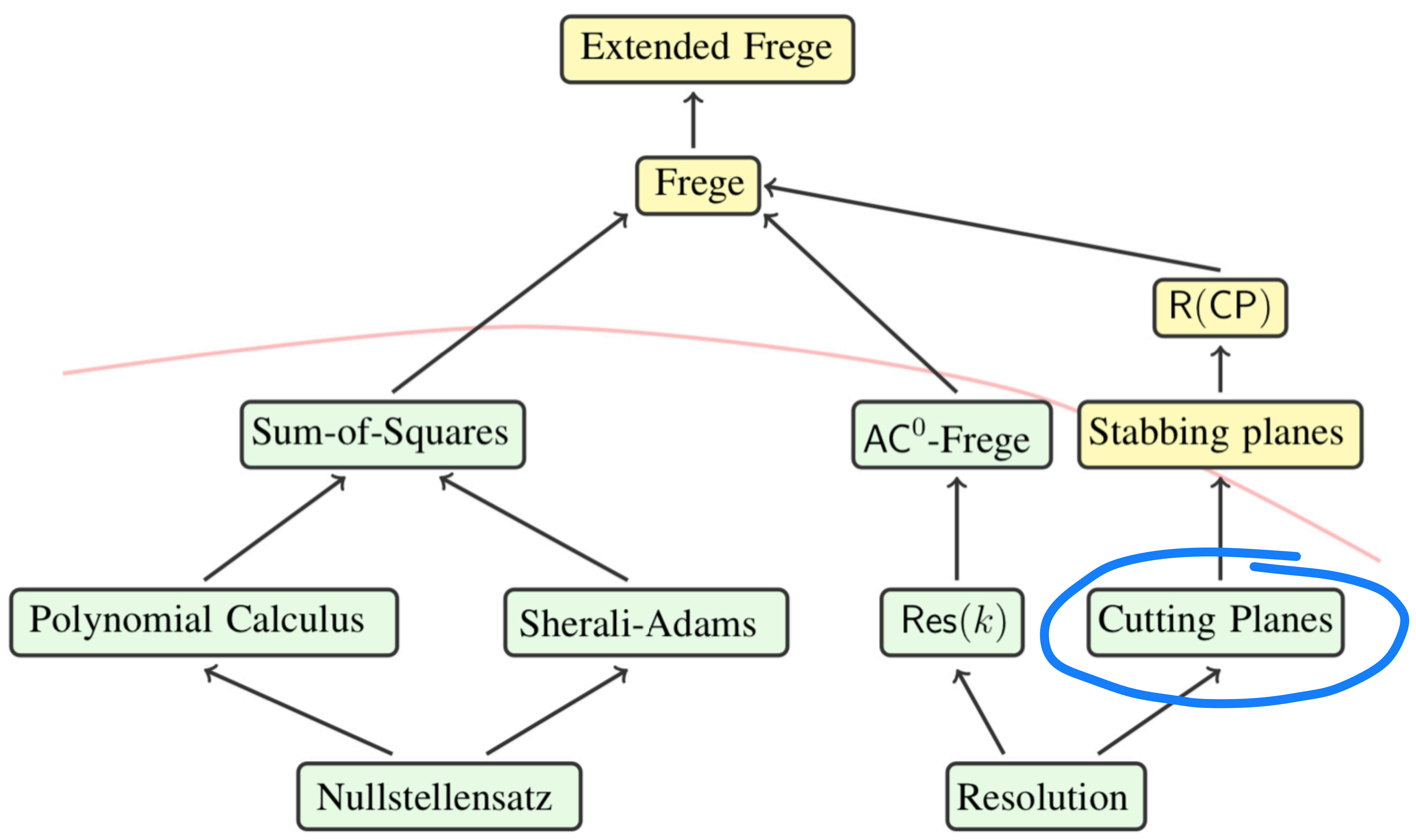
▷ Short proofs of PHP



Cutting Planes

Powerful & Well-studied algebraic proof system

- ▷ Short proofs of PHP
- ▷ Exponential lowerbounds [Pud97]



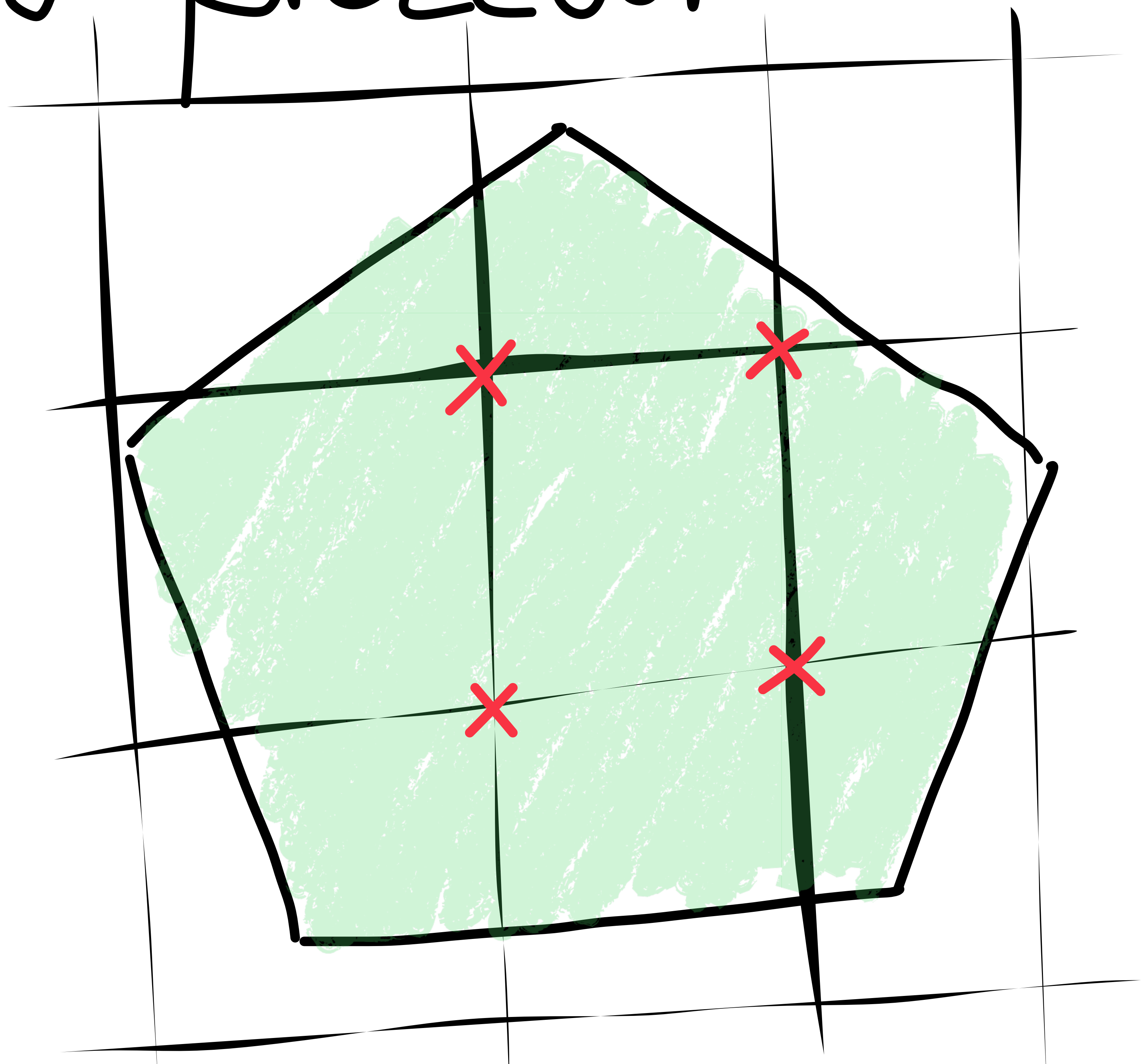
Integer Programming

Modern IP Algorithms combine cutting planes with a branch-and-bound procedure

Integer Programming

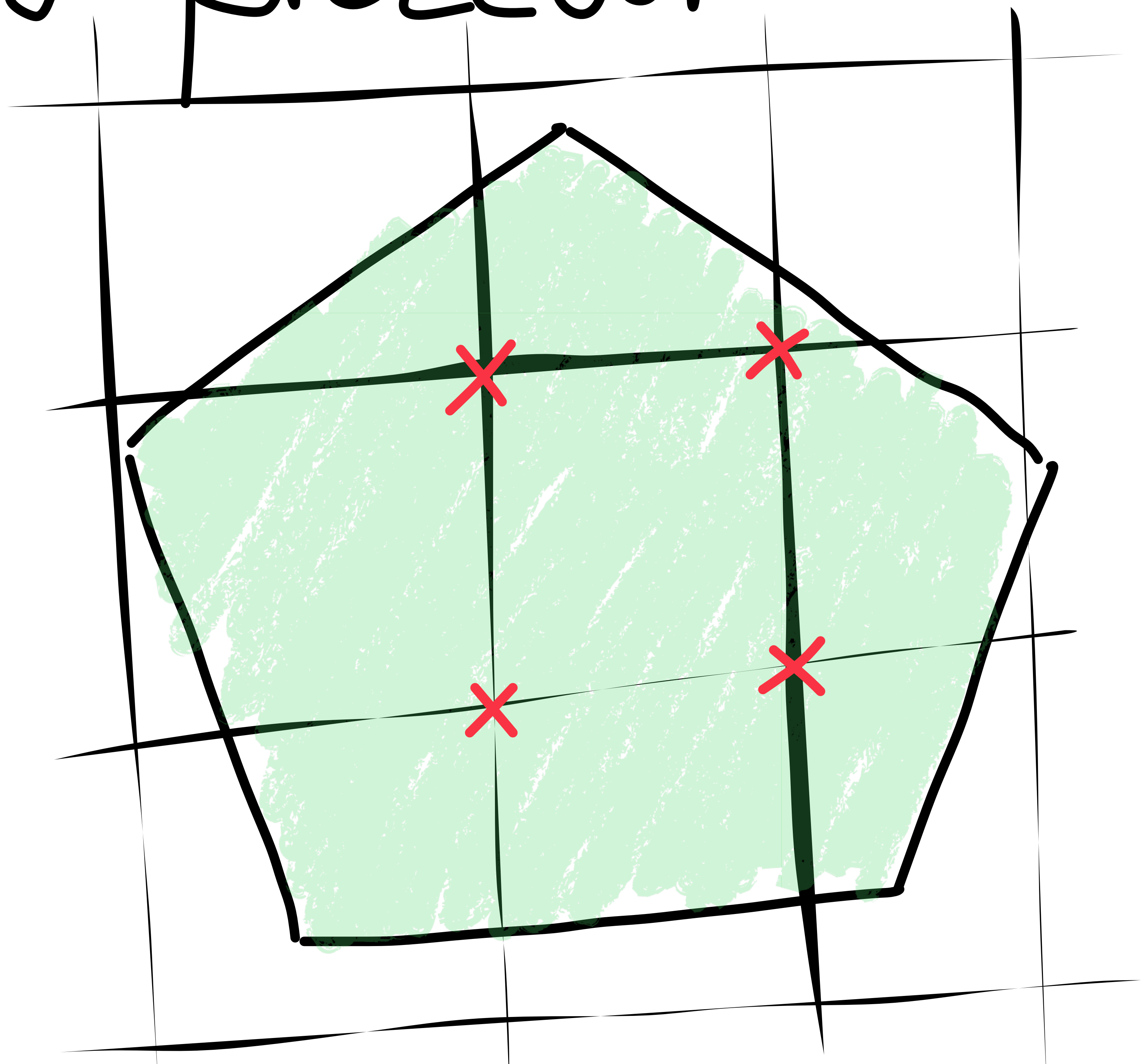
Modern IP Algorithms combine cutting planes with a branch-and-bound procedure

<u>Branch</u>-<u>and</u>-<u>Cut</u> On input P,



Integer Programming

Modern IP Algorithms combine cutting planes with a branch-and-bound procedure

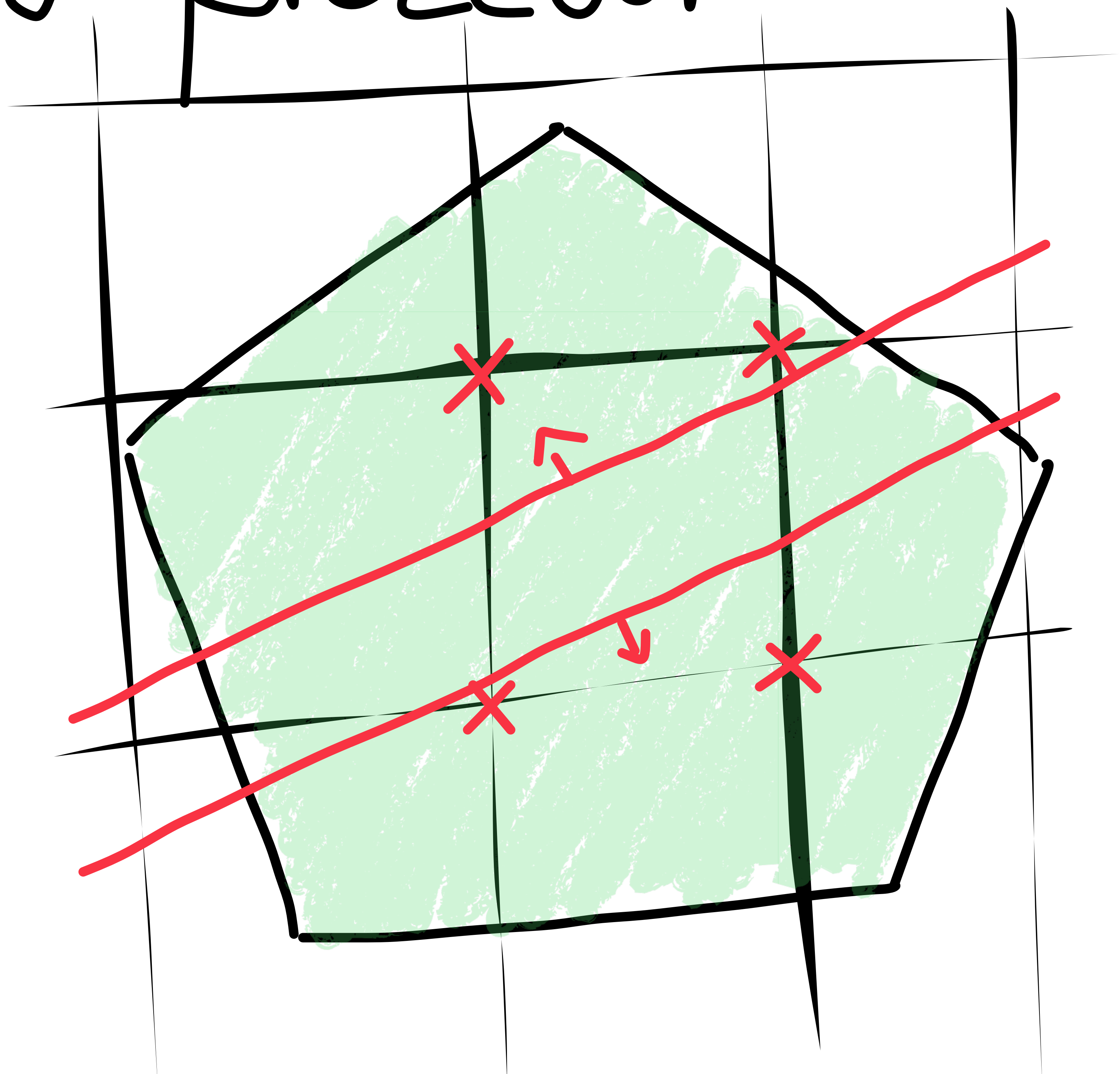


Branch-and-Cut on input P,

Branch: break P into $P_1, \dots, P_k$ s.t. $P \cap \mathbb{Z}^n \subseteq P_1 \cup \dots \cup P_k$

Integer Programming

Modern IP Algorithms combine cutting planes with a branch-and-bound procedure



Branch-and-Cut on input P,

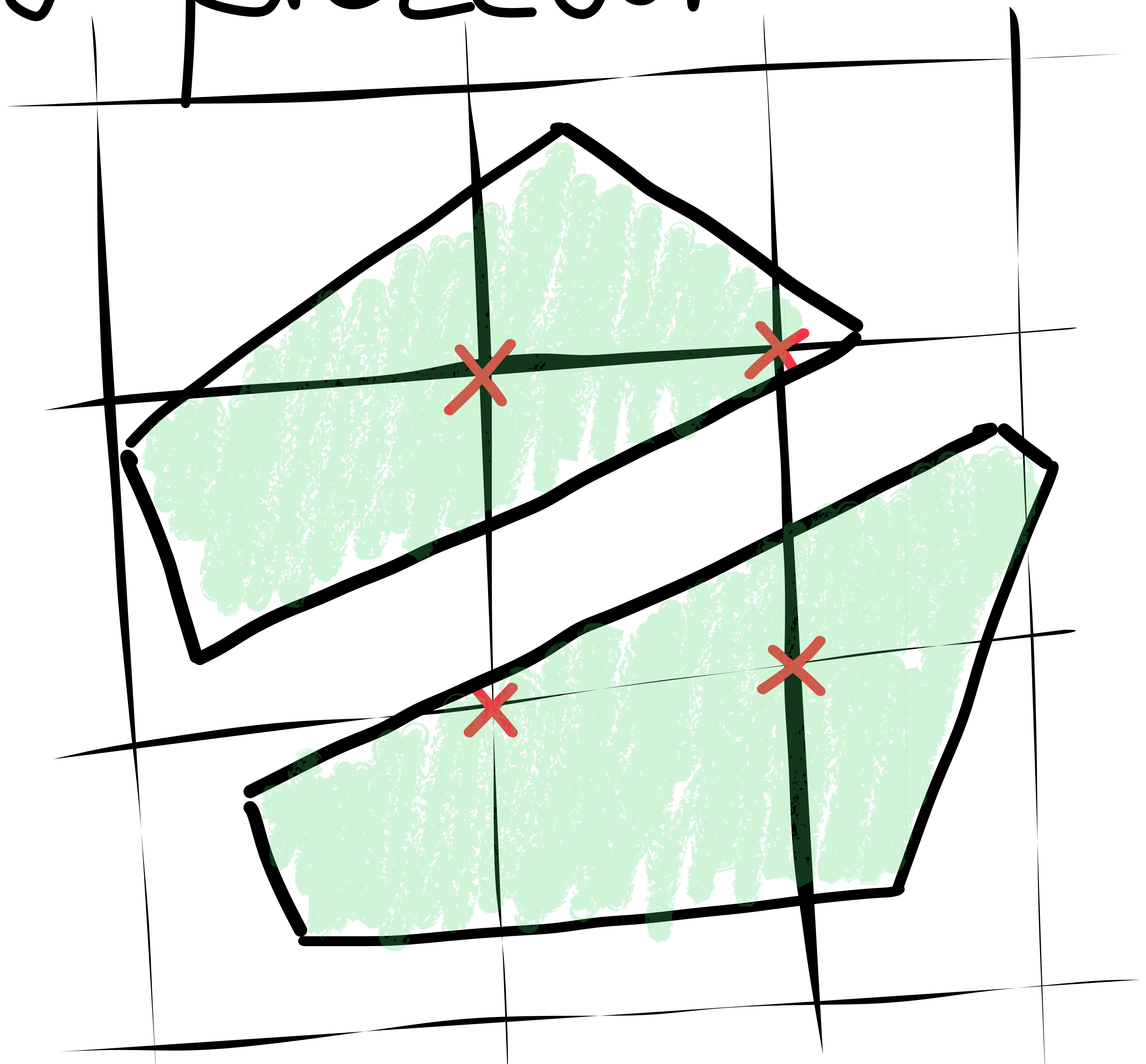
Branch: break P into $P_1, \dots, P_k$ s.t. $P \cap \mathbb{Z}^n \subseteq P_1 \cup \dots \cup P_k$

Integer Programming

Modern IP Algorithms combine cutting planes with a branch-and-bound procedure

Branch-and-Cut on input P,

Branch: break P into $P_1, \dots, P_k$ s.t. $P \cap \mathbb{Z}^n \subseteq P_1 \cup \dots \cup P_k$

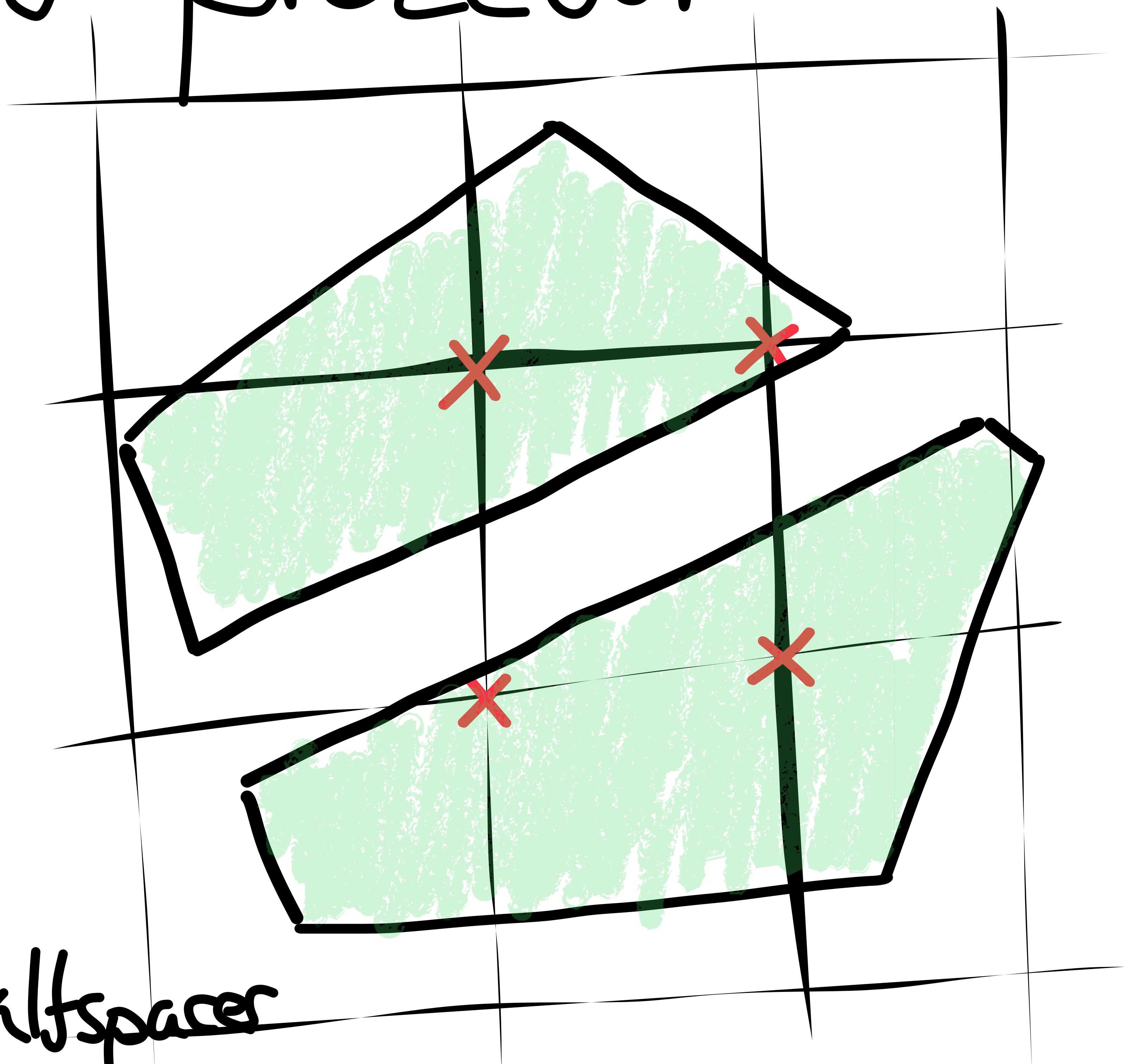


Integer Programming

Modern IP Algorithms combine cutting planes with a branch-and-bound procedure

Branch-and-Cut on input P,

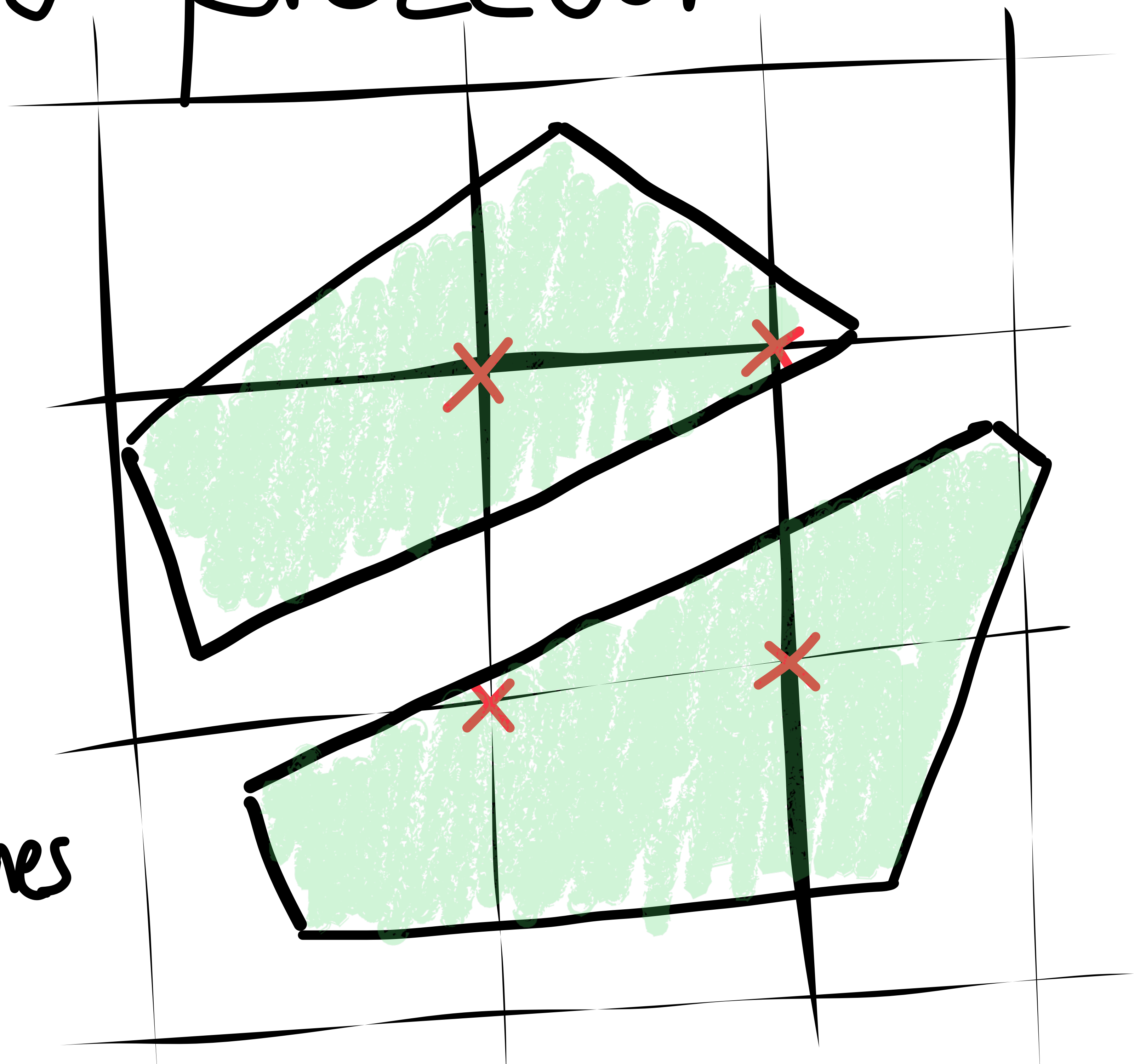
Branch: break P into $P_1, \ldots, P_k$ s.t. $P \cap \mathbb{Z}^n \subseteq P_1 \cup \ldots \cup P_k$



▷ In practice, P is broken into $P \cap \{x : ax \geq b\}$ and $P \cap \{x : ax \leq b-1\}$ for some class of halfspaces

Integer Programming

Modern IP Algorithms combine cutting planes with a branch-and-bound procedure



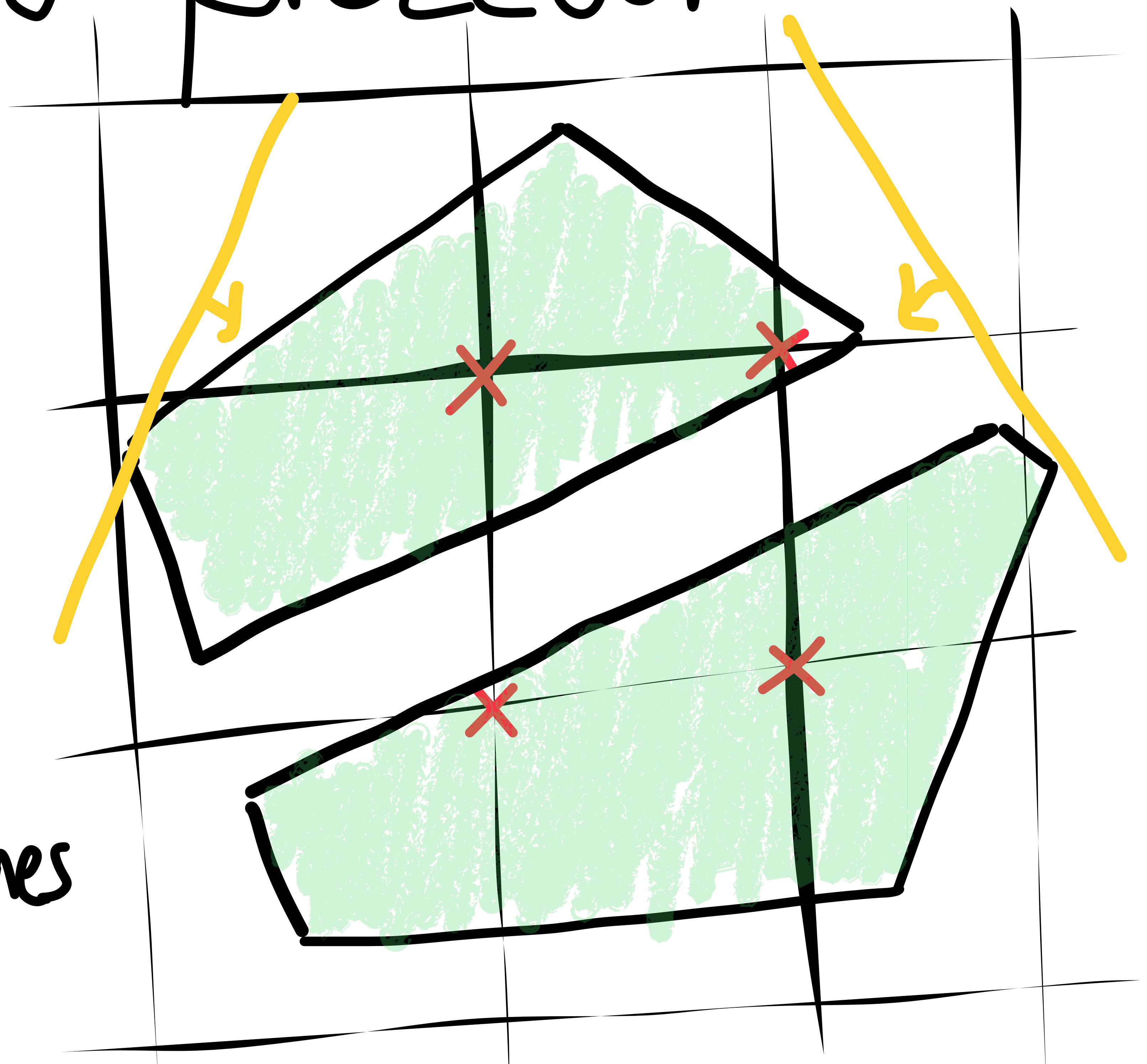
Branch-and-Cut On input P,

Branch: break P into $P_1, \dots, P_k$ s.t. $P \cap \mathbb{Z}^n \subseteq P_1 \cup \dots \cup P_k$

Cut: Refine $P_1, \dots, P_k$ with additional cutting planes

Integer Programming

Modern IP Algorithms combine cutting planes with a branch-and-bound procedure



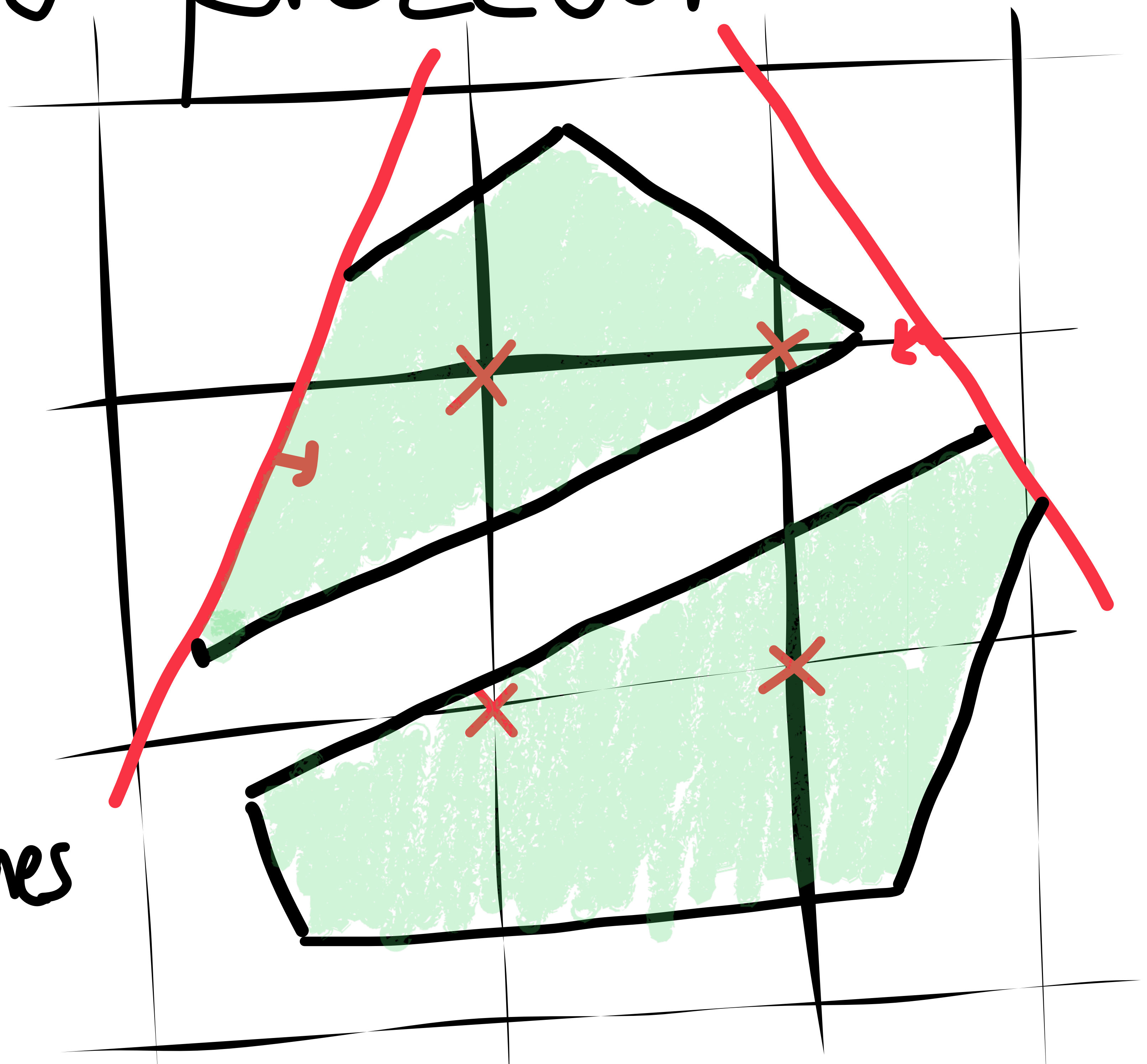
Branch-and-Cut on input P,

Branch: break P into $P_1, \dots, P_k$ s.t. $P \cap \mathbb{Z}^n \subseteq P_1 \cup \dots \cup P_k$

Cut: Refine $P_1, \dots, P_k$ with additional cutting planes

Integer Programming

Modern IP Algorithms combine cutting planes with a branch-and-bound procedure



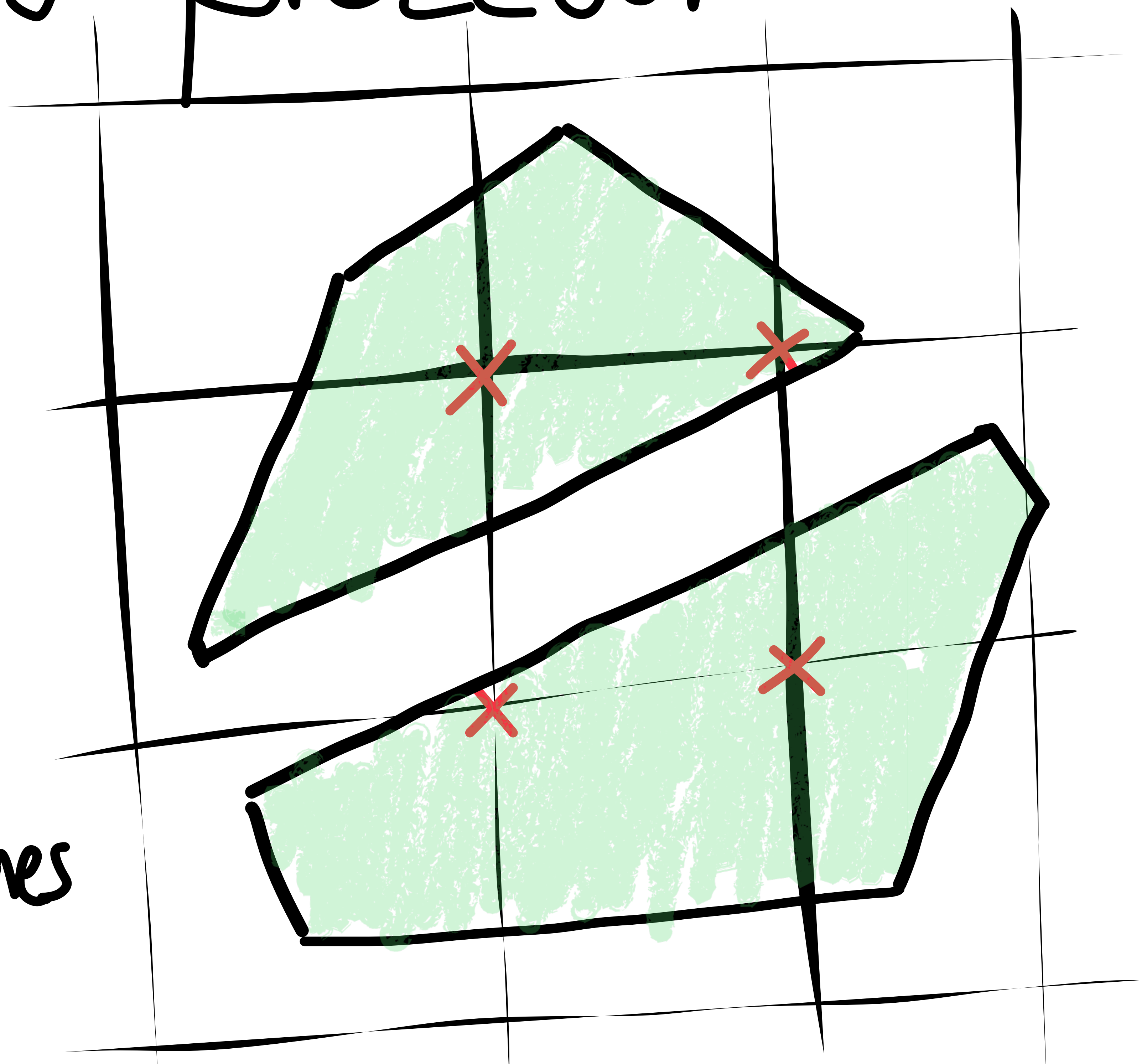
Branch-and-Cut On input P,

Branch: break P into $P_1, \dots, P_k$ s.t. $P \cap \mathbb{Z}^n \subseteq P_1 \cup \dots \cup P_k$

Cut: Refine $P_1, \dots, P_k$ with additional cutting planes

Integer Programming

Modern IP Algorithms combine cutting planes with a branch-and-bound procedure



Branch-and-Cut On input P,

Branch: break P into $P_1, \dots, P_k$ s.t. $P \cap \mathbb{Z}^n \subseteq P_1 \cup \dots \cup P_k$

Cut: Refine $P_1, \dots, P_k$ with additional cutting planes

Repeat

Stabbing Planes [BFI+18]

- Formalizes practical branch-and-cut as a proof system

Stabbing Planes [BFI+18]

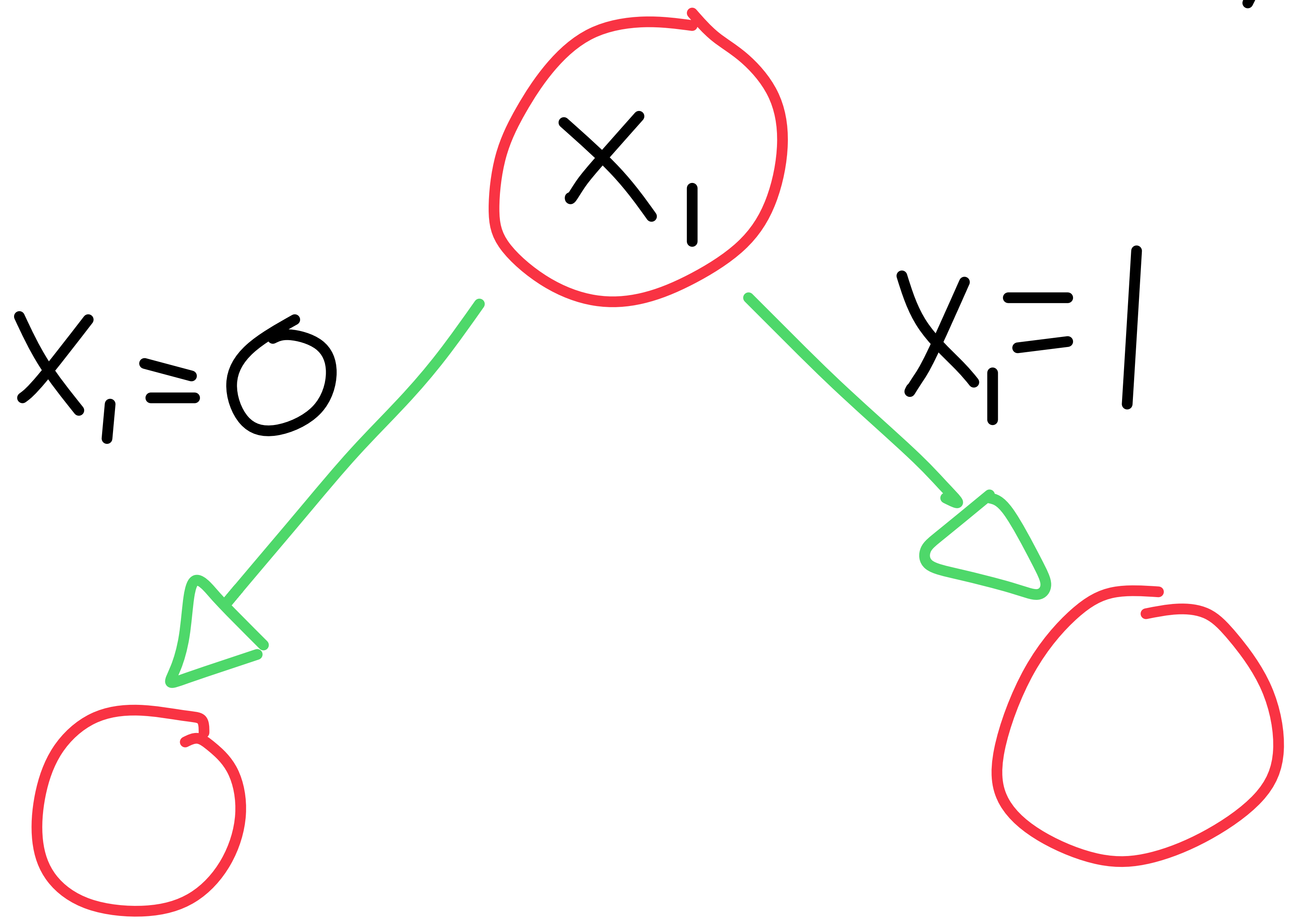
- Formalizes practical branch-and-cut as a proof system
- Extends DPLL to reason about linear inequalities

DPLL Refutation

$$\{x_1 \vee x_2,\ \bar{x}_1 \vee x_2,\ x_1 \vee \bar{x}_2,\ \bar{x}_1 \vee \bar{x}_2\}$$

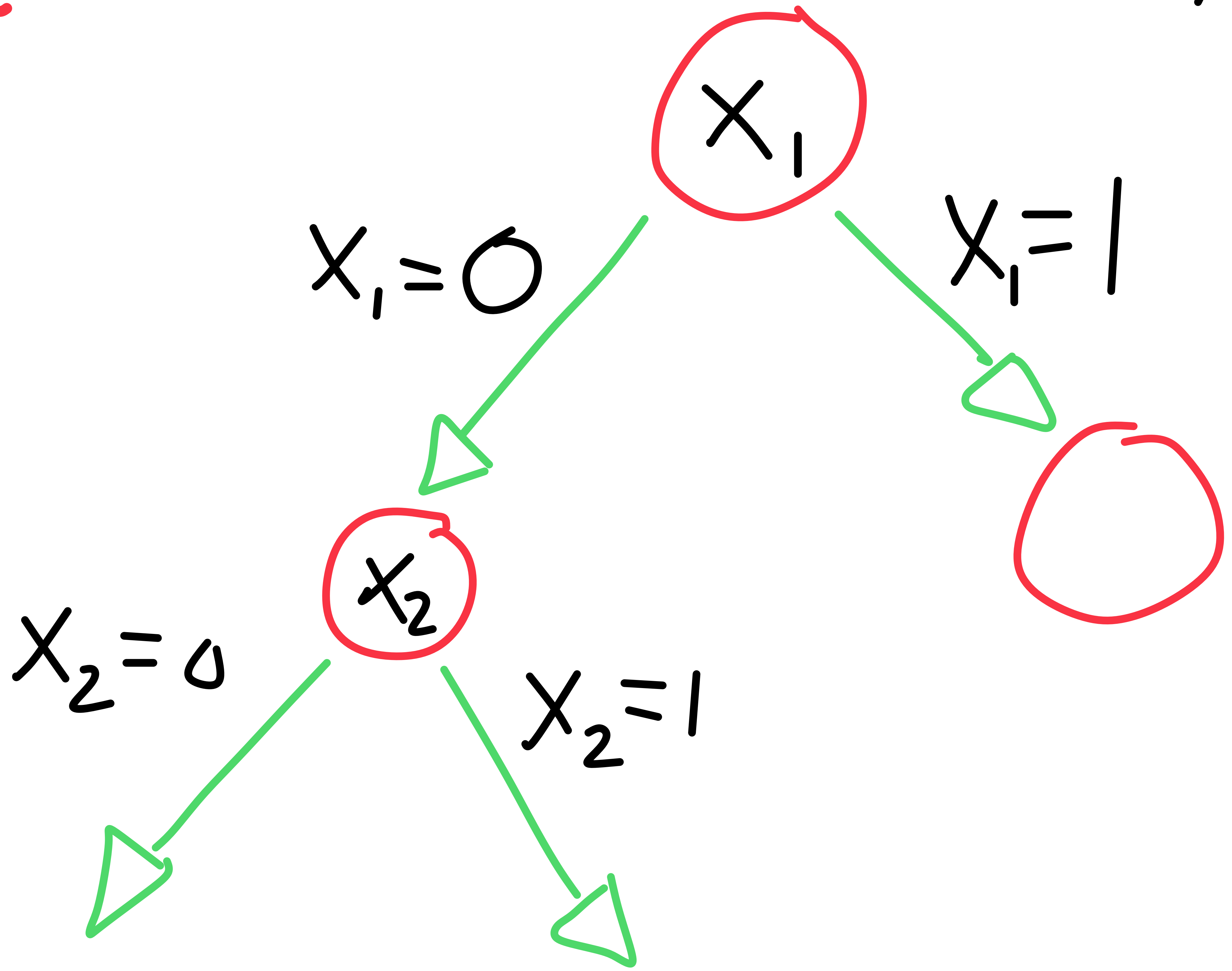
DPLL Refutation

$$\{x_1 \vee x_2,\ \bar{x}_1 \vee x_2,\ x_1 \vee \bar{x}_2,\ \bar{x}_1 \vee \bar{x}_2\}$$



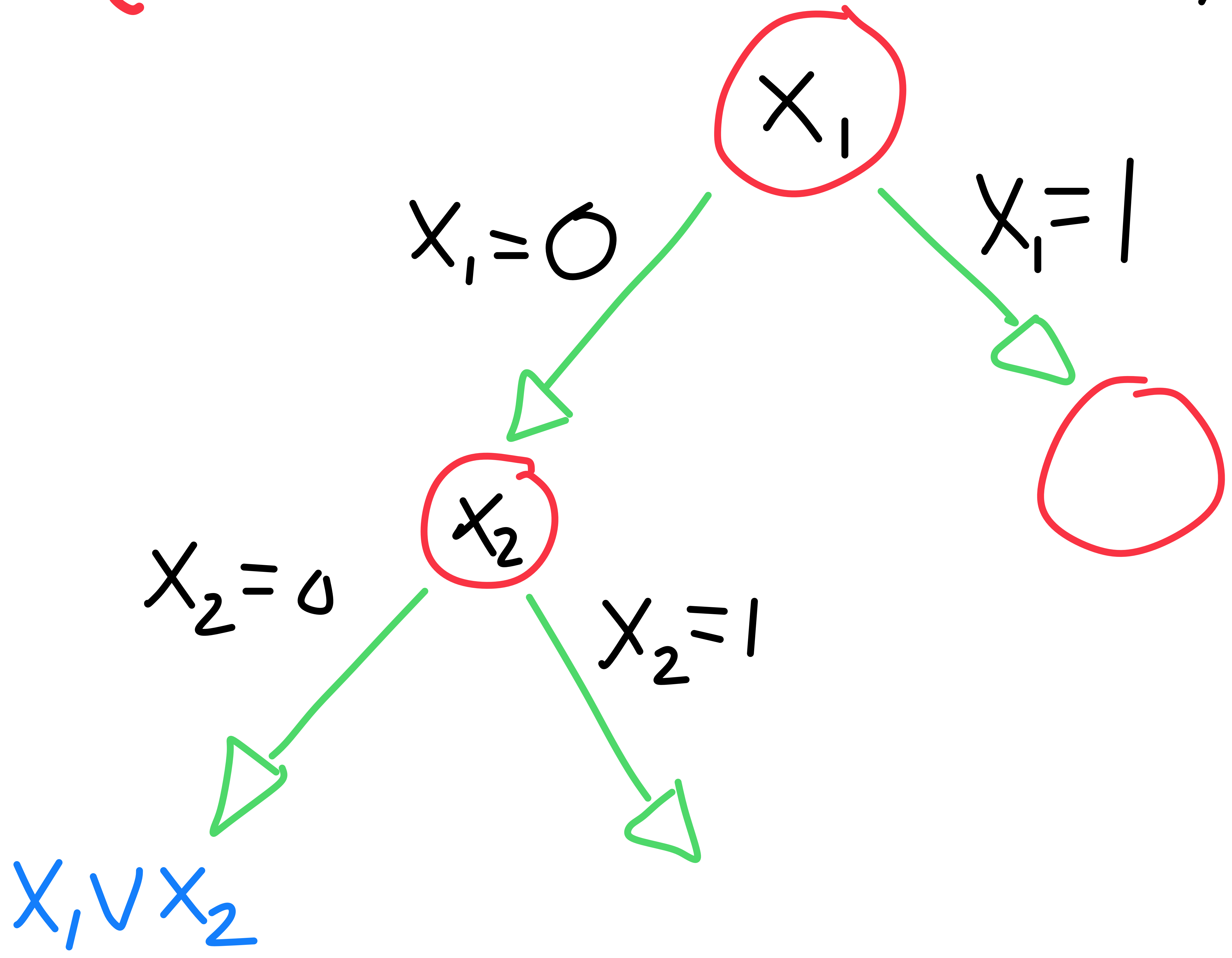
DPLL Refutation

$$\{x_1 \vee x_2,\ \bar{x}_1 \vee x_2,\ x_1 \vee \bar{x}_2,\ \bar{x}_1 \vee \bar{x}_2\}$$



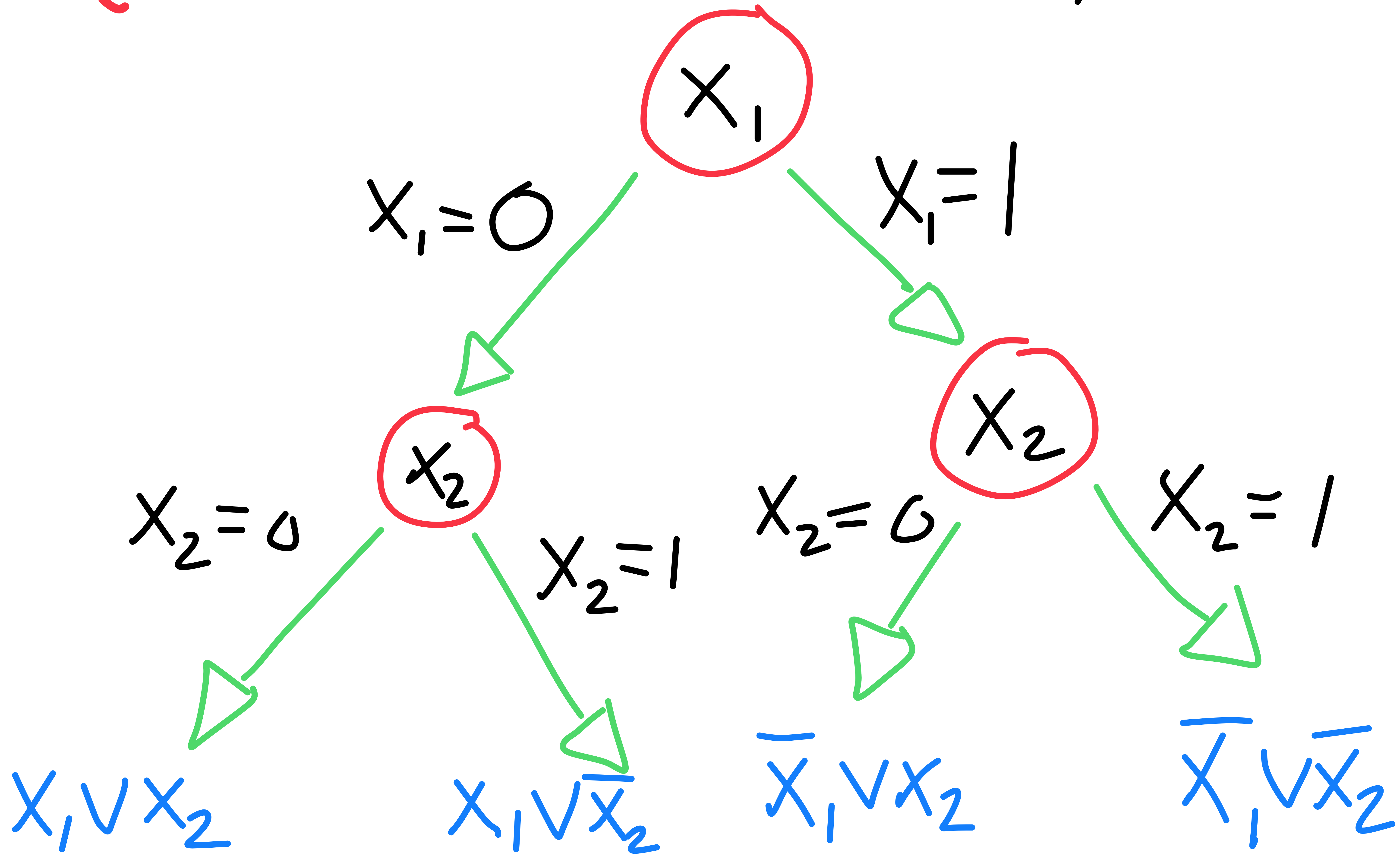
DPLL Refutation

$\{x_1 \vee x_2,\ \bar{x}_1 \vee x_2,\ x_1 \vee \bar{x}_2,\ \bar{x}_1 \vee \bar{x}_2\}$



DPLL Refutation

$$\{x_1 \vee x_2,\ \bar{x}_1 \vee x_2,\ x_1 \vee \bar{x}_2,\ \bar{x}_1 \vee \bar{x}_2\}$$



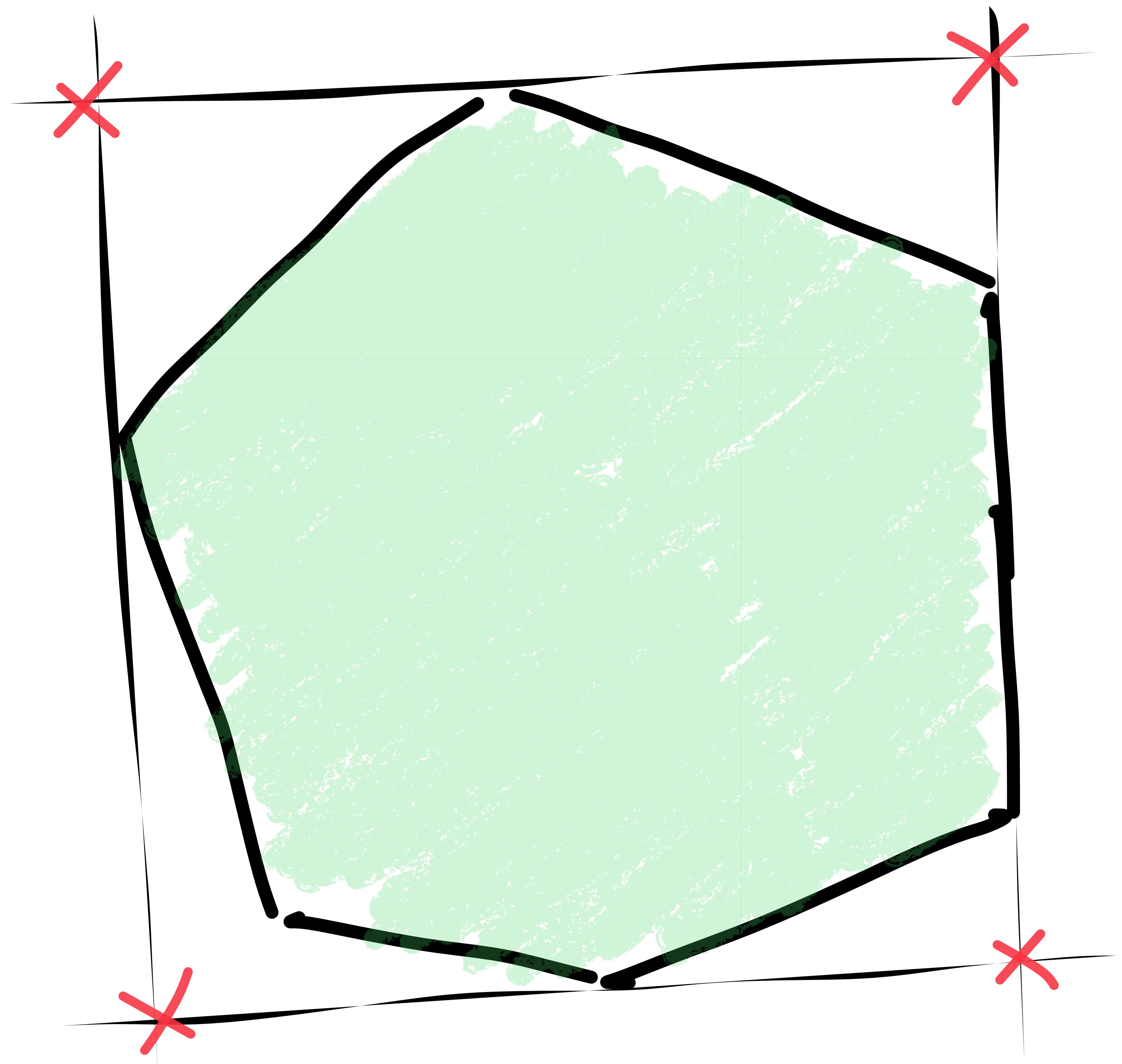
DPLL as Polytopes

$$P = \{ x_1 + x_2 \geq 1,\ x_1 - x_2 \geq 0,\ x_2 - x_1 \geq 0,\ -x_1 - x_2 \geq -1,\ 0 \leq x_i \leq 1 \}$$

DPLL as Polytopes

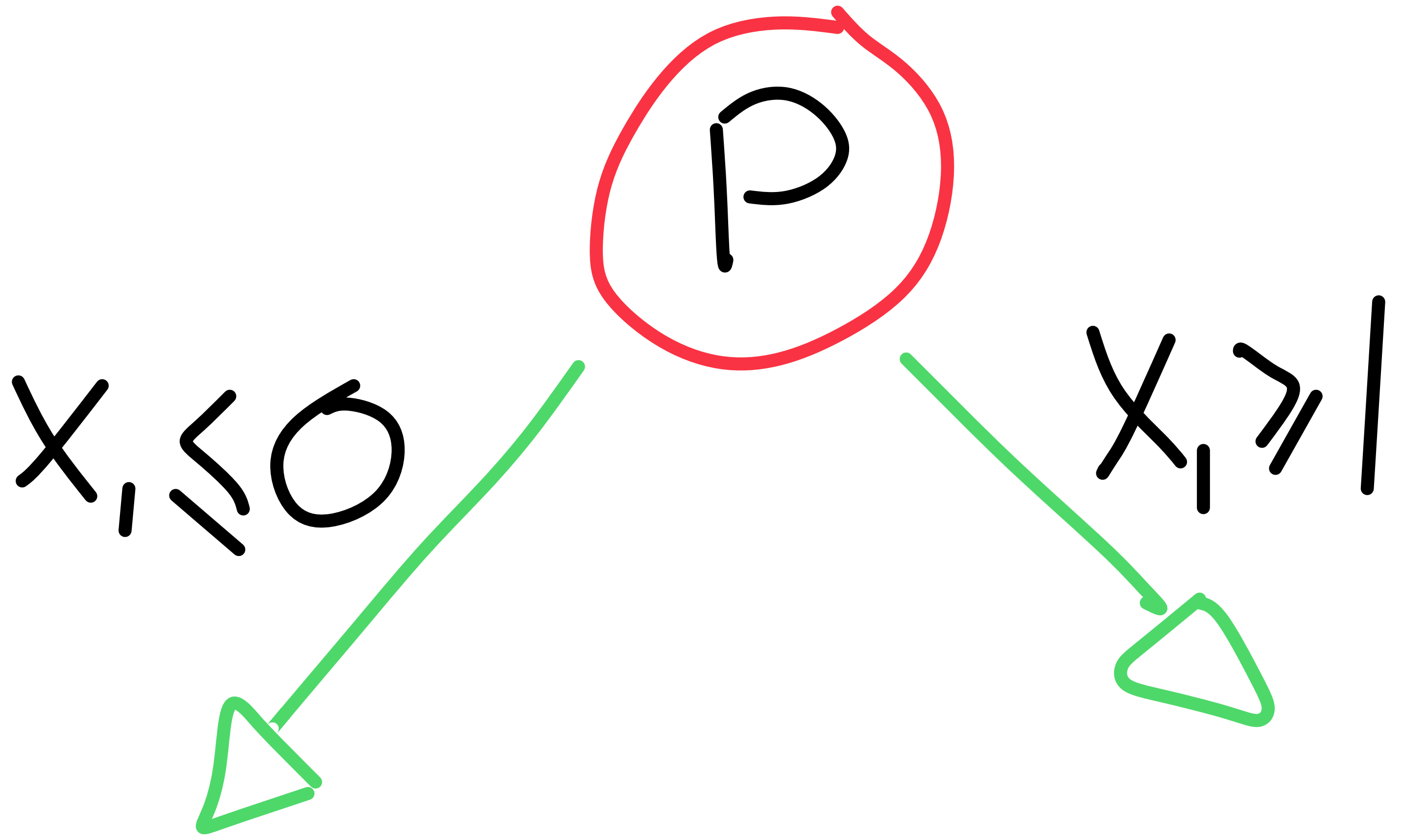
$$P = \{ x_1 + x_2 \geq 1,\ x_1 - x_2 \geq 0,\ x_2 - x_1 \geq 0,\ -x_1 - x_2 \geq -1,\ 0 \leq x_i \leq 1 \}$$

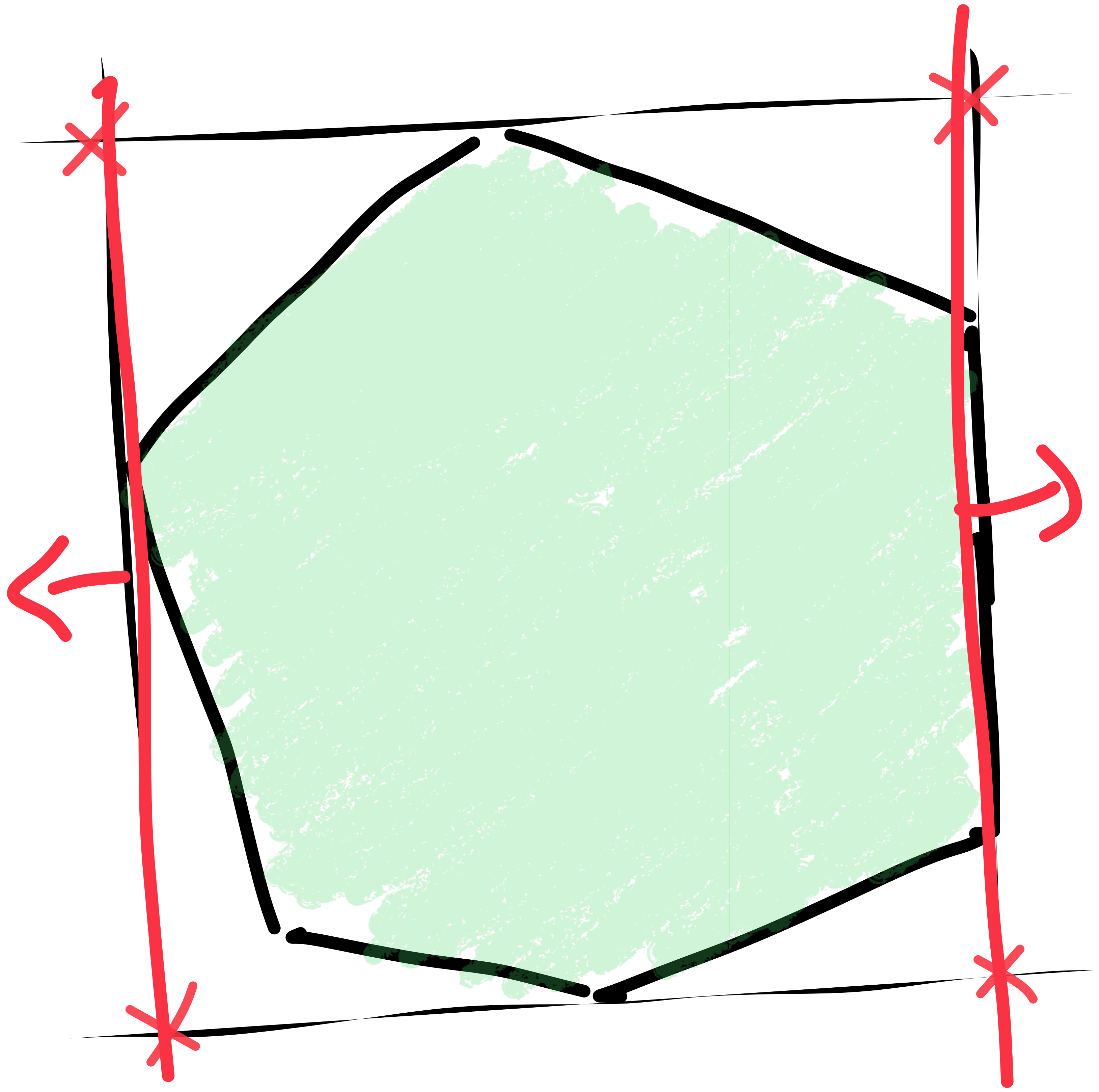
P



DPLL as Polytopes

$$P = \{ x_1 + x_2 \geq 1,\ x_1 - x_2 \geq 0,\ x_2 - x_1 \geq 0,\ -x_1 - x_2 \geq -1,\ 0 \leq x_i \leq 1 \}$$

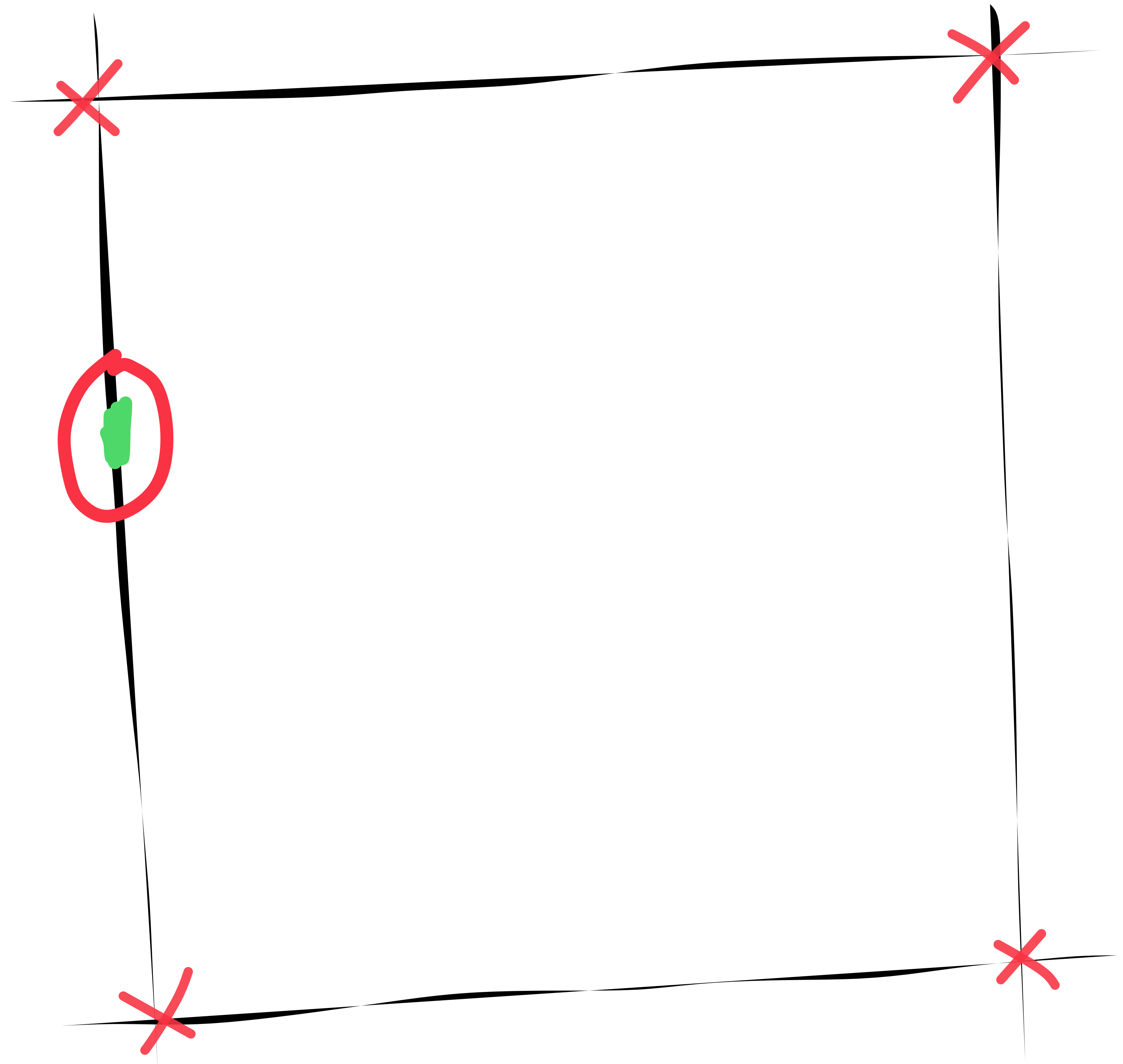




DPLL as Polytopes

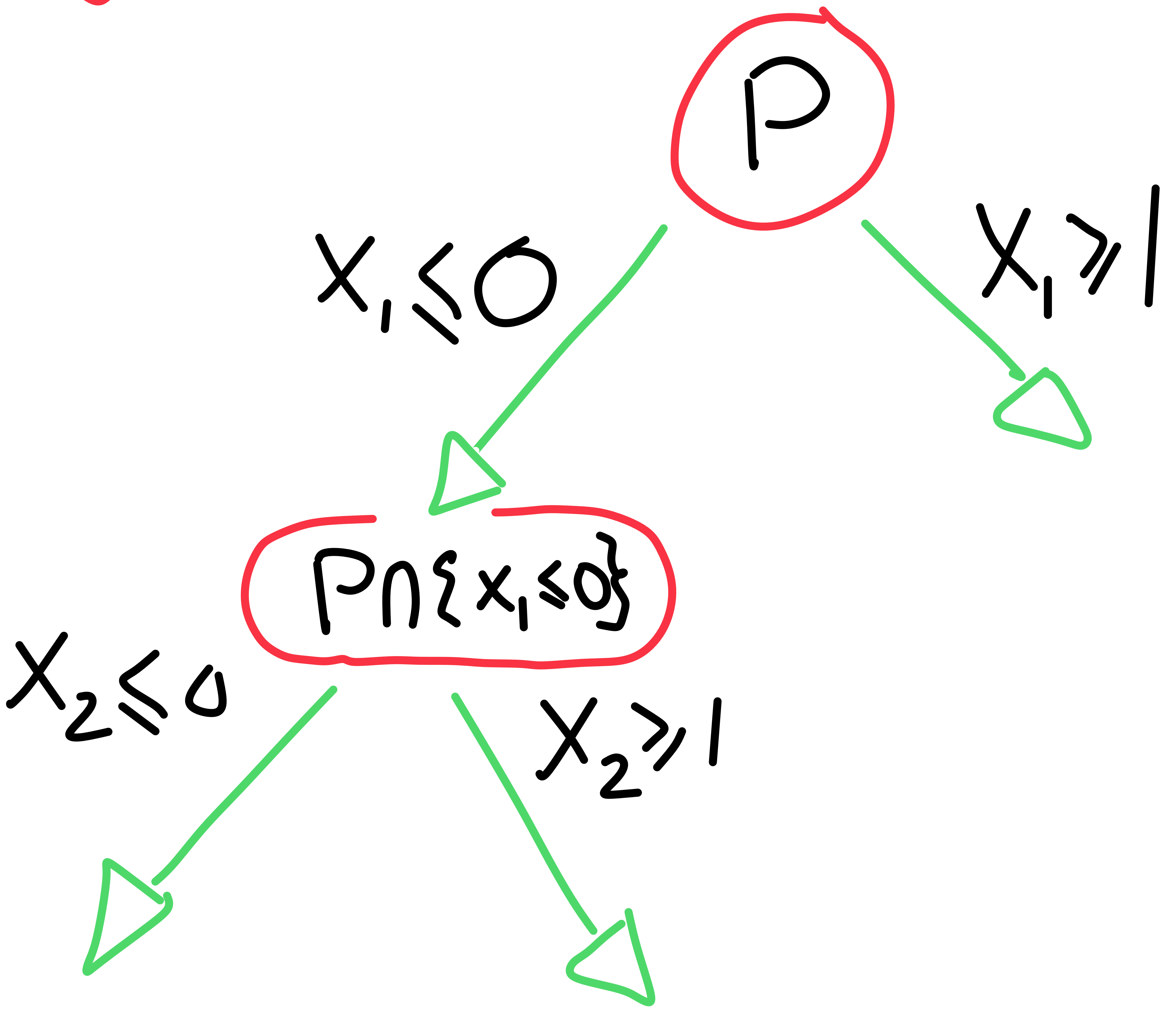
$$P = \{ x_1 + x_2 \geq 1,\ x_1 - x_2 \geq 0,\ x_2 - x_1 \geq 0,\ -x_1 - x_2 \geq -1,\ 0 \leq x_i \leq 1 \}$$

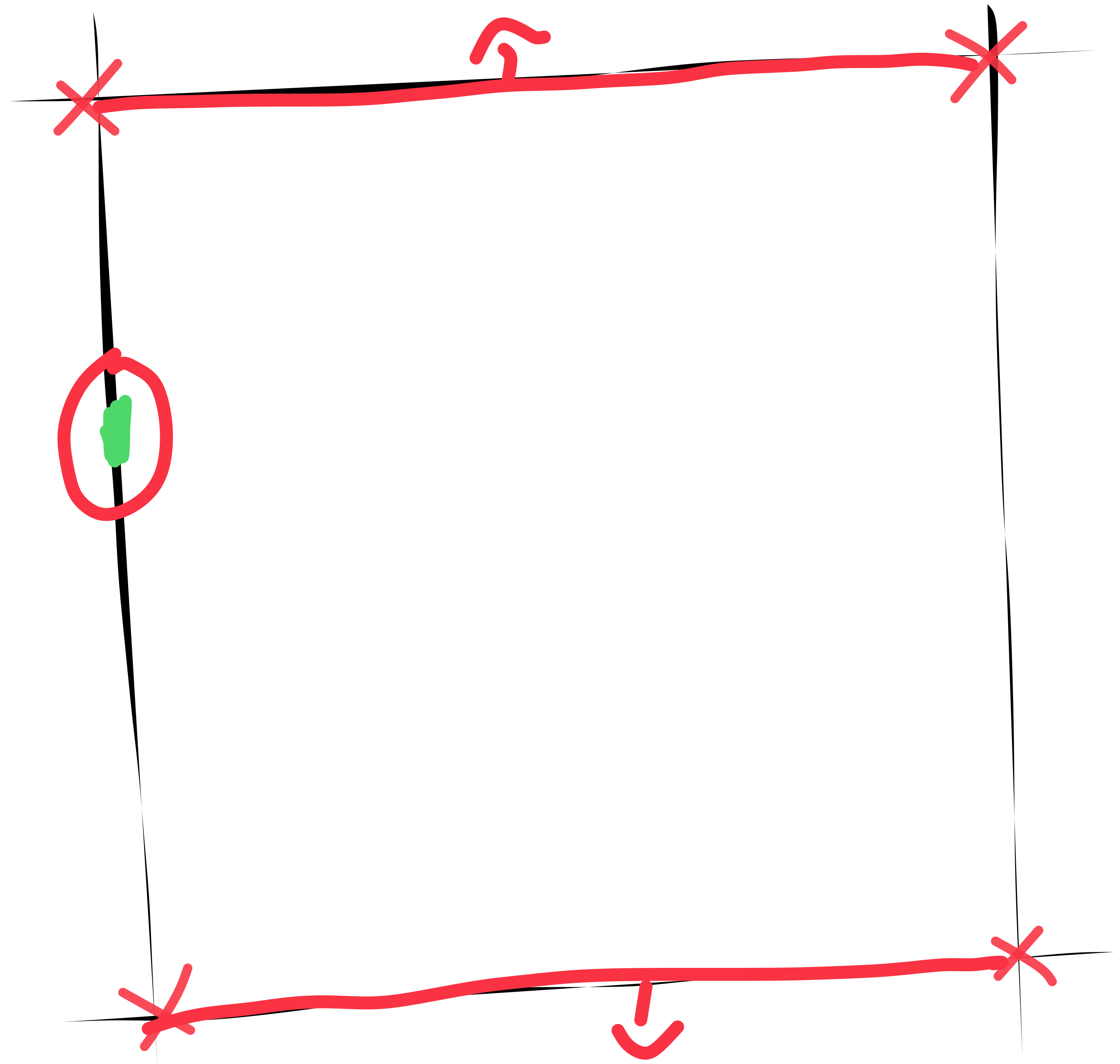
$$\begin{tikzcd}
 & P \arrow[dl, "x_1 \leq 0"'] \arrow[dr, "x_1 \geq 1"] & \\
P \cap \{x_1 \leq 0\} & & {}
\end{tikzcd}$$



DPLL as Polytopes

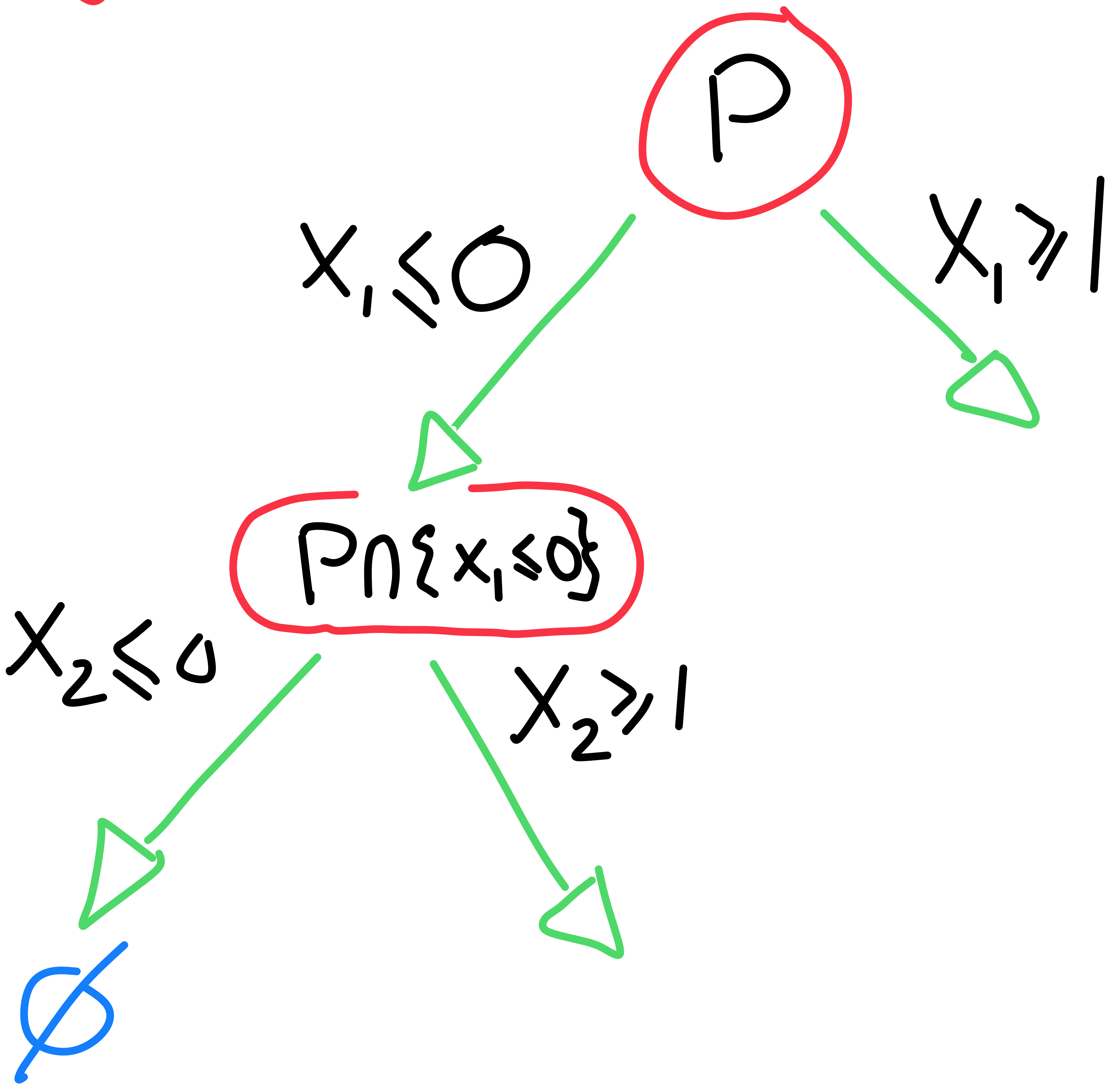
$$P = \{ x_1 + x_2 \geq 1,\ x_1 - x_2 \geq 0,\ x_2 - x_1 \geq 0,\ -x_1 - x_2 \geq -1,\ 0 \leq x_i \leq 1 \}$$

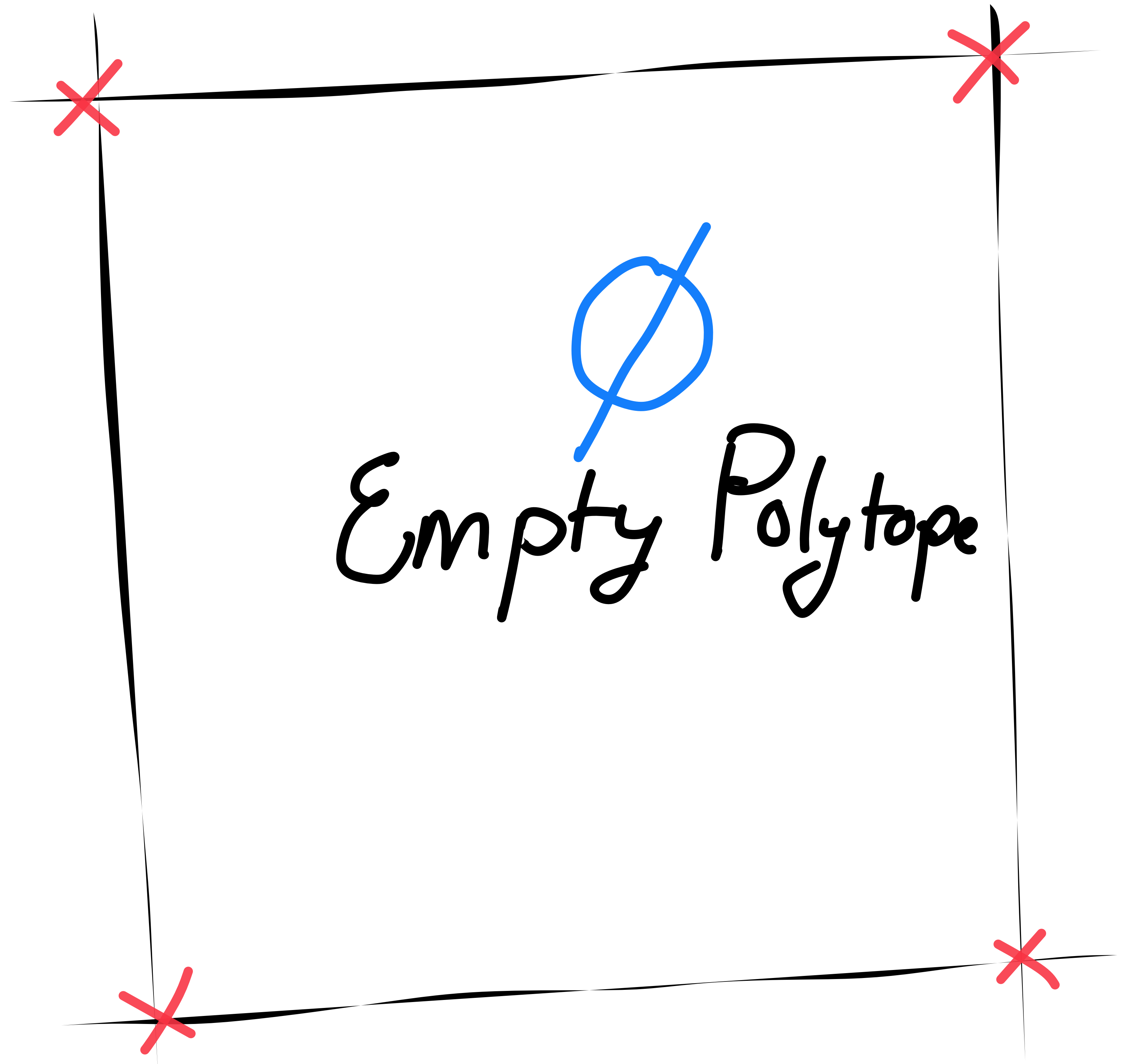




DPLL as Polytopes

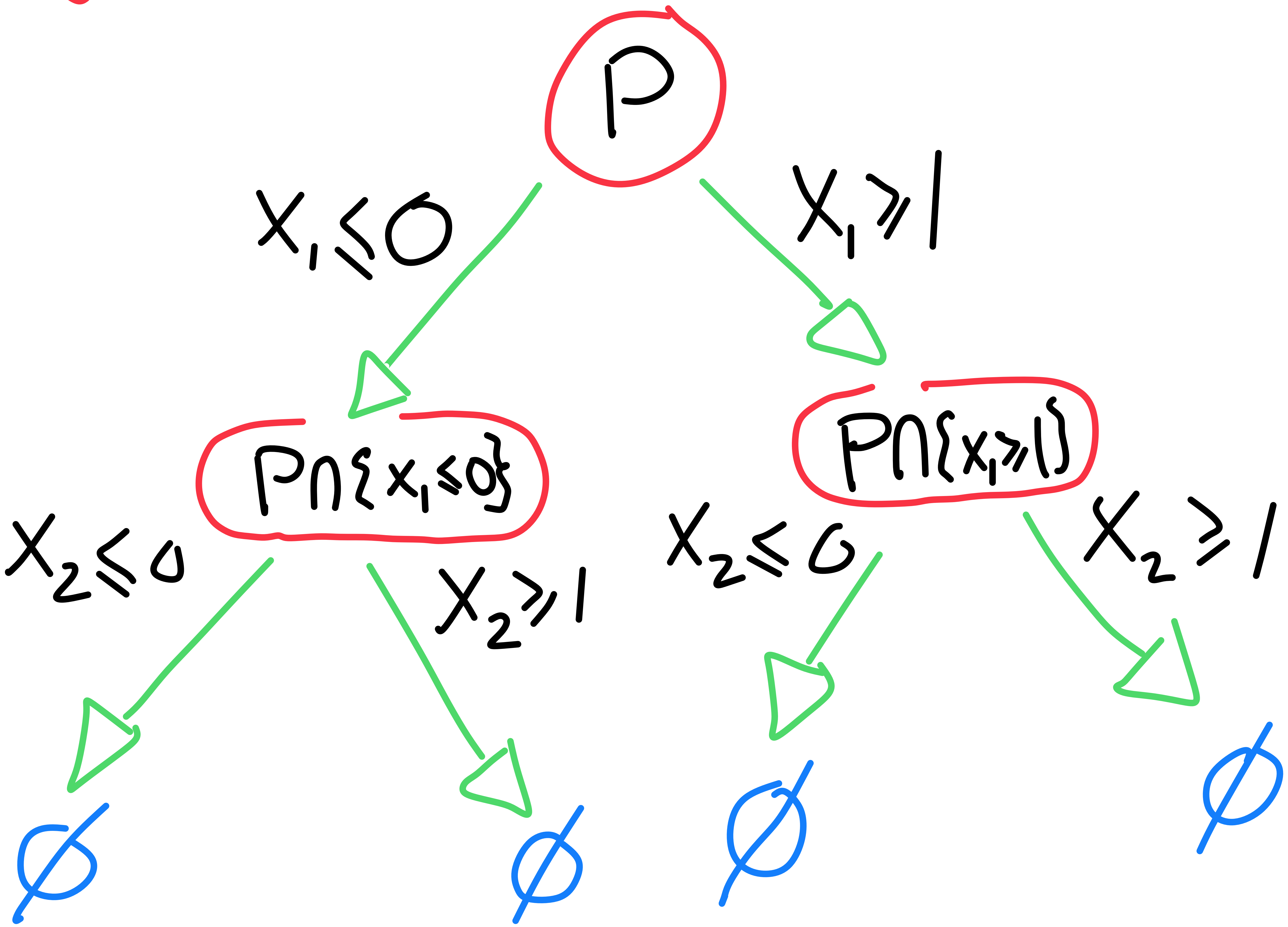
$$P = \{x_1 + x_2 \geq 1,\ x_1 - x_2 \geq 0,\ x_2 - x_1 \geq 0,\ -x_1 - x_2 \geq -1,\ 0 \leq x_i \leq 1\}$$

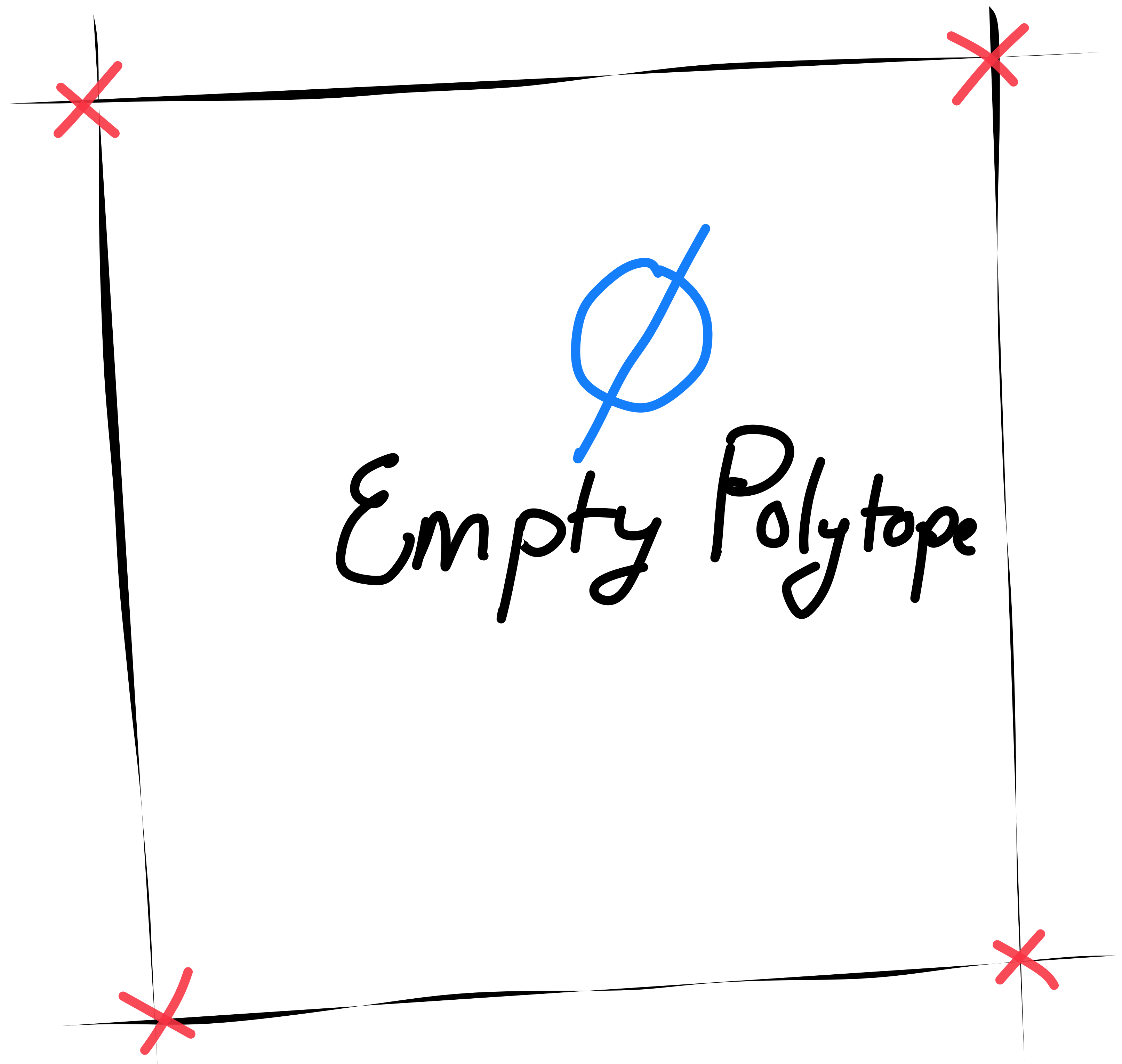




DPLL as Polytopes

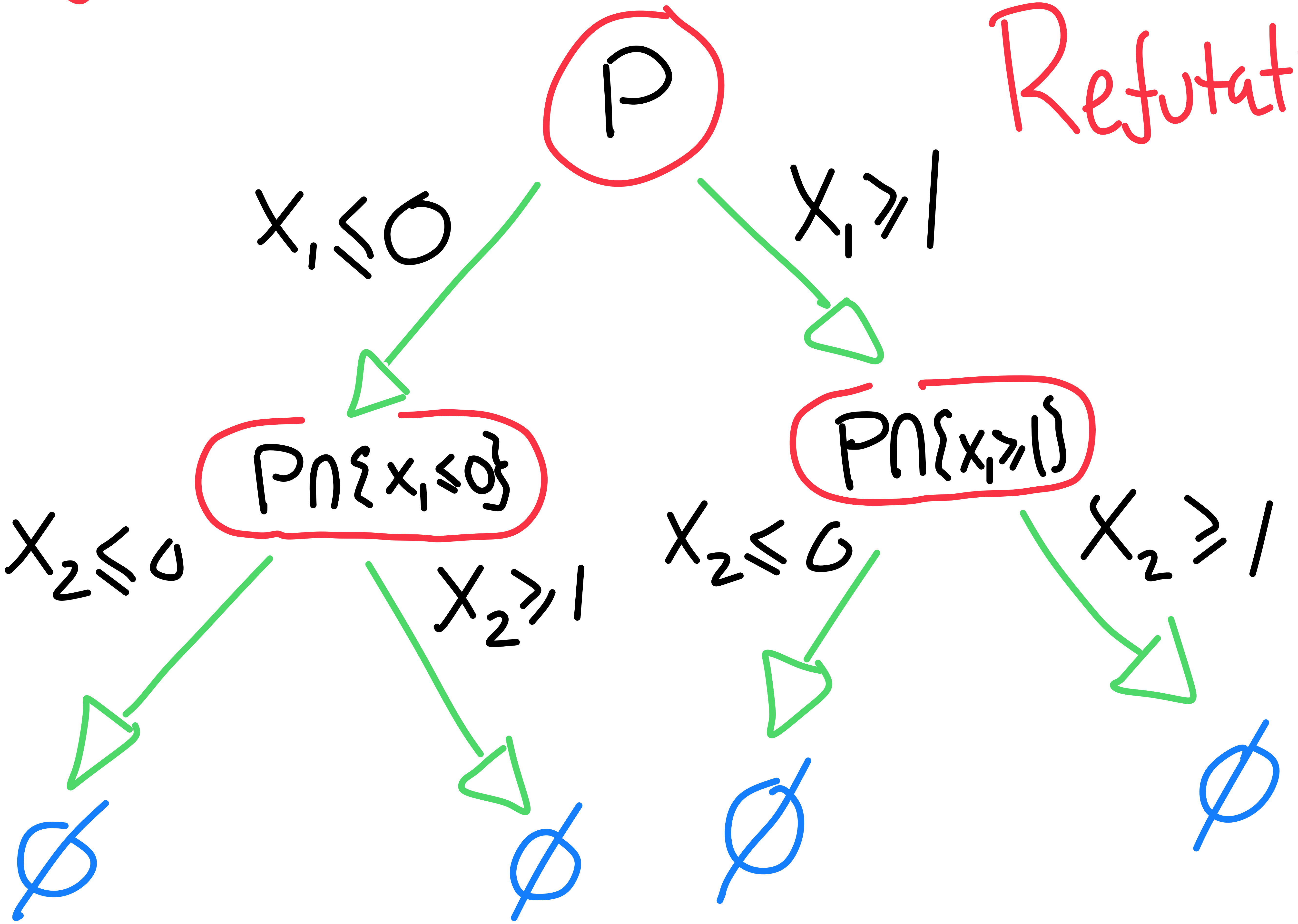
$$P = \{ x_1 + x_2 \geq 1,\ x_1 - x_2 \geq 0,\ x_2 - x_1 \geq 0,\ -x_1 - x_2 \geq -1,\ 0 \leq x_i \leq 1 \}$$



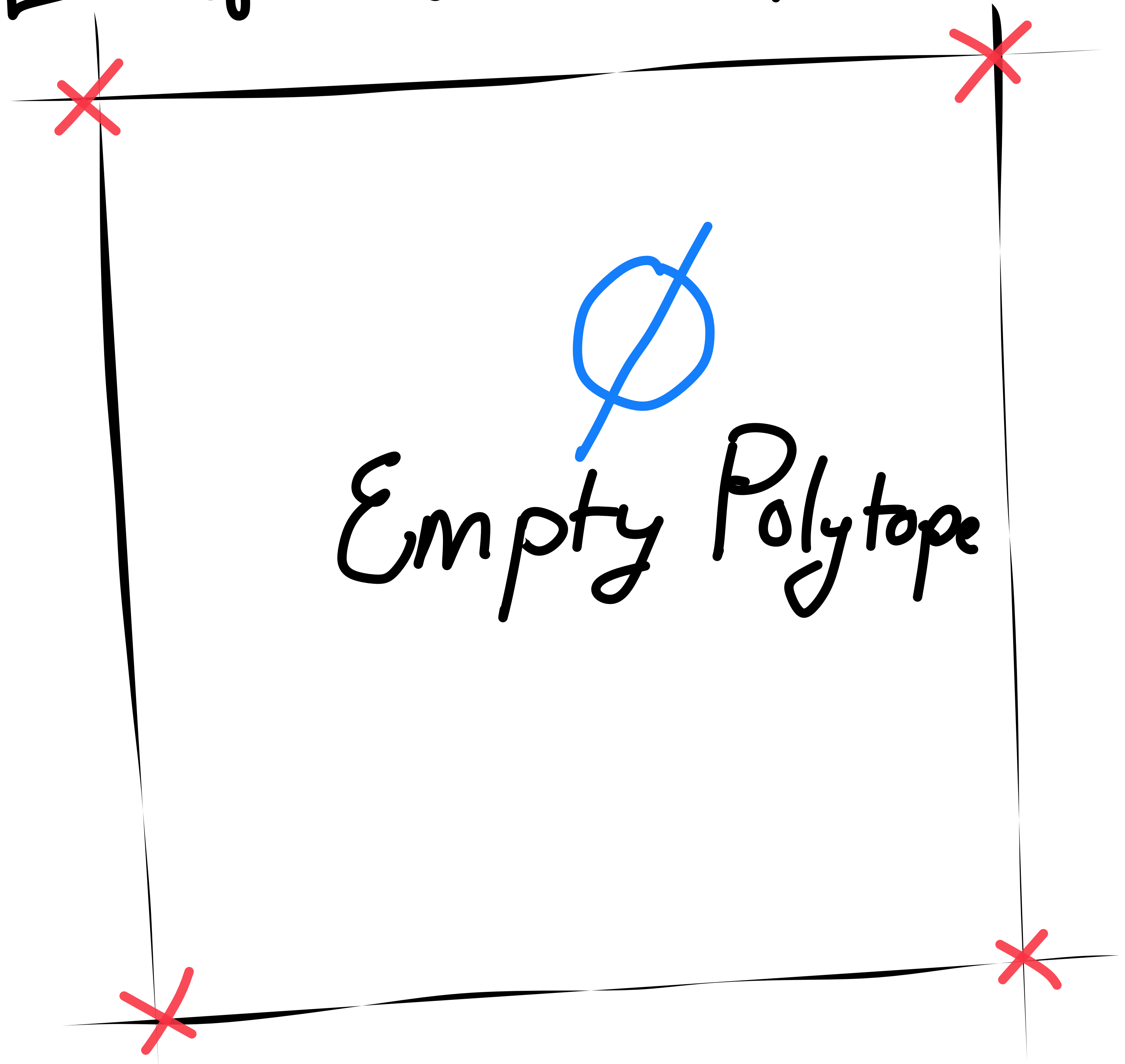


DPLL as Polytopes

$$P = \{ x_1 + x_2 \geq 1,\ x_1 - x_2 \geq 0,\ x_2 - x_1 \geq 0,\ -x_1 - x_2 \geq -1,\ 0 \leq x_i \leq 1 \}$$



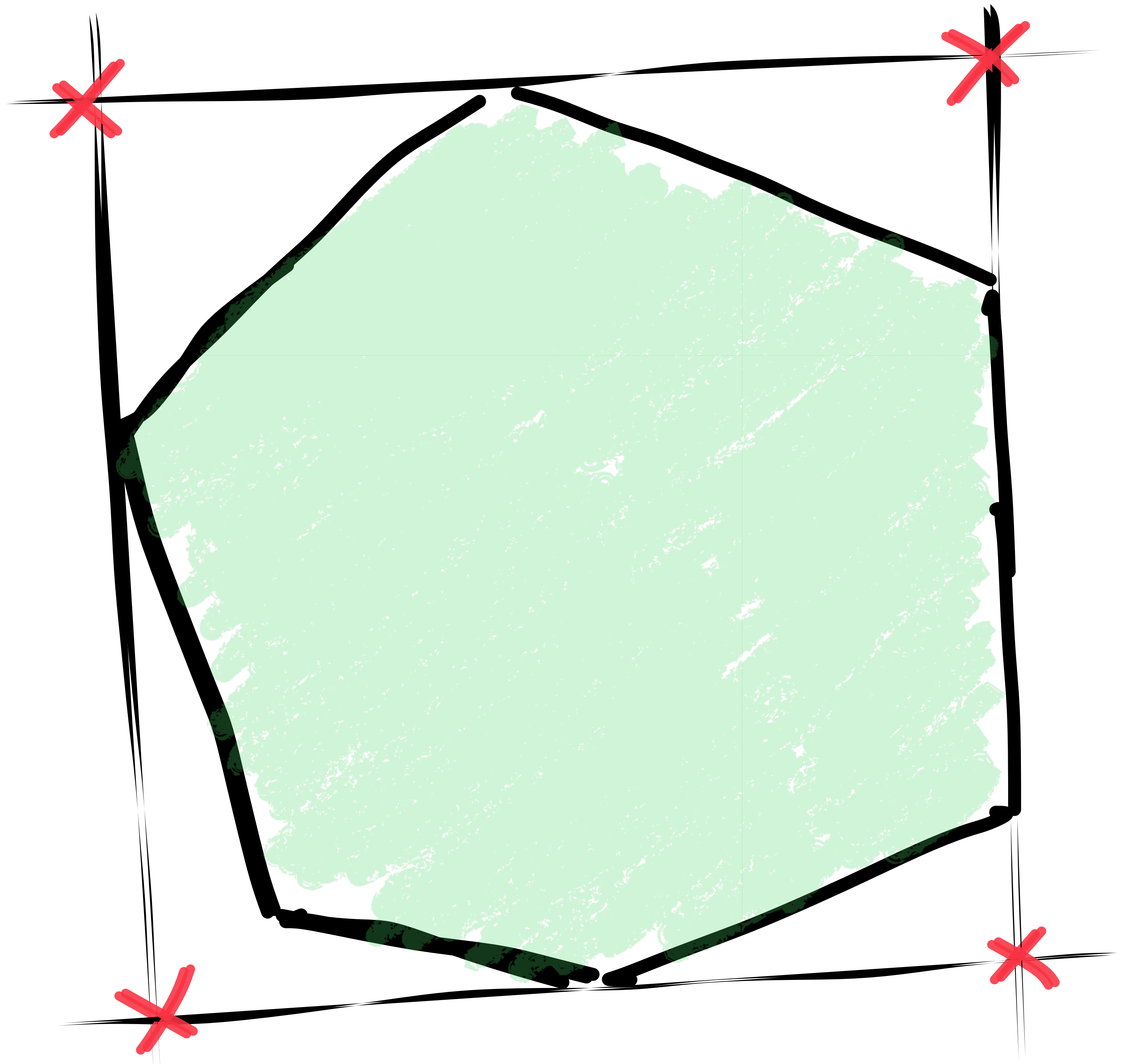
Refutation! $\emptyset$ Derived at every Leaf of the tree



Stabbing Planes

$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.

P

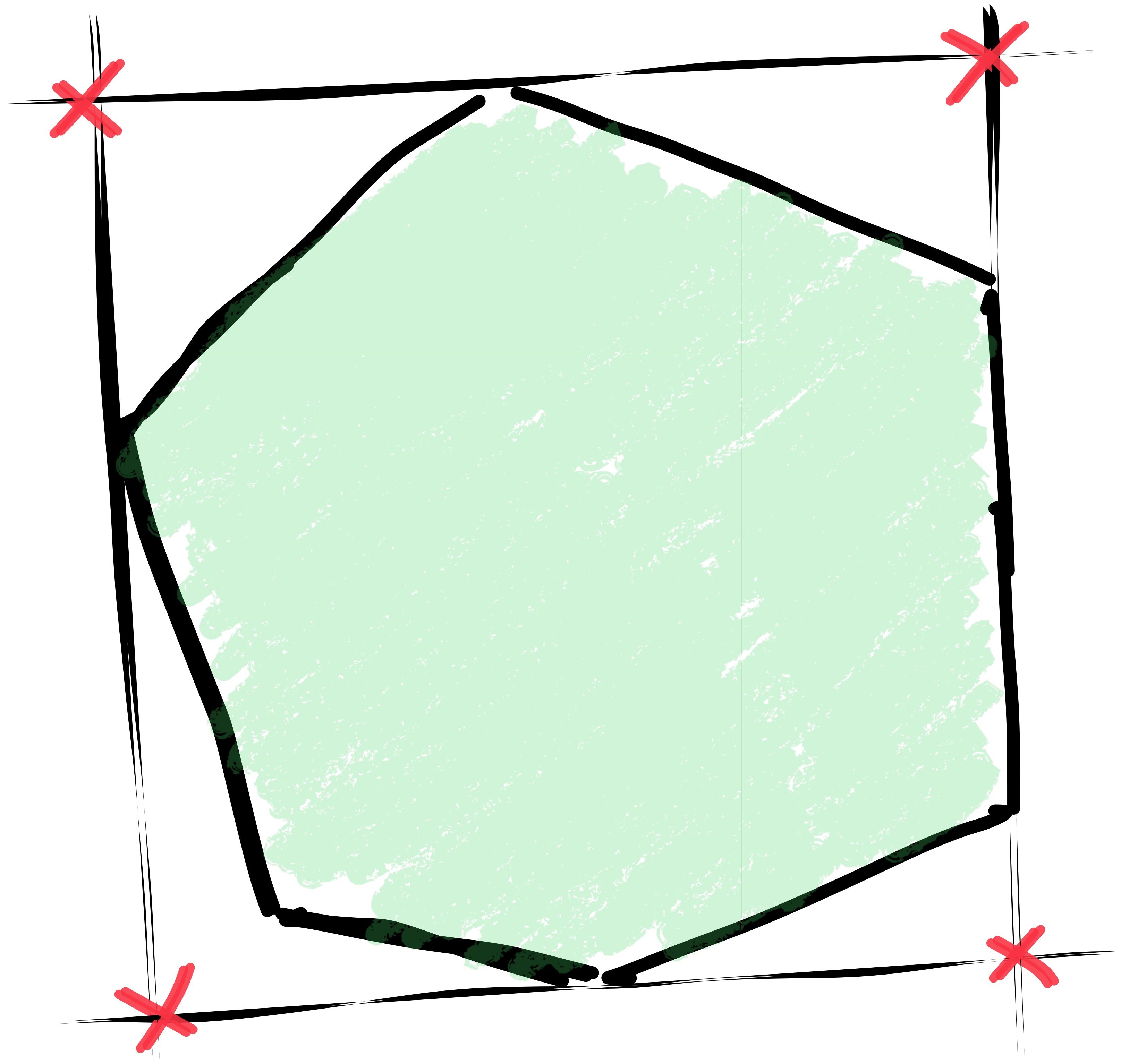


Stabbing Planes

"query arbitrary linear inequalities"

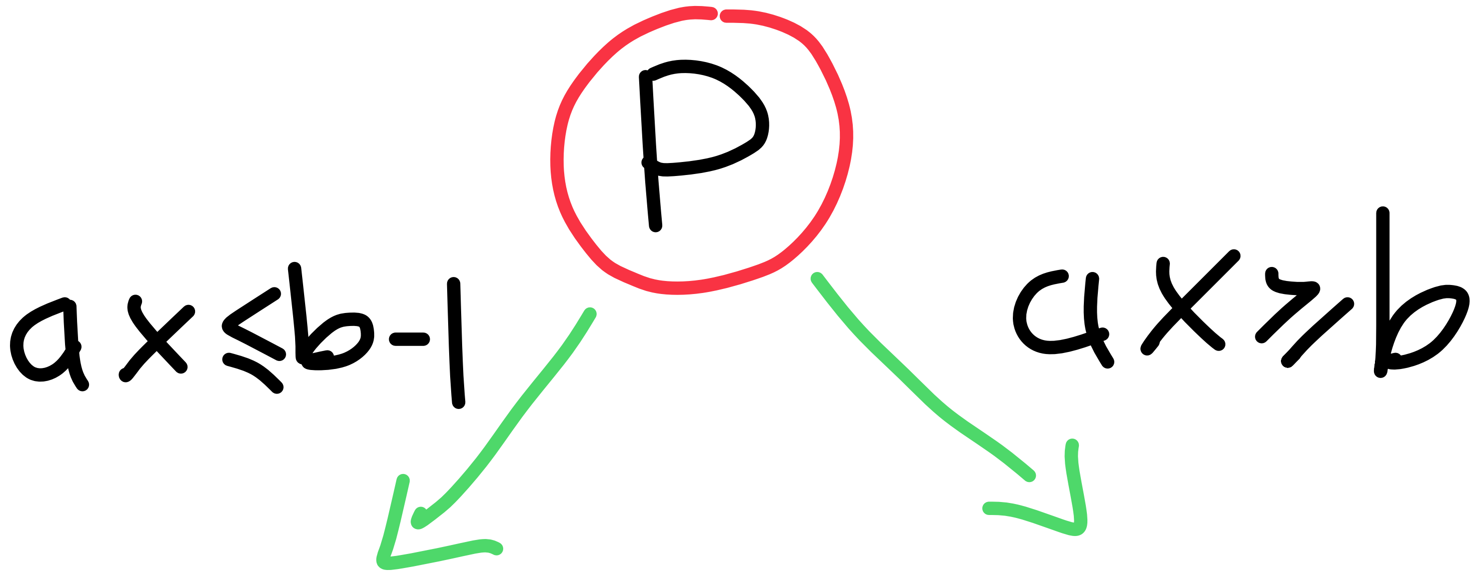
$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.

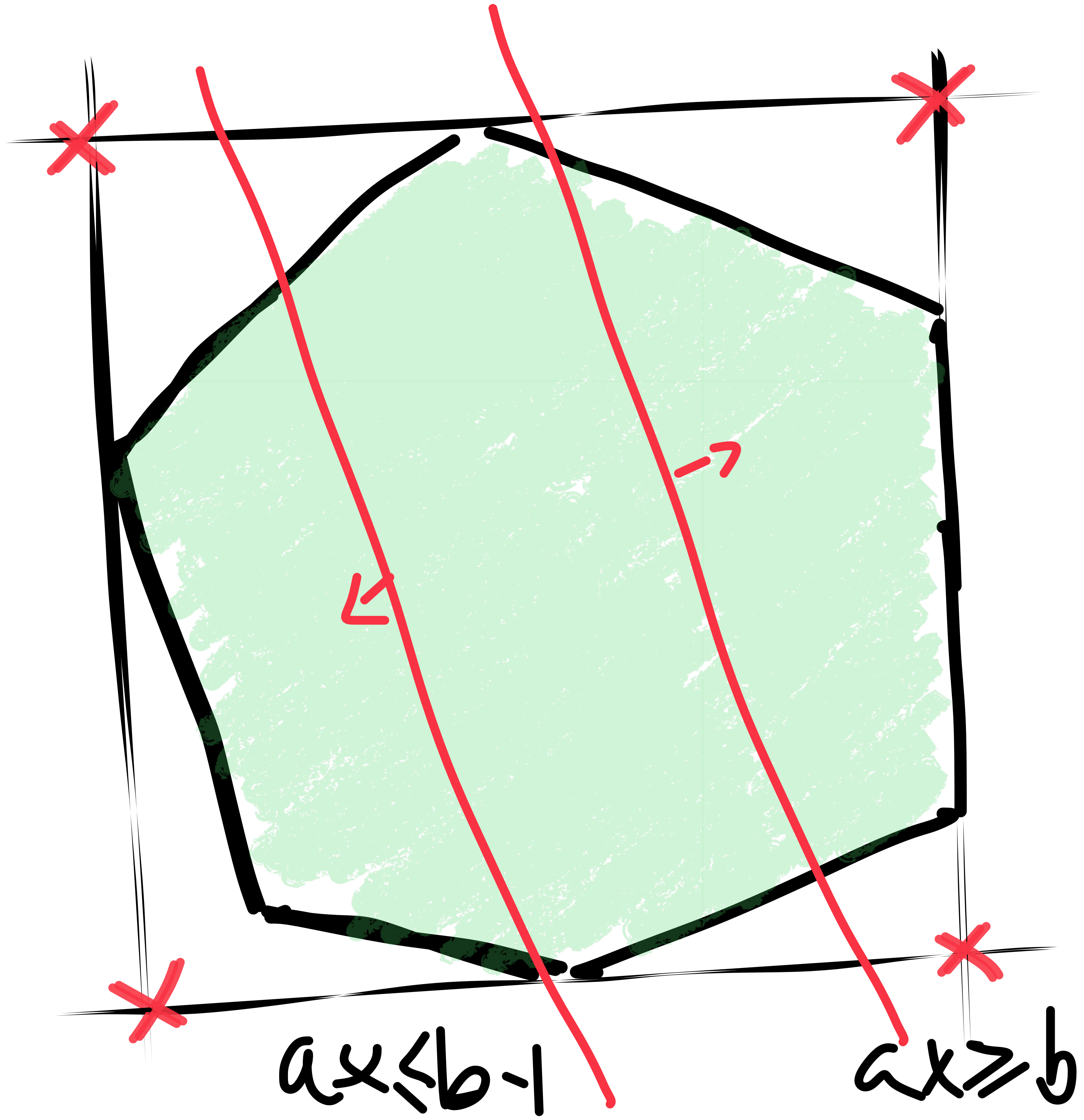
P



Stabbing Planes

$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.

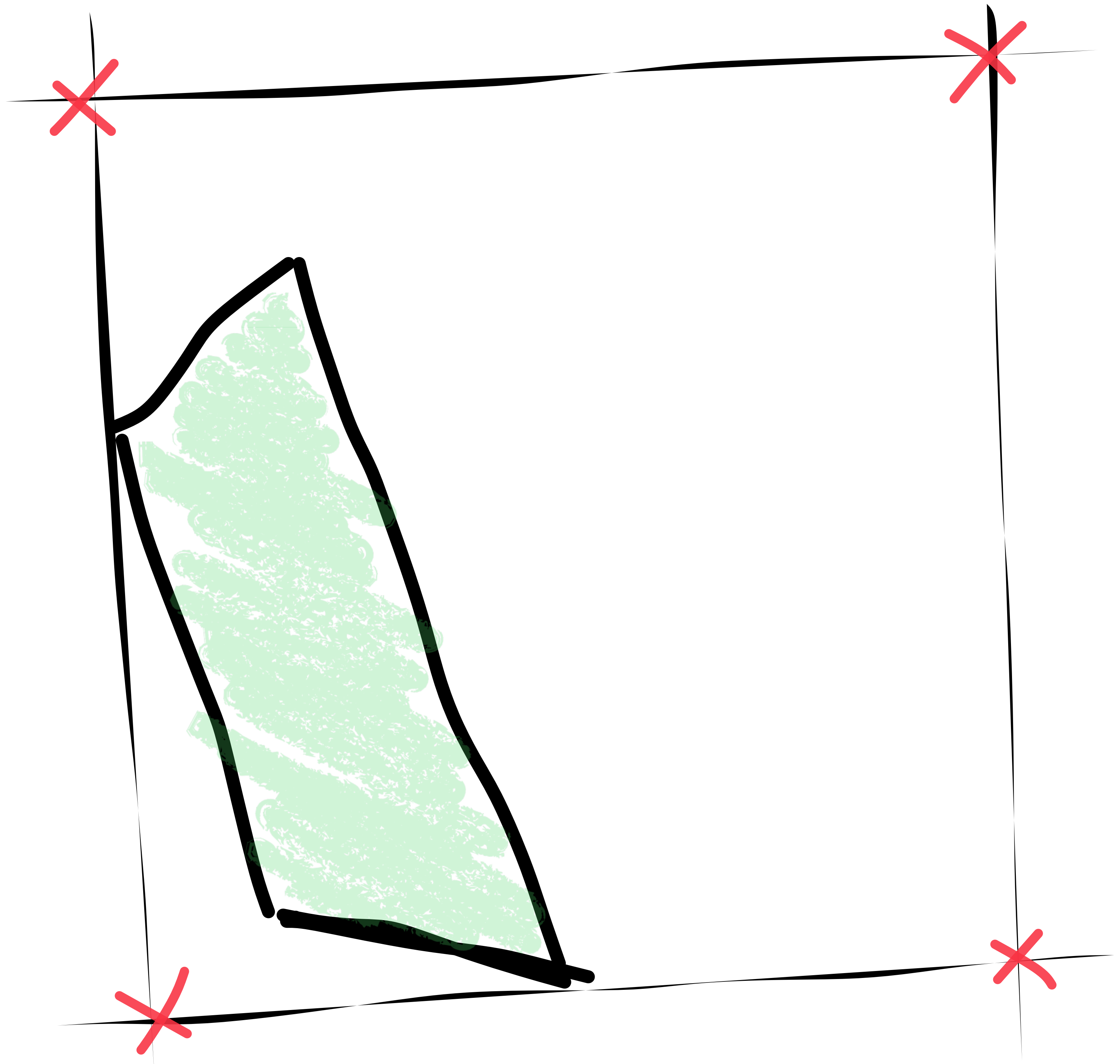




Stabbing Planes

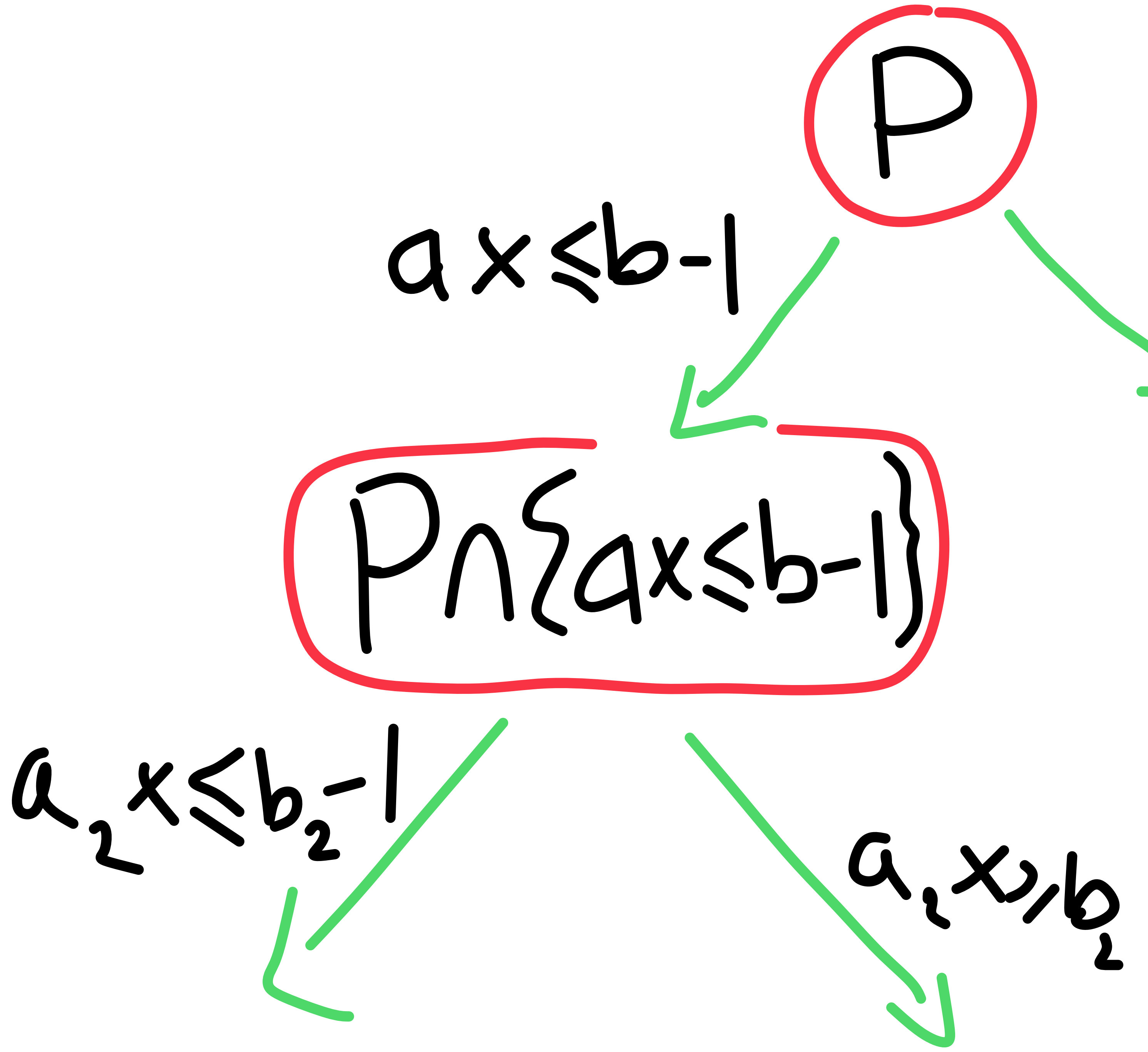
$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.

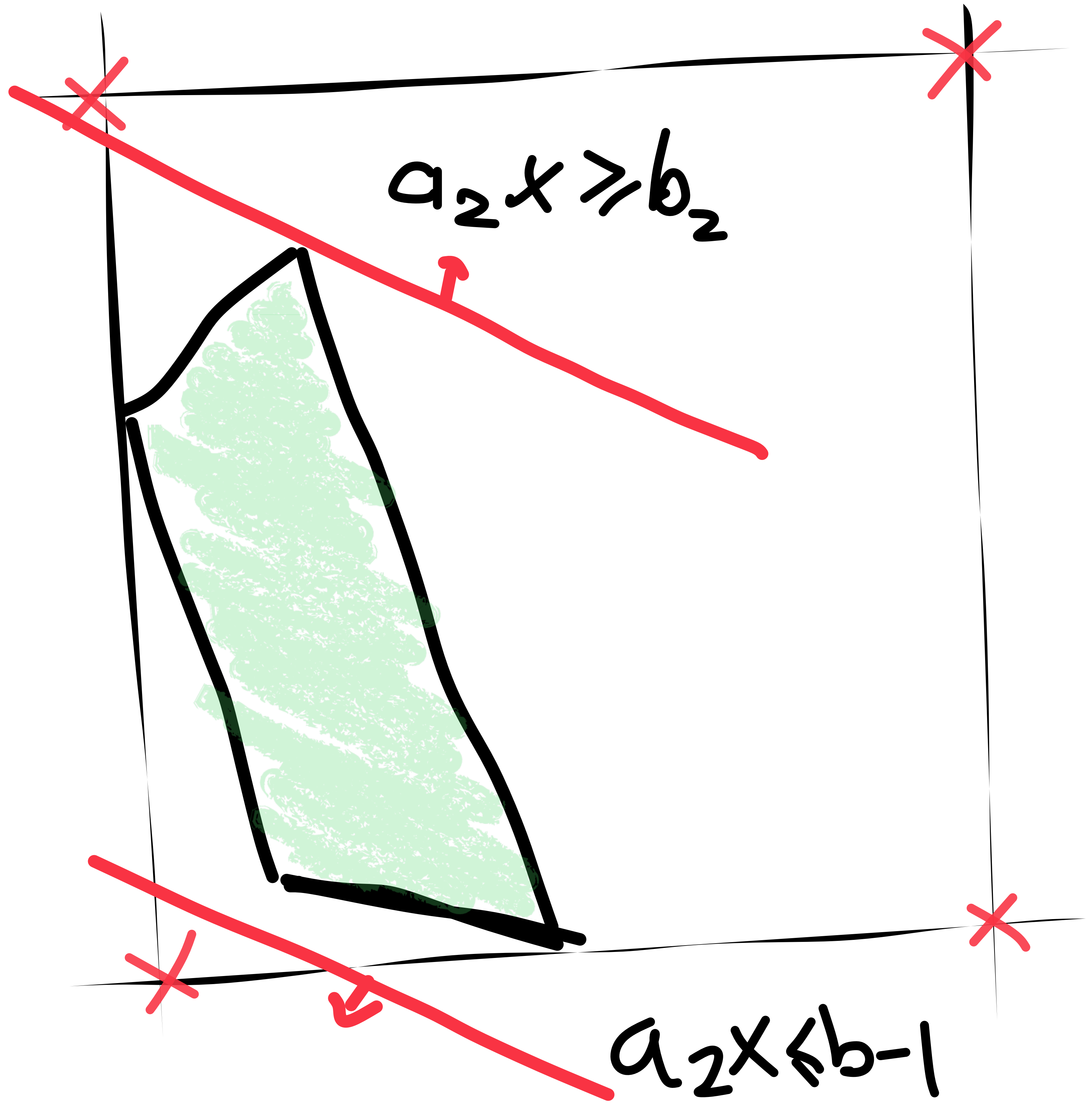
$$\begin{tikzcd} & P \arrow[dl, "ax \leq b-1"'] \arrow[dr, "ax \geq b"] & \\ \boxed{P \cap \{ax \leq b-1\}} & & {} \end{tikzcd}$$



Stabbing Planes

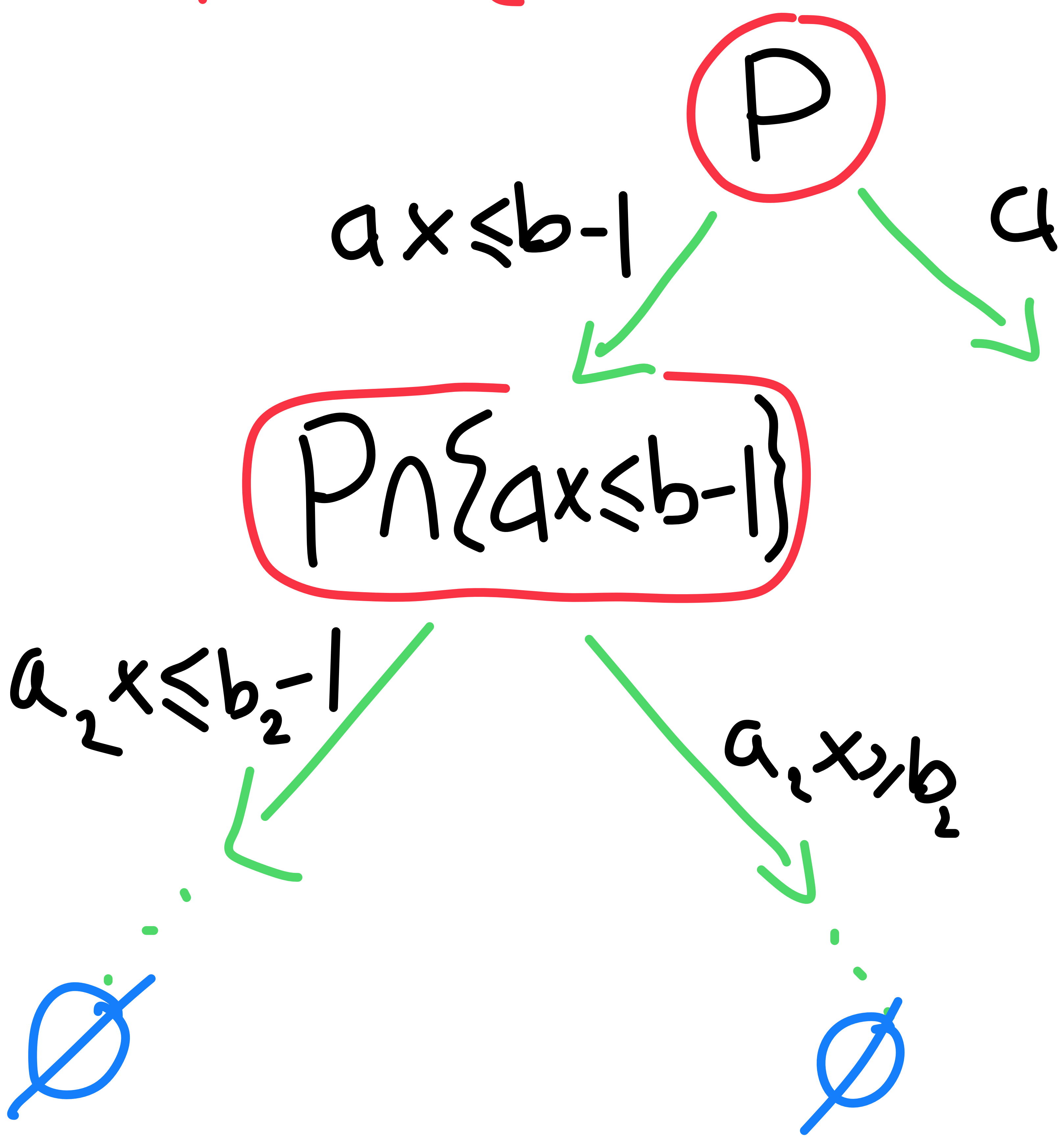
$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.

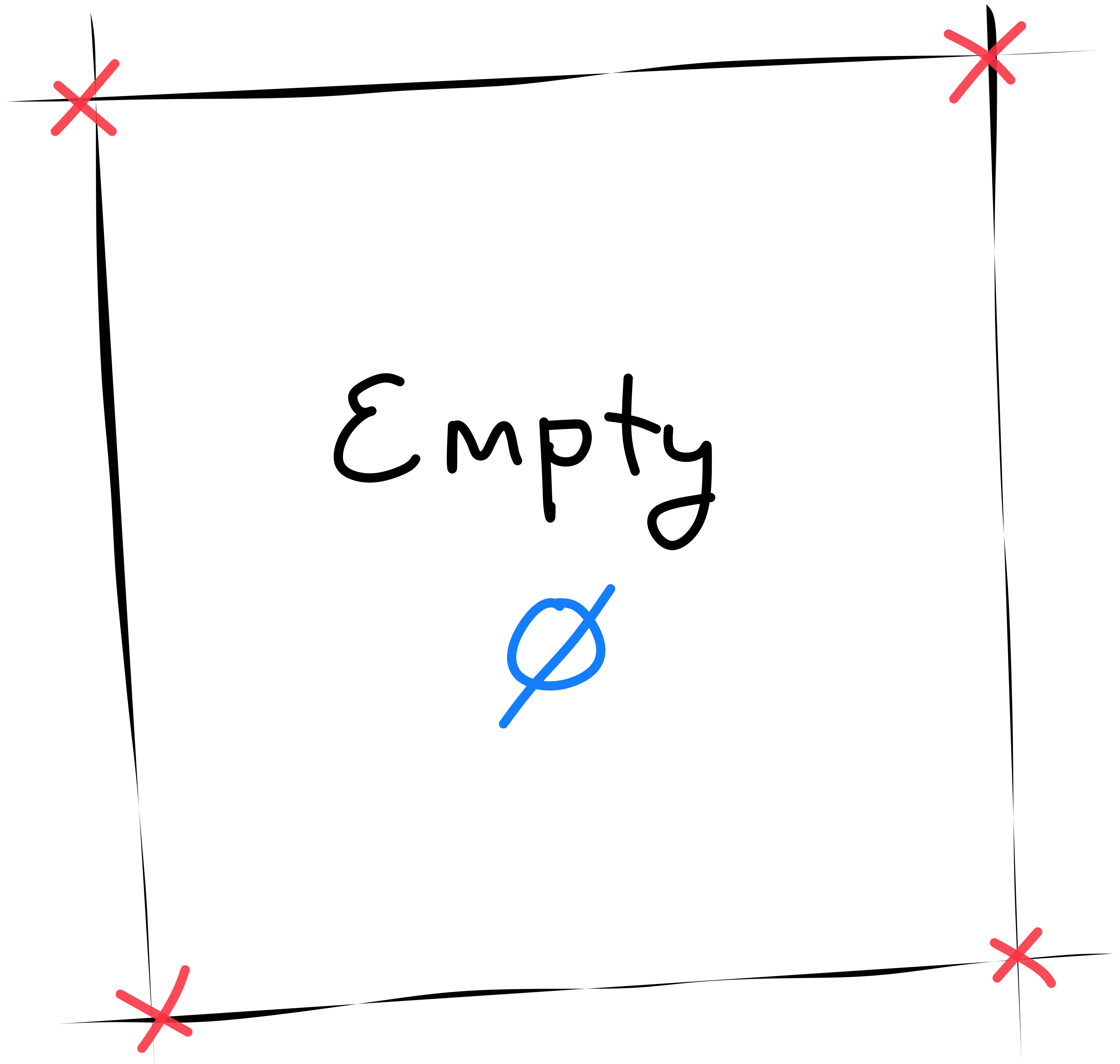




Stabbing Planes

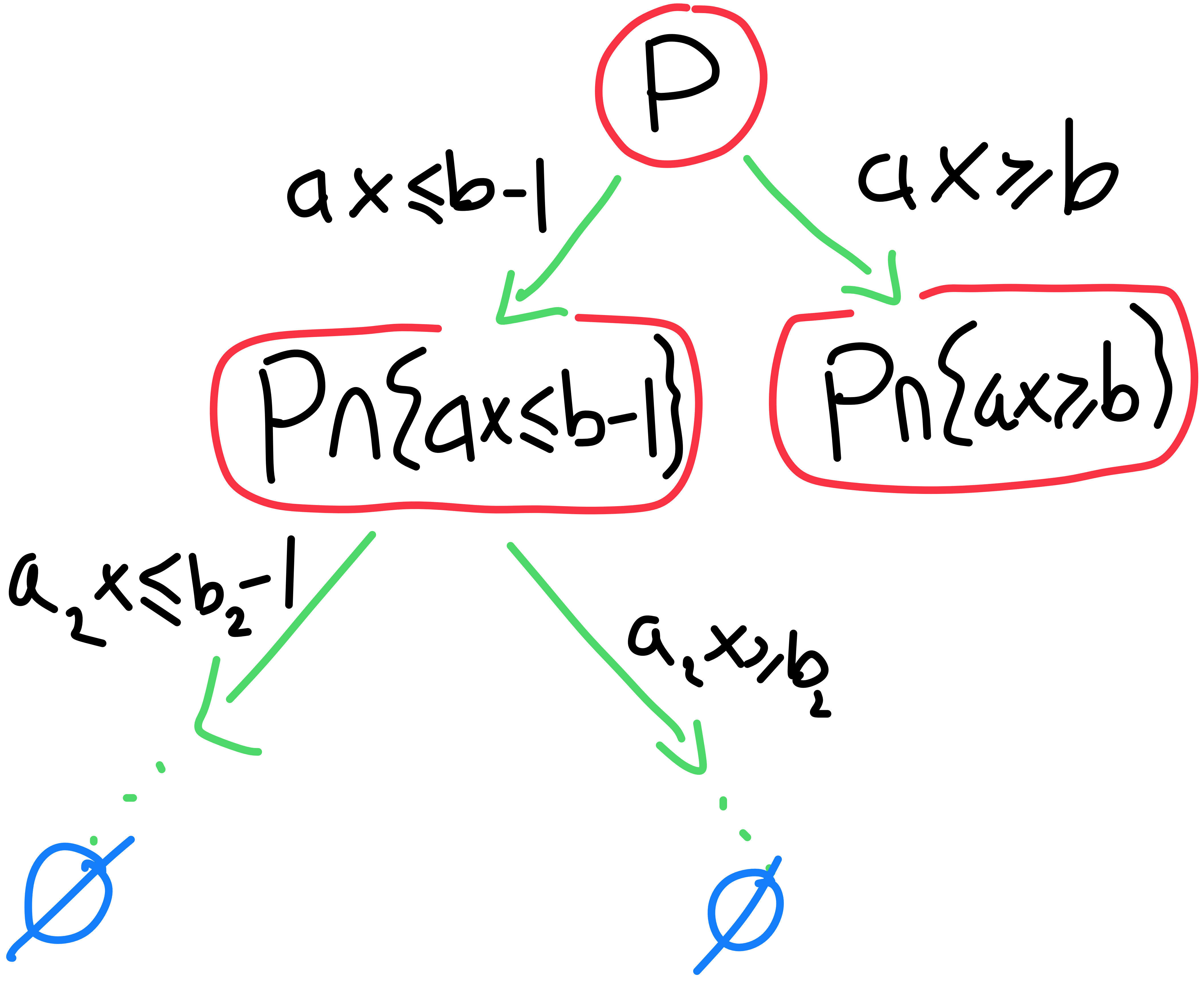
$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.

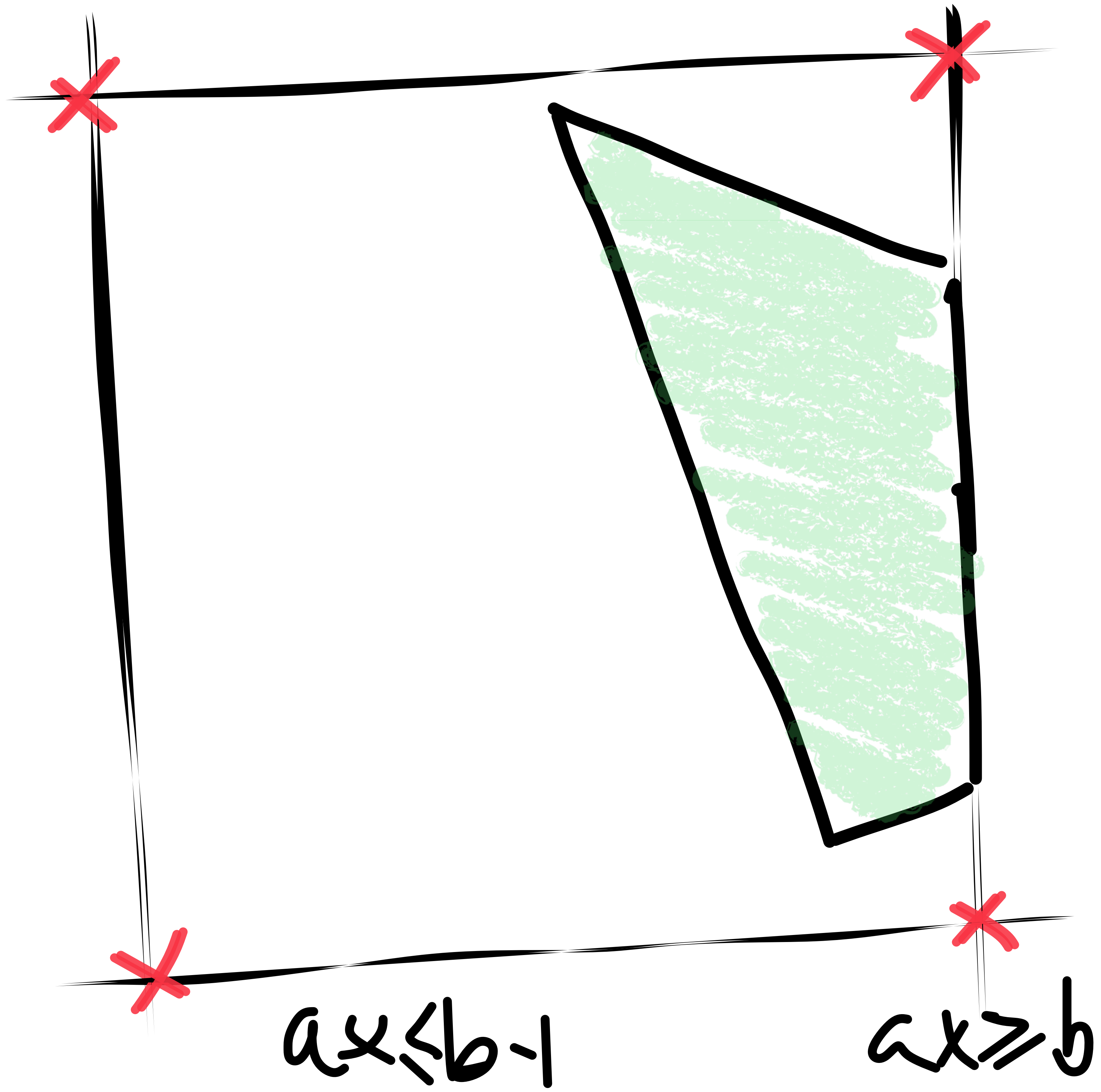




Stabbing Planes

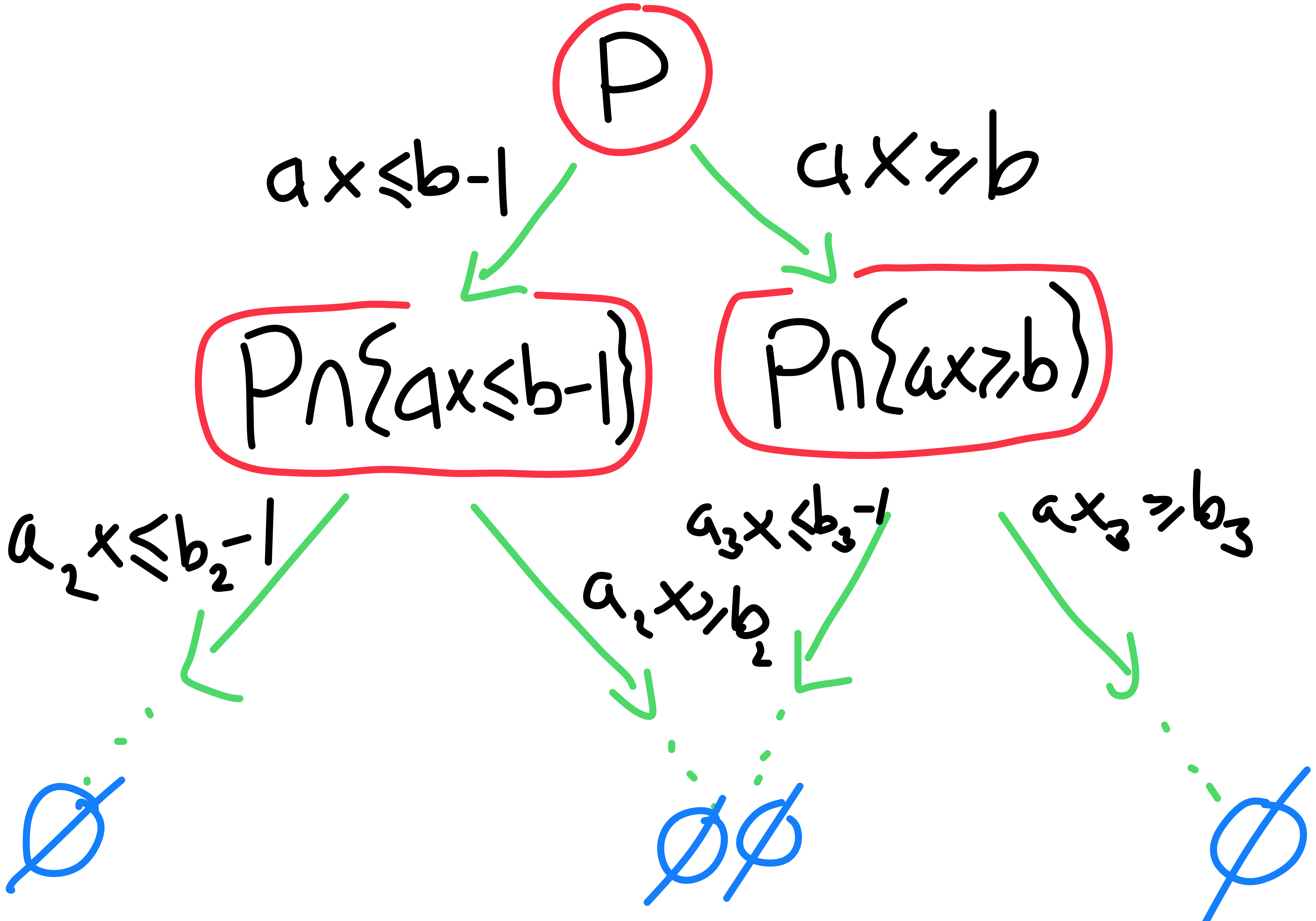
$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.





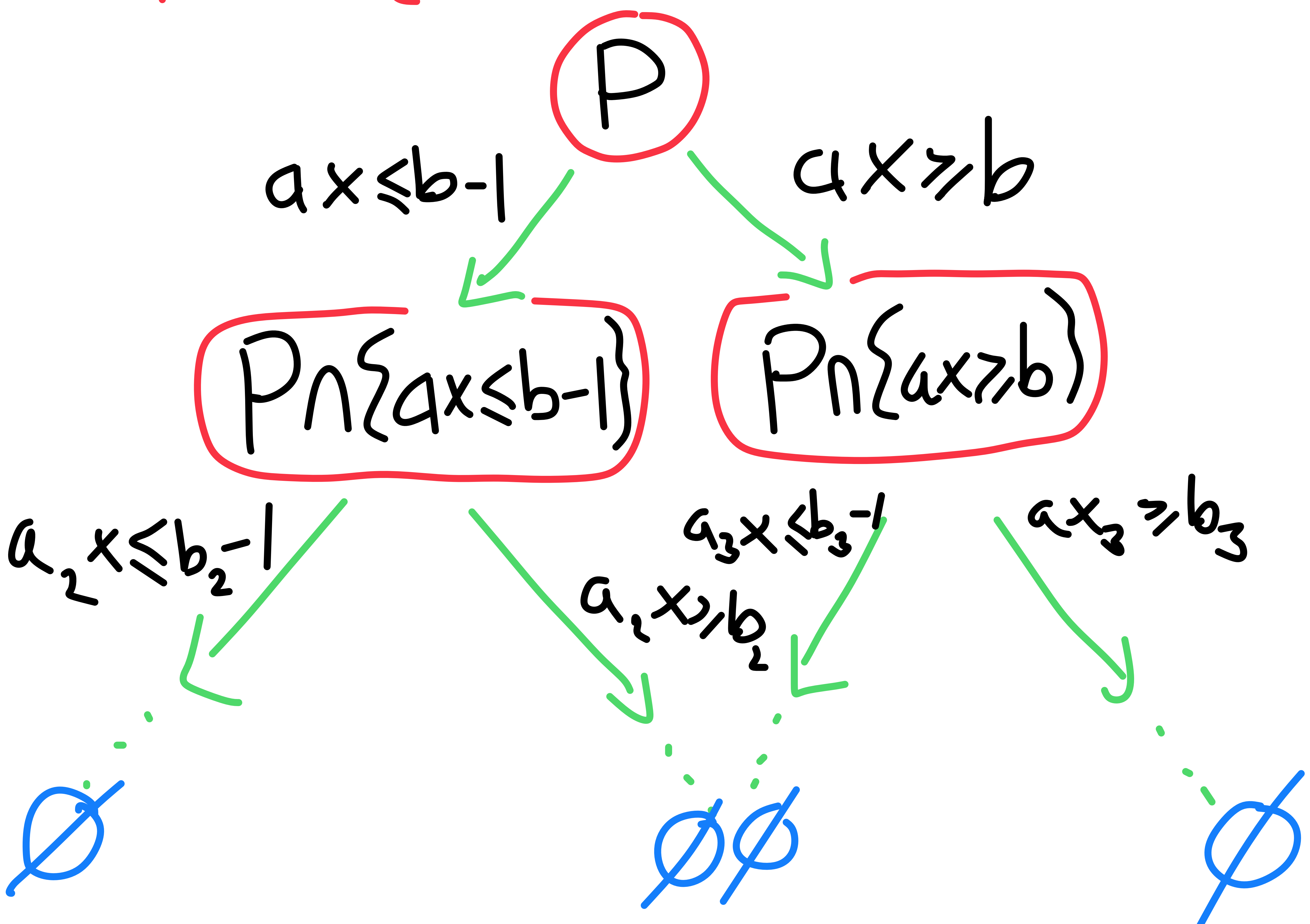
Stabbing Planes

$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.



Stabbing Planes

$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.

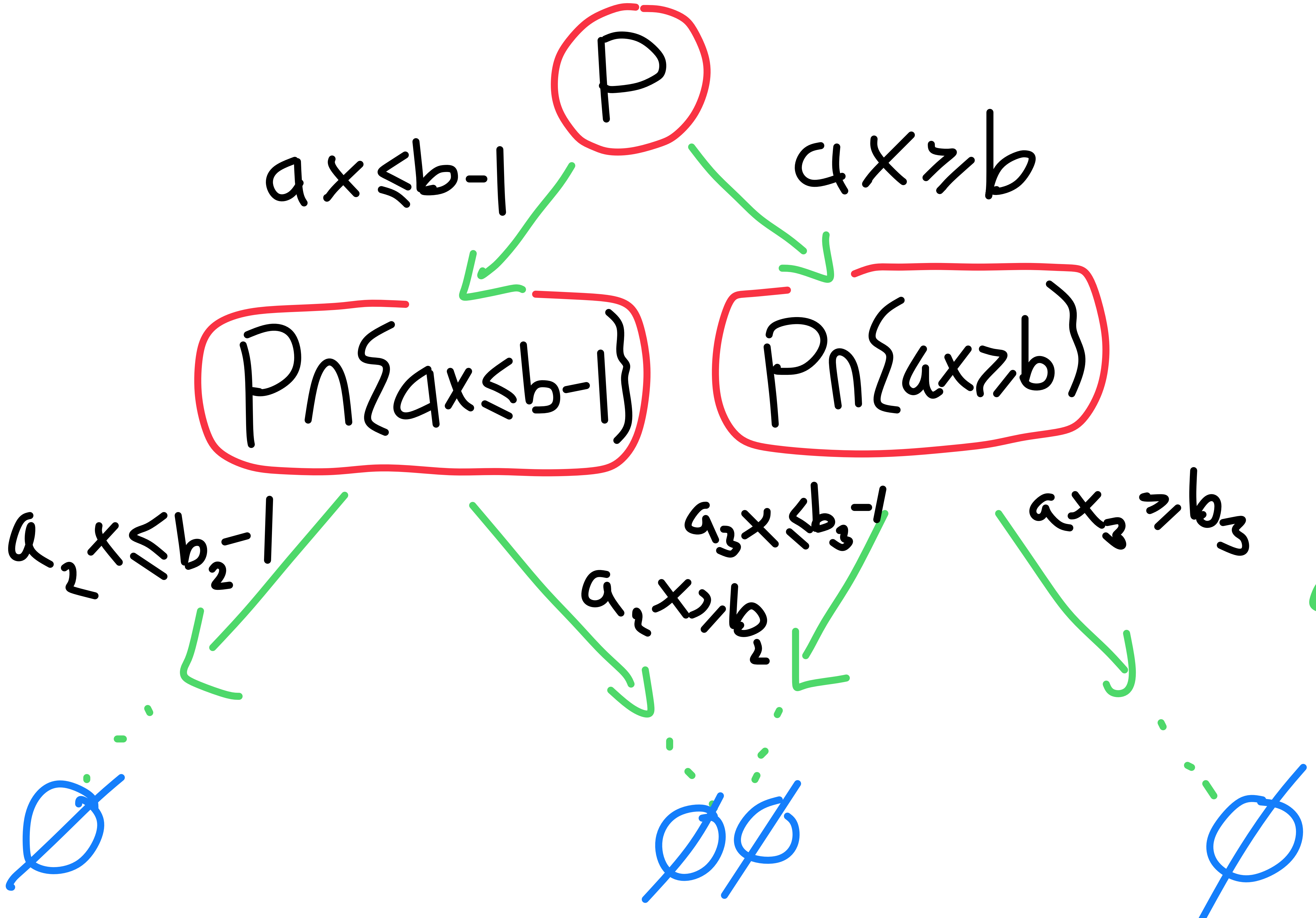


Refutation

Empty polytope $\emptyset$ deduced at every leaf proves $P \cap \mathbb{Z}^n = \emptyset$

Stabbing Planes

$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.



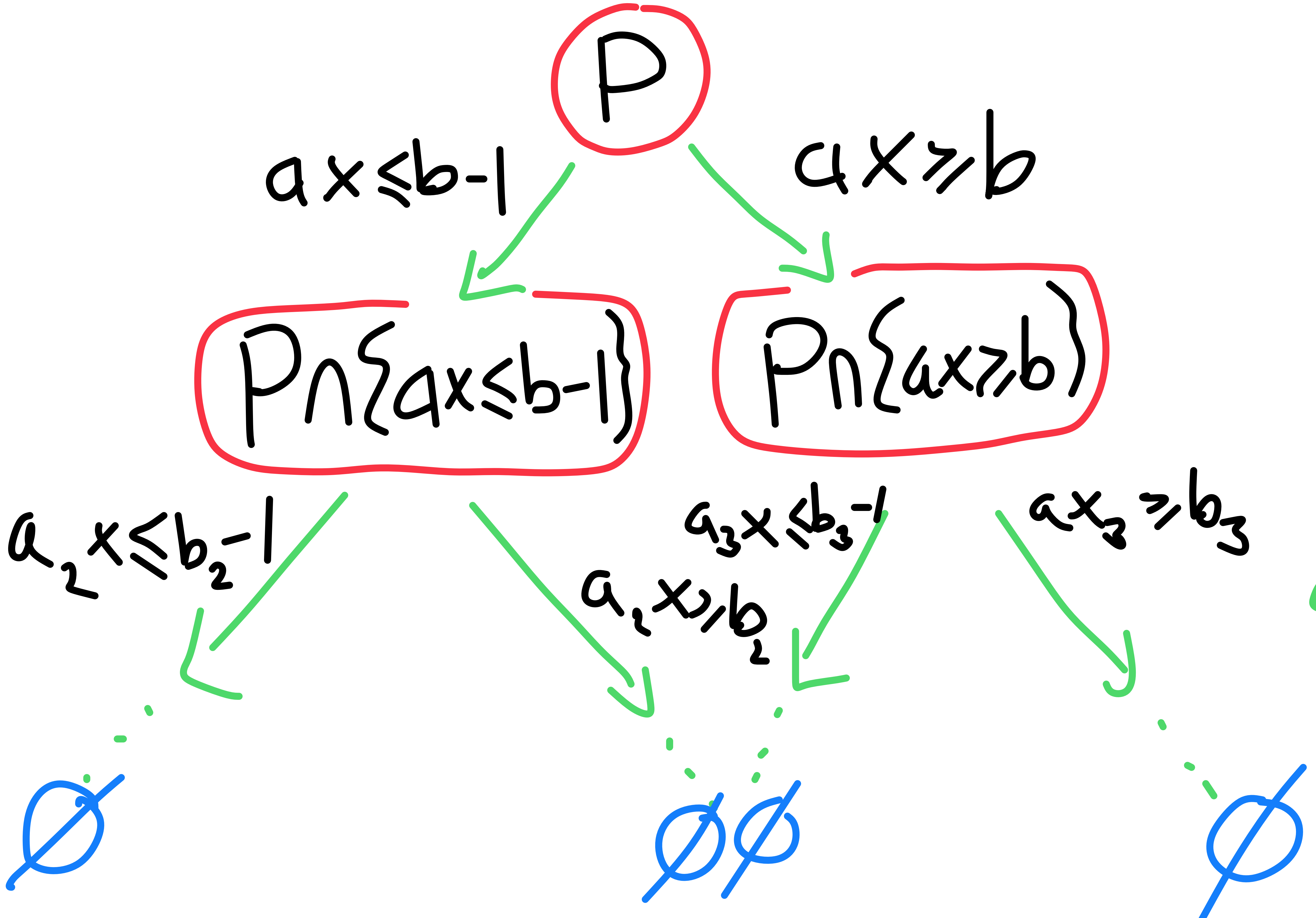
Refutation

Empty polytope $\emptyset$ deduced at every leaf proves $P \cap \mathbb{Z}^n = \emptyset$

Size: # of queries

Stabbing Planes

$P = \{Ax \geq b\}$ s.t. $P \cap \mathbb{Z}^n = \emptyset$.



Refutation

Empty polytope $\emptyset$ deduced at every leaf proves $P \cap \mathbb{Z}^n = \emptyset$

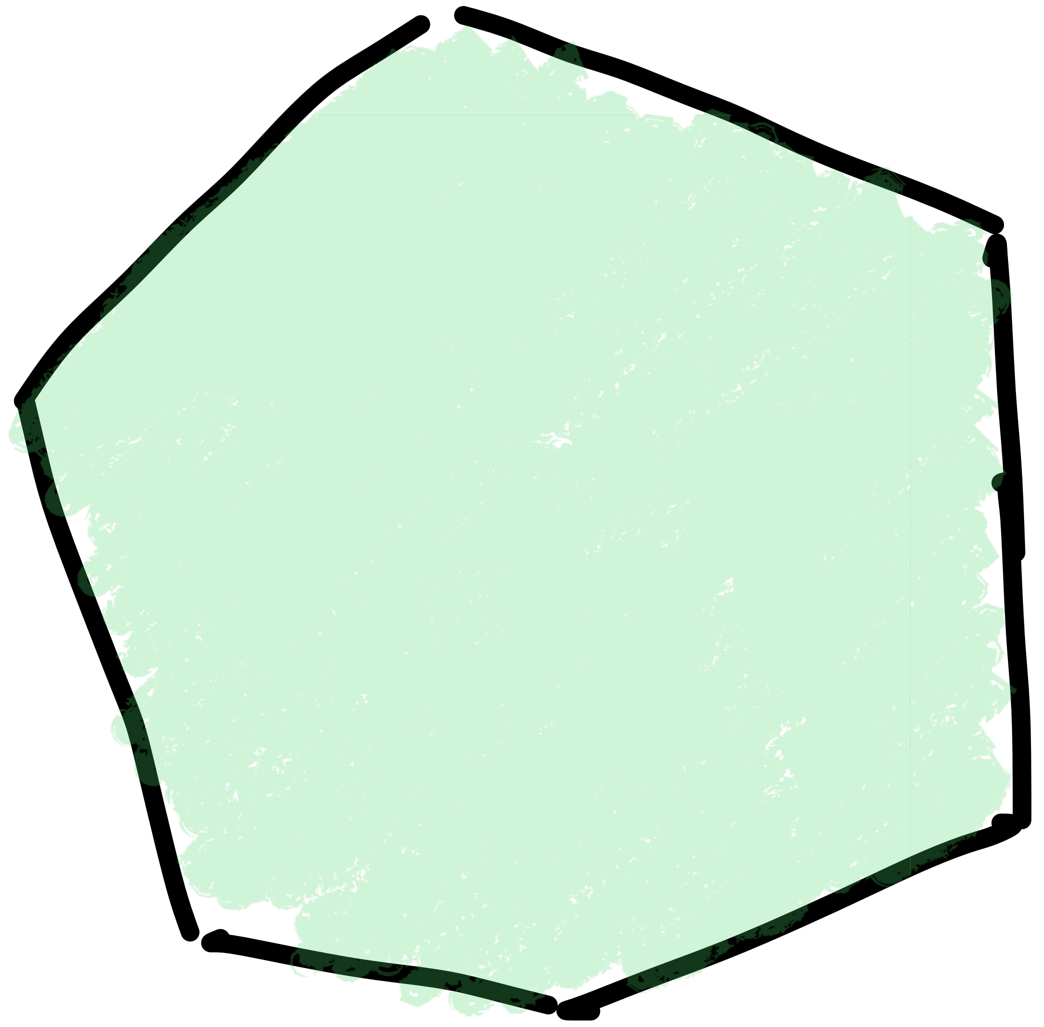
Size: # of queries $\approx$ bit-size [DT20]

Stabbing Planes vs Cutting Planes

- Branch-and-cut allows Cutting planes deductions

Stabbing Planes vs Cutting Planes

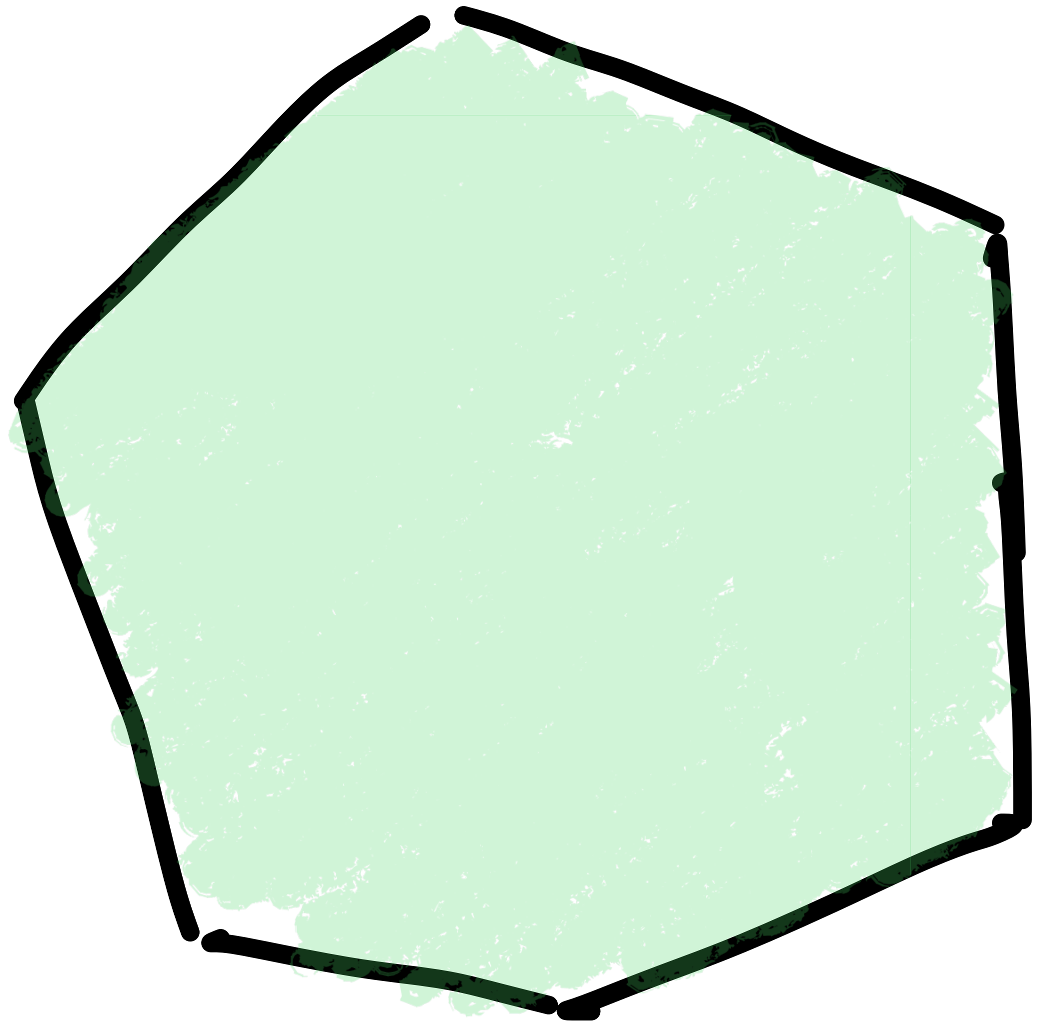
- Branch-and-cut allows Cutting planes deductions
  - → Superfluous as SP can simulate CP



Stabbing Planes vs Cutting Planes

- Branch-and-cut allows Cutting planes deductions
  - → Superfluous as SP can simulate CP

Clm: pathlike SP = CP

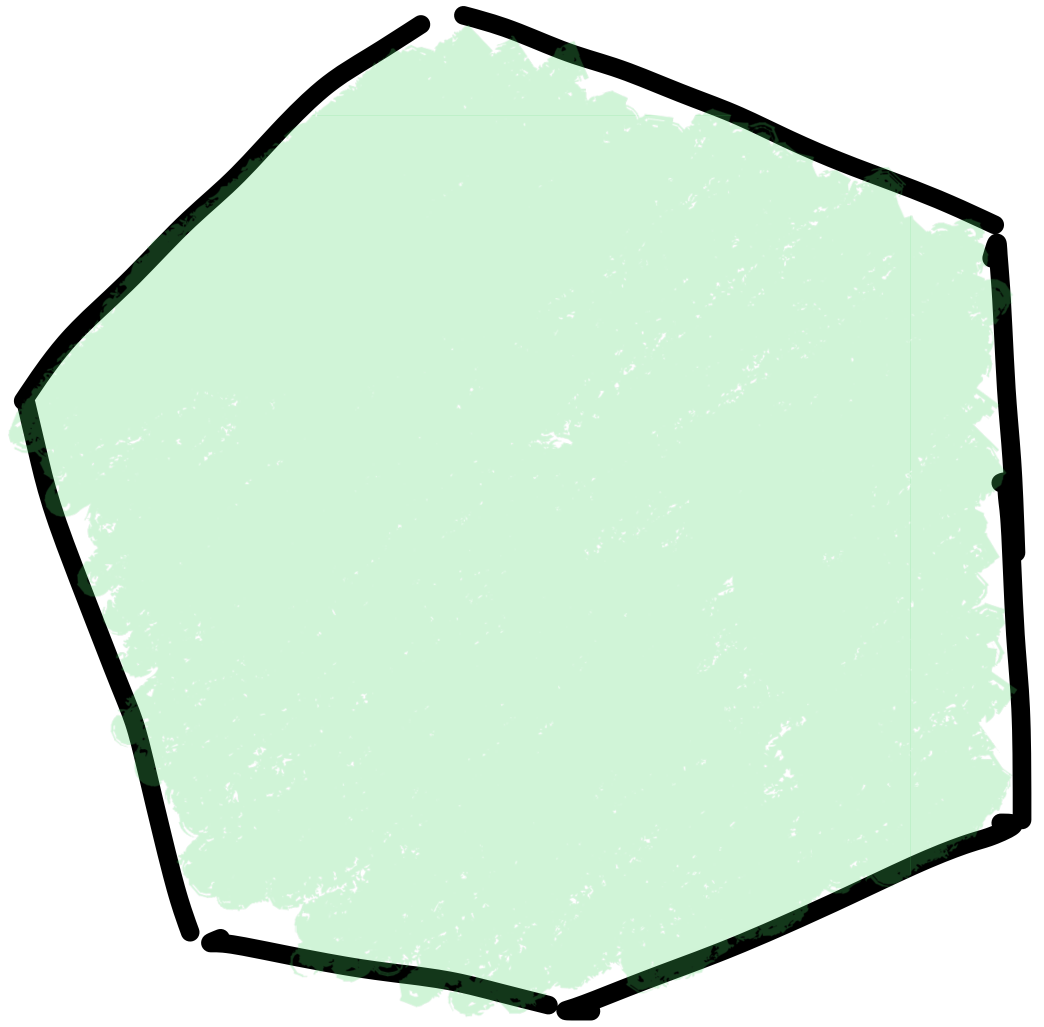


Stabbing Planes vs Cutting Planes

- Branch-and-cut allows Cutting planes deductions
  - → Superfluous as SP can simulate CP

Pathlike SP: A SP query $(ax \leq b-1, \ ax \geq b)$ at P is pathlike if $P \cap \{ax \leq b-1\} = \emptyset$ or $P \cap \{ax \geq b\}$

Clm: pathlike SP = CP

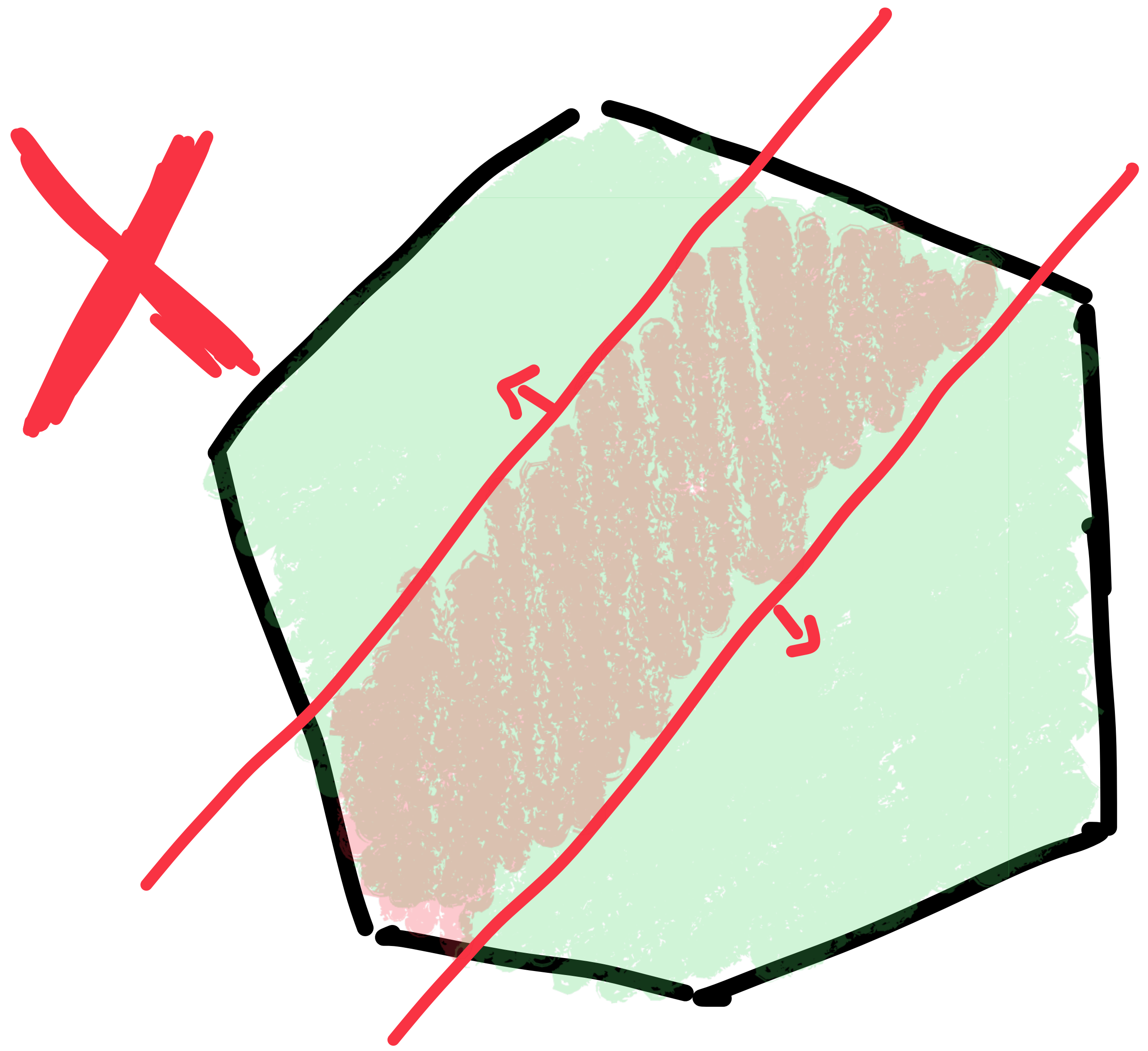


Stabbing Planes vs Cutting Planes

- Branch-and-cut allows Cutting planes deductions
  - → Superfluous as SP can simulate CP

Pathlike SP: A SP query $(ax \leq b-1,\ ax \geq b)$ at P is pathlike if $P \cap \{ax \leq b-1\} = \emptyset$ or $P \cap \{ax \geq b\}$

Clm: pathlike SP = CP

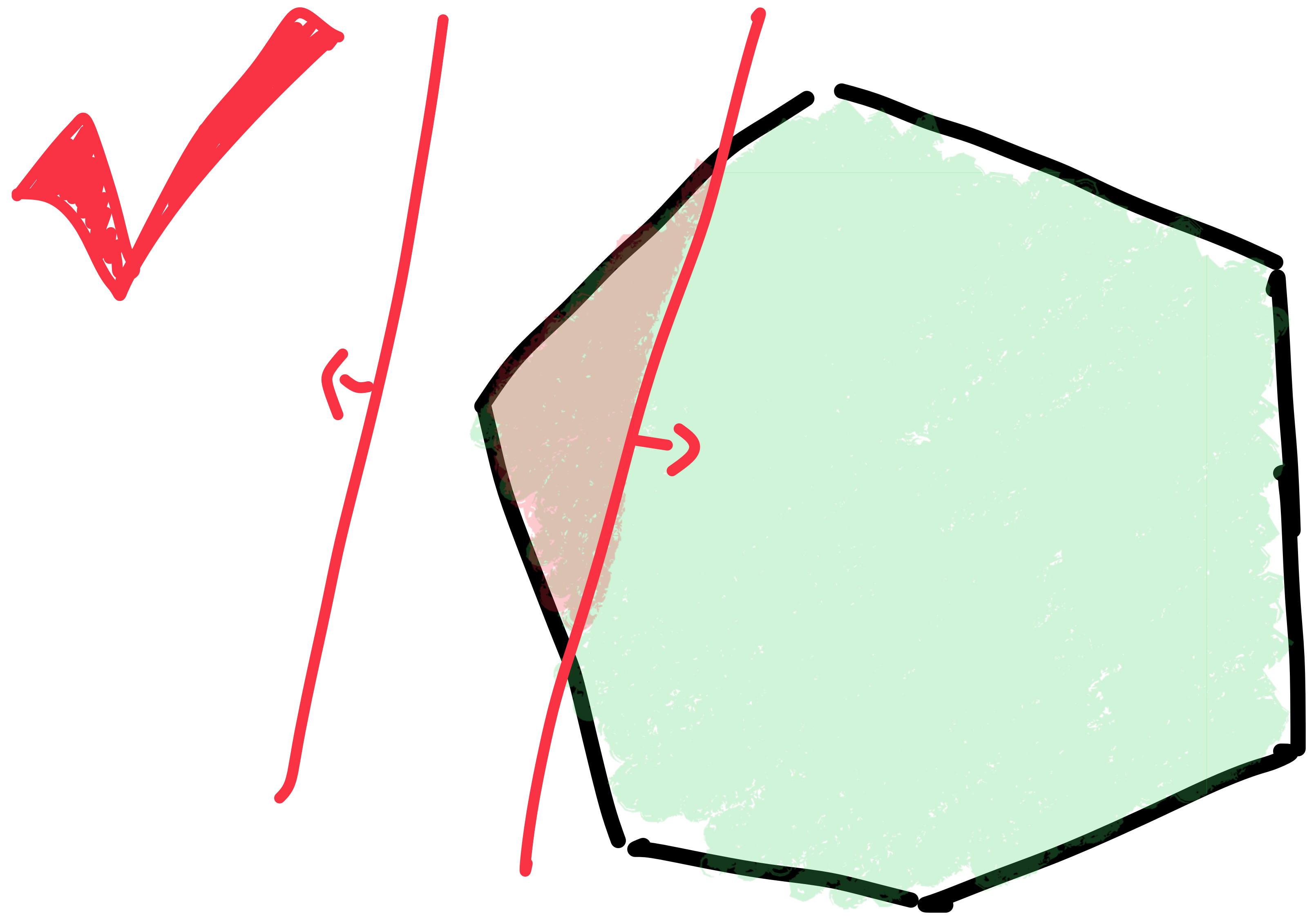


Stabbing Planes vs Cutting Planes

- Branch-and-cut allows Cutting planes deductions
  - → Superfluous as SP can simulate CP

Pathlike SP: A SP query $(ax \leq b-1,\ ax \geq b)$ at P is pathlike if $P \cap \{ax \leq b-1\} = \emptyset$ or $P \cap \{ax \geq b\}$

Clm: pathlike SP = CP



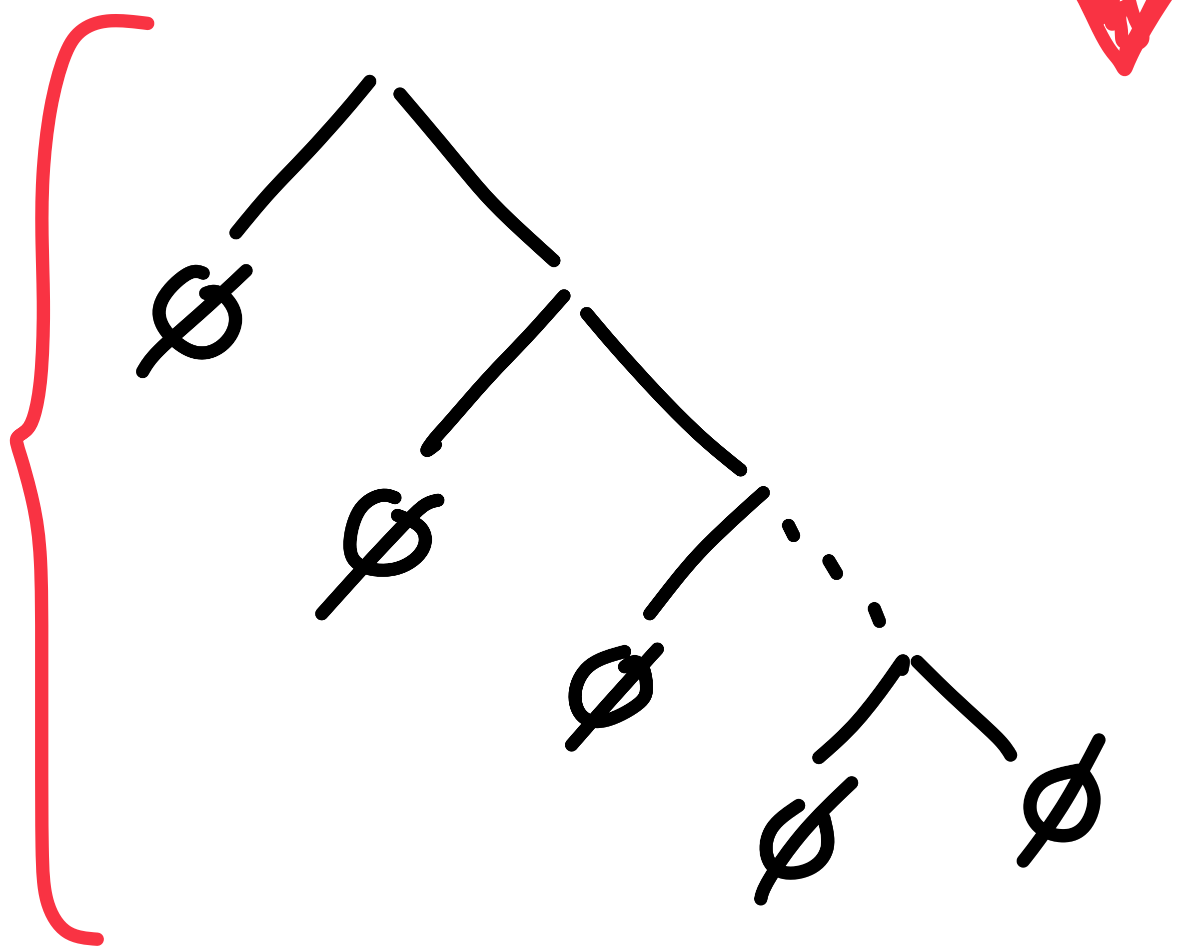
Stabbing Planes vs Cutting Planes

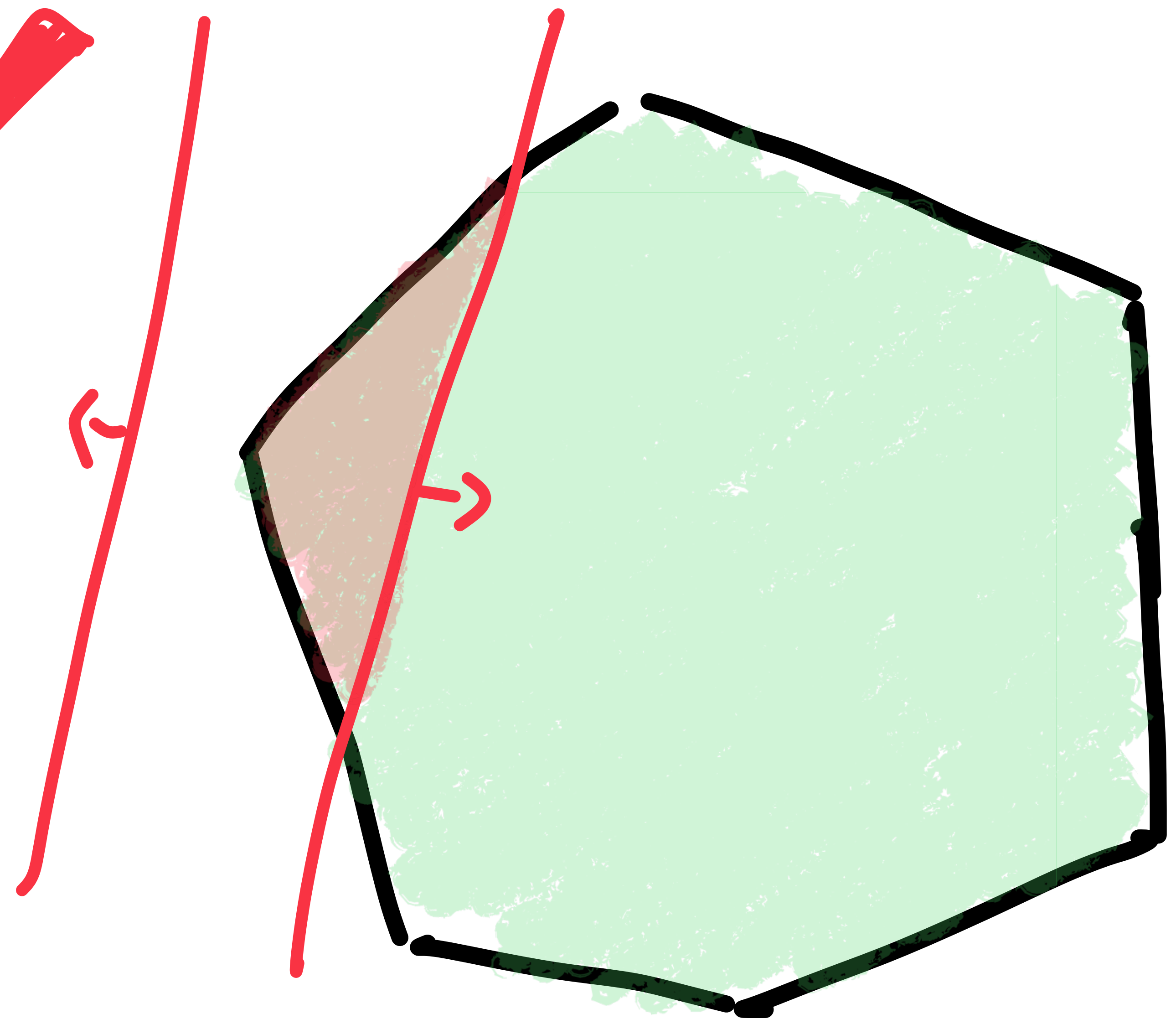
- Branch-and-cut allows Cutting planes deductions
  - → Superfluous as SP can simulate CP

Pathlike SP: A SP query $(ax \leq b-1,\ ax \geq b)$ at P is pathlike if $P \cap \{ax \leq b-1\} = \emptyset$ or $P \cap \{ax \geq b\}$

Clm: pathlike SP = CP

Pathlike SP Proof





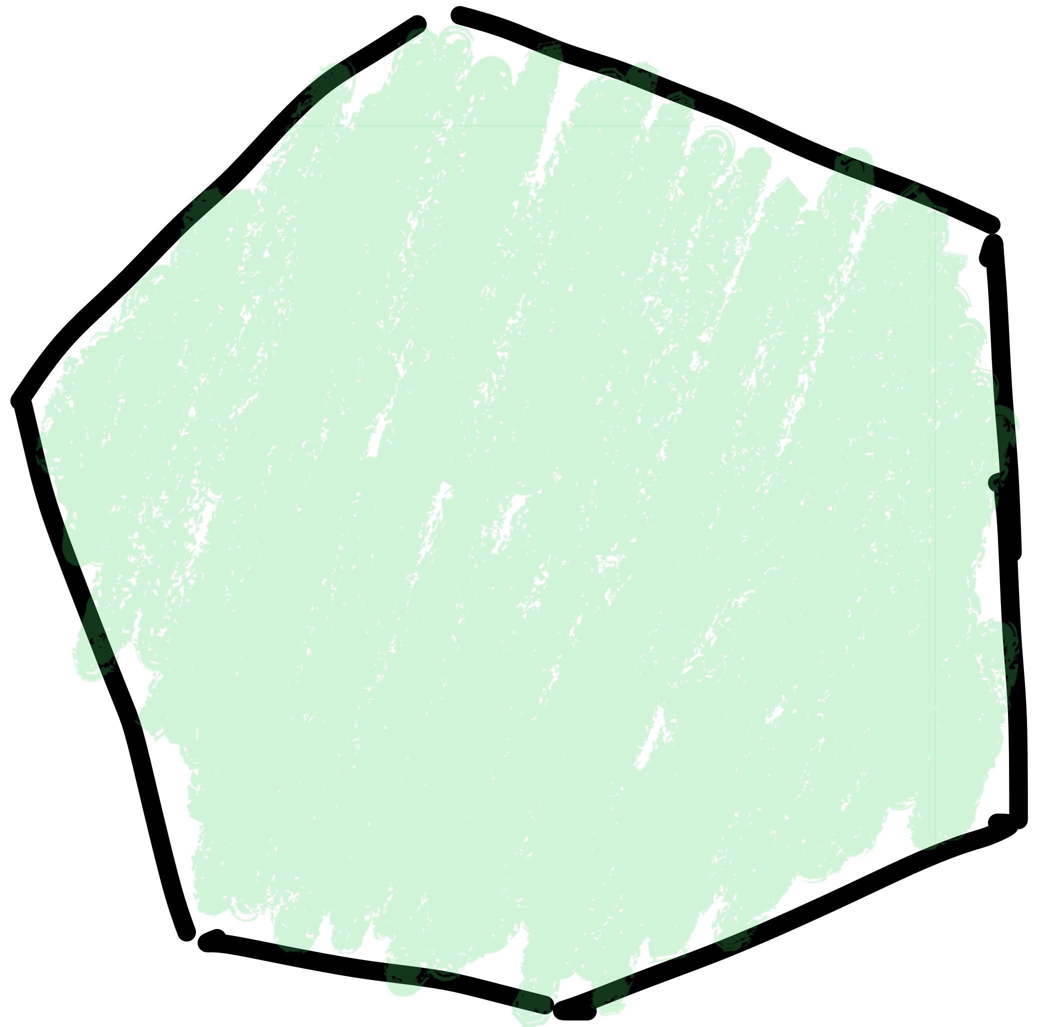
Stabbing Planes vs Cutting Planes

- Branch-and-cut allows Cutting planes deductions
  - → Superfluous as SP can simulate CP

Pathlike SP: A SP query $(ax \leq b-1, ax \geq b)$ at P is pathlike if $P \cap \{ax \leq b-1\} = \emptyset$ or $P \cap \{ax \geq b\}$

Clm: pathlike SP = CP

Pf: Show each CG-cut is a pathlike query



Stabbing Planes vs Cutting Planes

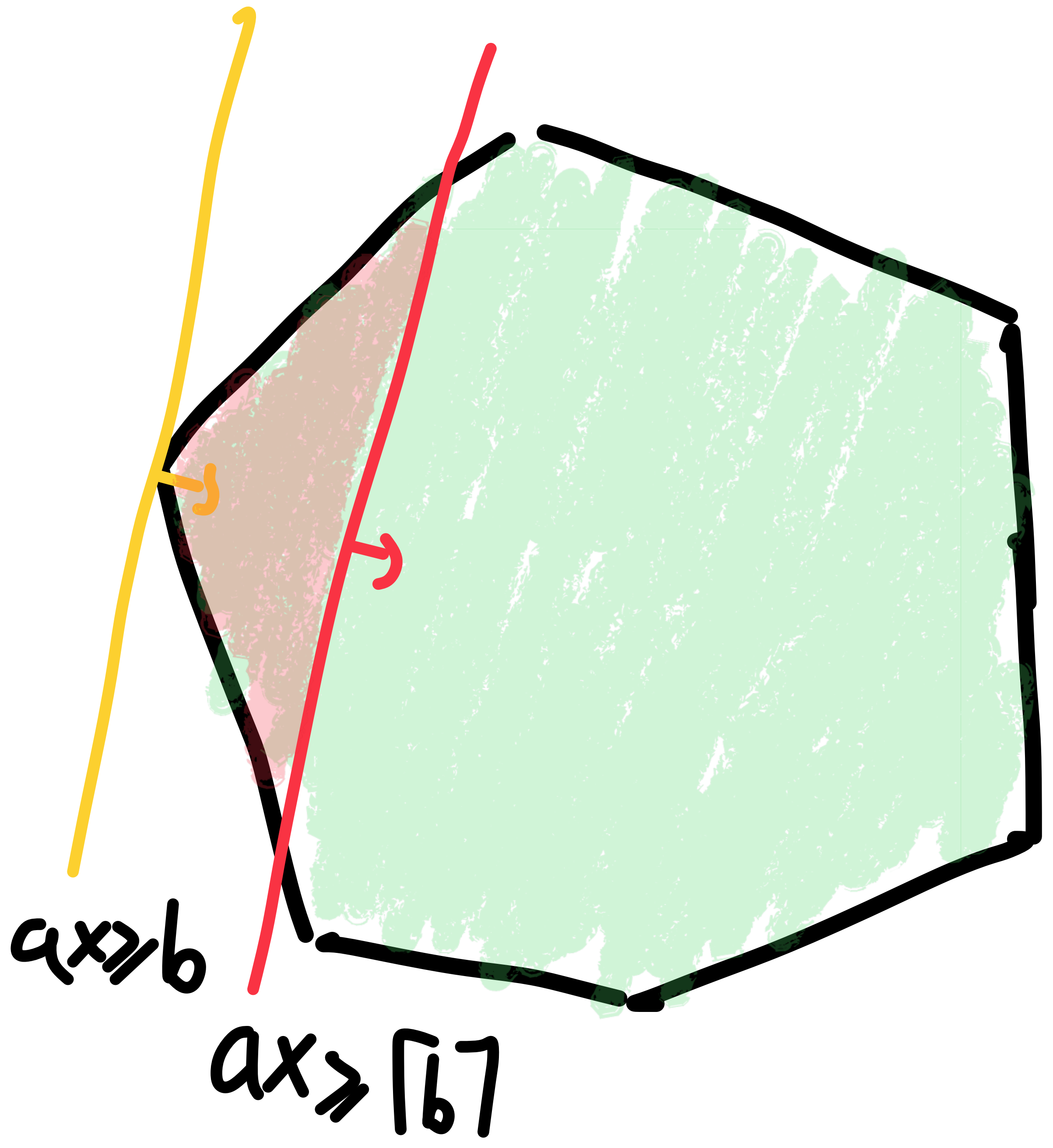
- Branch-and-cut allows Cutting planes deductions
  - → Superfluous as SP can simulate CP

Pathlike SP: A SP query $(ax \le b-1,\ ax \ge b)$ at P is pathlike if $P \cap \{ax \le b-1\} = \emptyset$ or $P \cap \{ax \ge b\}$

Clm: pathlike SP = CP

Pf: Show each CG-cut is a pathlike query

$ax \ge \lceil b \rceil$ is a CG-cut for P if $ax \ge b$ is valid for P



Stabbing Planes vs Cutting Planes

- Branch-and-cut allows Cutting planes deductions
  → Superfluous as SP can simulate CP

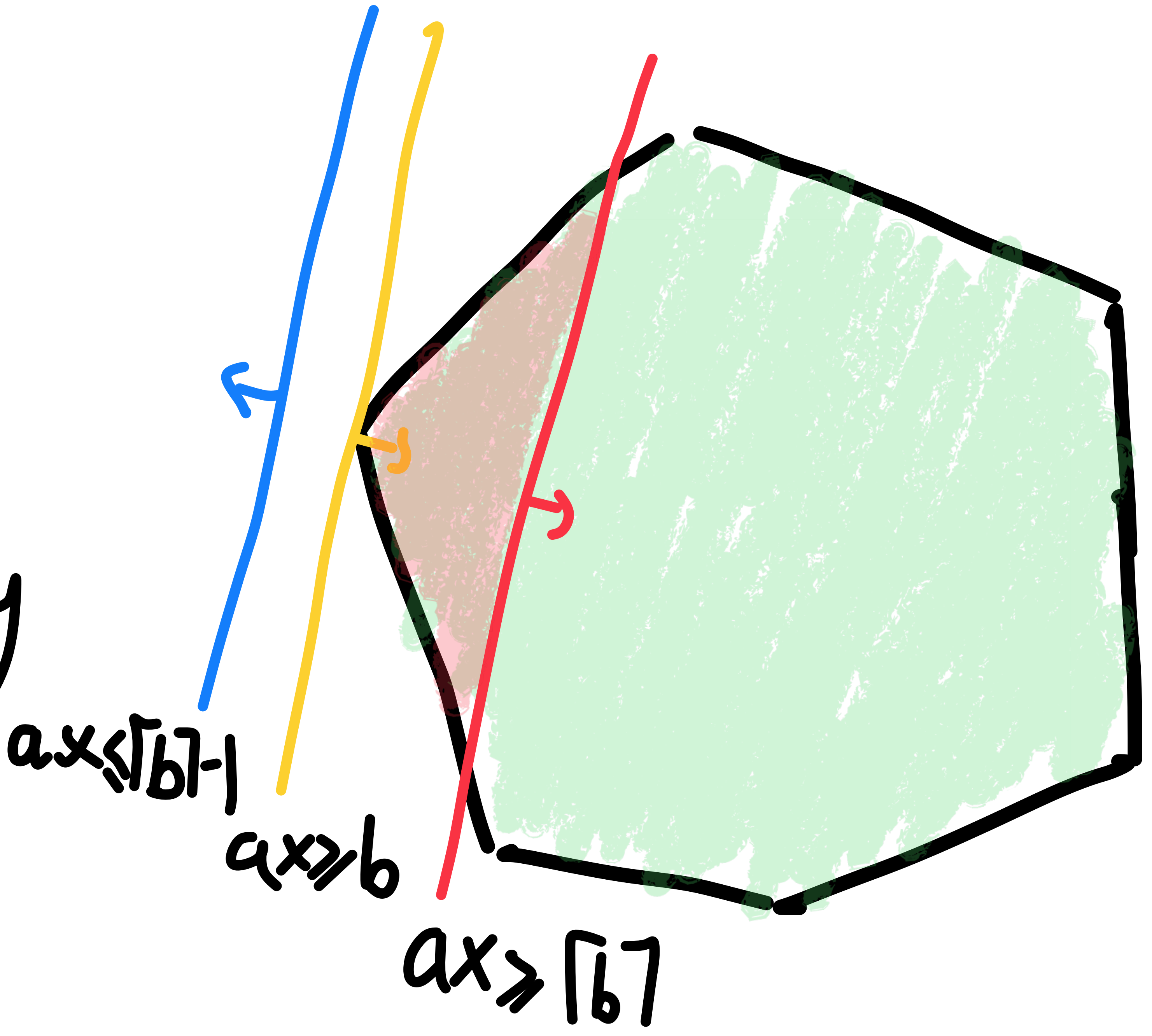
Pathlike SP: A SP query $(ax \leq b-1,\ ax \geq b)$ at P is pathlike if $P \cap \{ax \leq b-1\} = \emptyset$ or $P \cap \{ax \geq b\}$

Clm: pathlike SP = CP

Pf: Show each CG-cut is a pathlike query

$ax \geq \lceil b \rceil$ is a CG-cut for P if $ax \geq b$ is valid for P

$\Rightarrow P \cap \{ax \leq \lceil b \rceil - 1\} = \emptyset$



Stabbing Planes vs Cutting Planes

- Branch-and-cut allows Cutting planes deductions
  - → Superfluous as SP can simulate CP

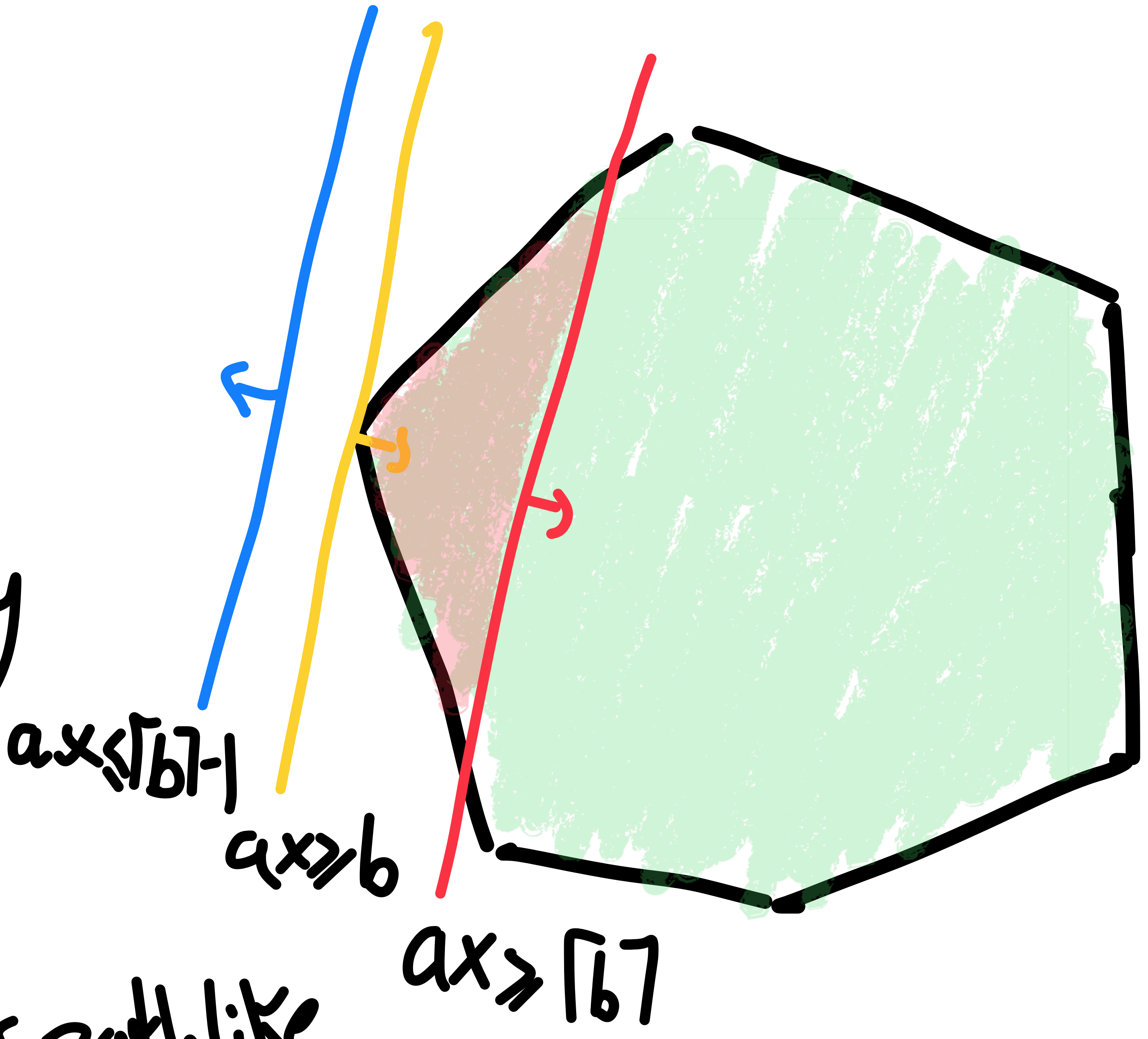
Pathlike SP: A SP query $(ax \le b-1,\ ax \ge b)$ at P is pathlike if $P \cap \{ax \le b-1\} = \emptyset$ or $P \cap \{ax \ge b\}$

Clm: pathlike SP = CP

Pf: Show each CG-cut is a pathlike query

$ax \ge \lceil b \rceil$ is a CG-cut for P if $ax \ge b$ is valid for P

$\Rightarrow P \cap \{ax \le \lceil b \rceil - 1\} = \emptyset$ & $(ax \le \lceil b \rceil - 1,\ ax \ge \lceil b \rceil)$ is pathlike



Stabbing Planes vs Cutting Planes

Q Can we separate CP from SP?

Stabbing Planes vs Cutting Planes

Q Can we separate CP from SP?

- Candidate: Tseitin formulas

Stabbing Planes vs Cutting Planes

Q Can we separate CP from SP?

- ▷ Candidate: Tseitin formulas
  - ⟶ quasipoly-size SP proofs of Tseitin [BFI+18]

Stabbing Planes vs Cutting Planes

Q Can we separate CP from SP?

- Candidate: Tseitin formulas
  - → quasipoly-size SP proofs of Tseitin [BFI+18]
  - → Conjectured to be hard for CP

Stabbing Planes vs Cutting Planes

Q Can we separate CP from SP?

▷ Candidate: Tseitin formulas

→ quasipoly-size SP proofs of Tseitin [BFI+18]

→ Conjectured to be hard for CP

[DT20] There are quasi-poly size CP proofs of Tseitin

Stabbing Planes vs Cutting Planes

Q Can we separate CP from SP?

- ▷ Candidate: Tseitin formulas
  - ⟶▷ quasipoly-size SP proofs of Tseitin [BFI+18]
  - ⟶▷ Conjectured to be hard for CP

[DT20] There are quasi-poly size CP proofs of Tseitin

- ▷ Translate the SP proof of Tseitin into CP

Stabbing Planes vs Cutting Planes

Q Can we separate CP from SP?

- Candidate: Tseitin formulas
  - quasipoly-size SP proofs of Tseitin [BFI+18]
  - Conjectured to be hard for CP

[DT20] There are quasi-poly size CP proofs of Tseitin

- Translate the SP proof of Tseitin into CP

Q Can every SP proof be translated into CP?

Stabbing Planes vs Cutting Planes

Thm: Every SP^* proof can be quasipolynomially translated into CP

Stabbing Planes vs Cutting Planes

Thm: Every SP^* proof can be quasipolynomially translated into CP

Coefficients in queries are quasi-poly bounded

Stabbing Planes vs Cutting Planes

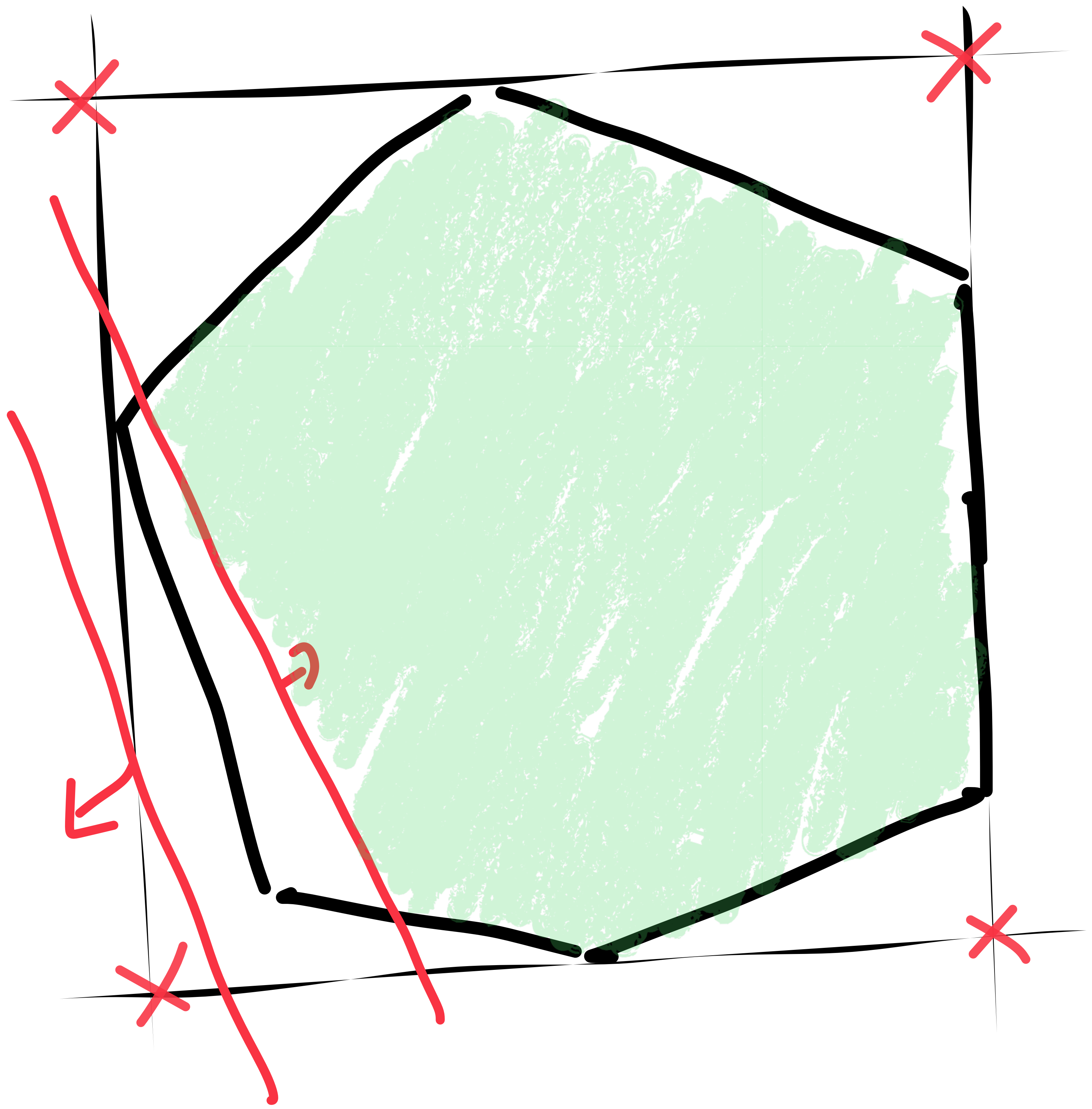
Thm: Every SP* proof can be quasipolynomially translated into CP

Pf: a) CP = pathlike SP = facelike SP

Stubbing Planes vs Cutting Planes

Thm: Every SP^* proof can be quasipolynomially translated into CP

Pf: a) CP = pathlike SP = facelike SP

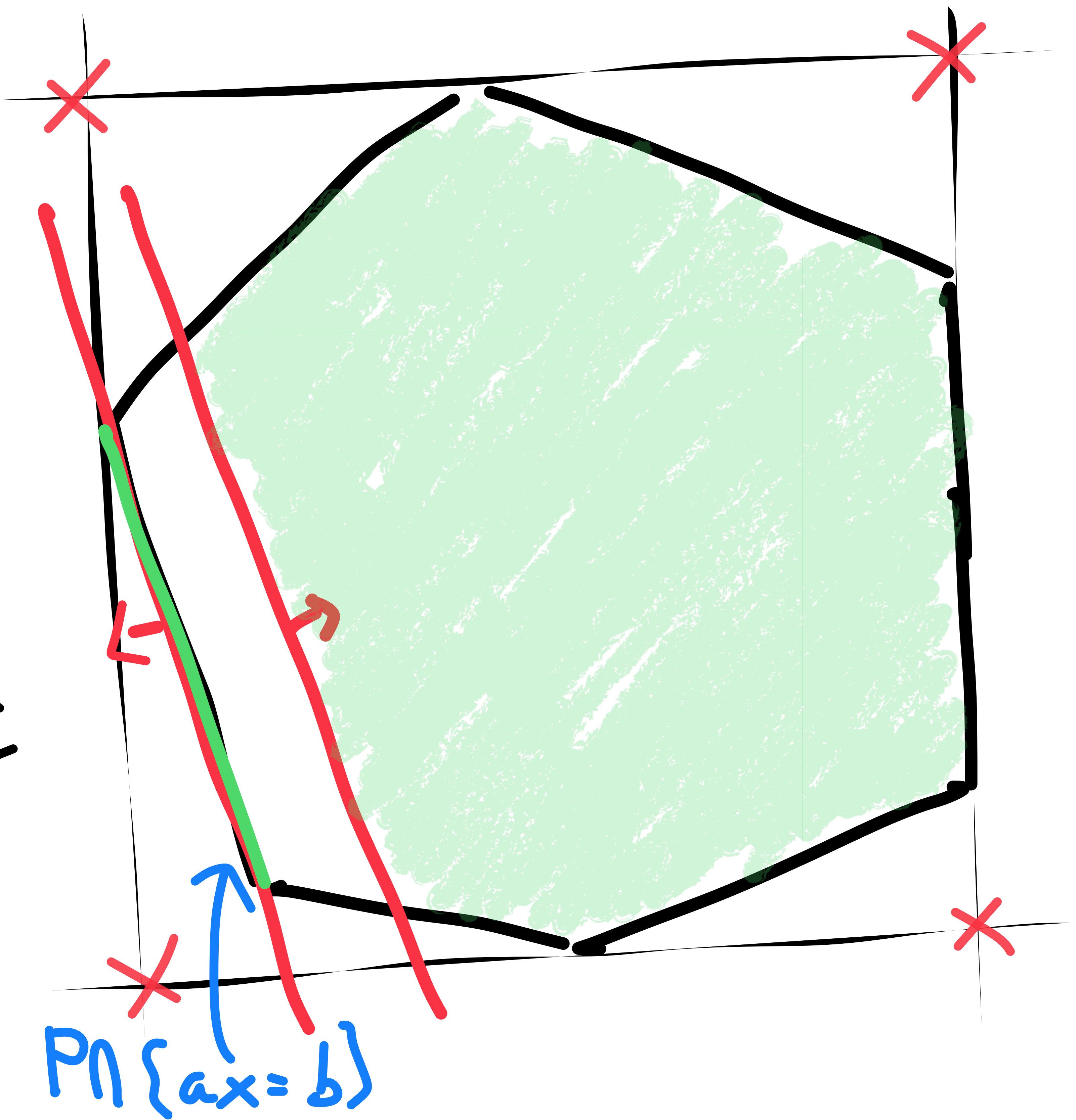


Pathlike SP: A query $(ax \leq b-1, ax \geq b)$ is pathlike if
$P \cap \{ax \leq b-1\} = \emptyset$ or $P \cap \{ax \geq b\} = \emptyset$

Stabbing Planes vs Cutting Planes

Thm: Every SP* proof can be quasipolynomially translated into CP

Pf: a) CP = pathlike SP = facelike SP



Pathlike SP: A query $(ax \leq b-1, ax \geq b)$ is pathlike if
$P \cap \{ax \leq b-1\} = \emptyset$ or $P \cap \{ax \geq b\} = \emptyset$

Facelike SP: A query $(ax \leq b-1, ax \geq b)$ is facelike if
$P \cap \{ax < b-1\} = \emptyset$ or $P \cap \{ax > b\} = \emptyset$

Stabbing Planes vs Cutting Planes

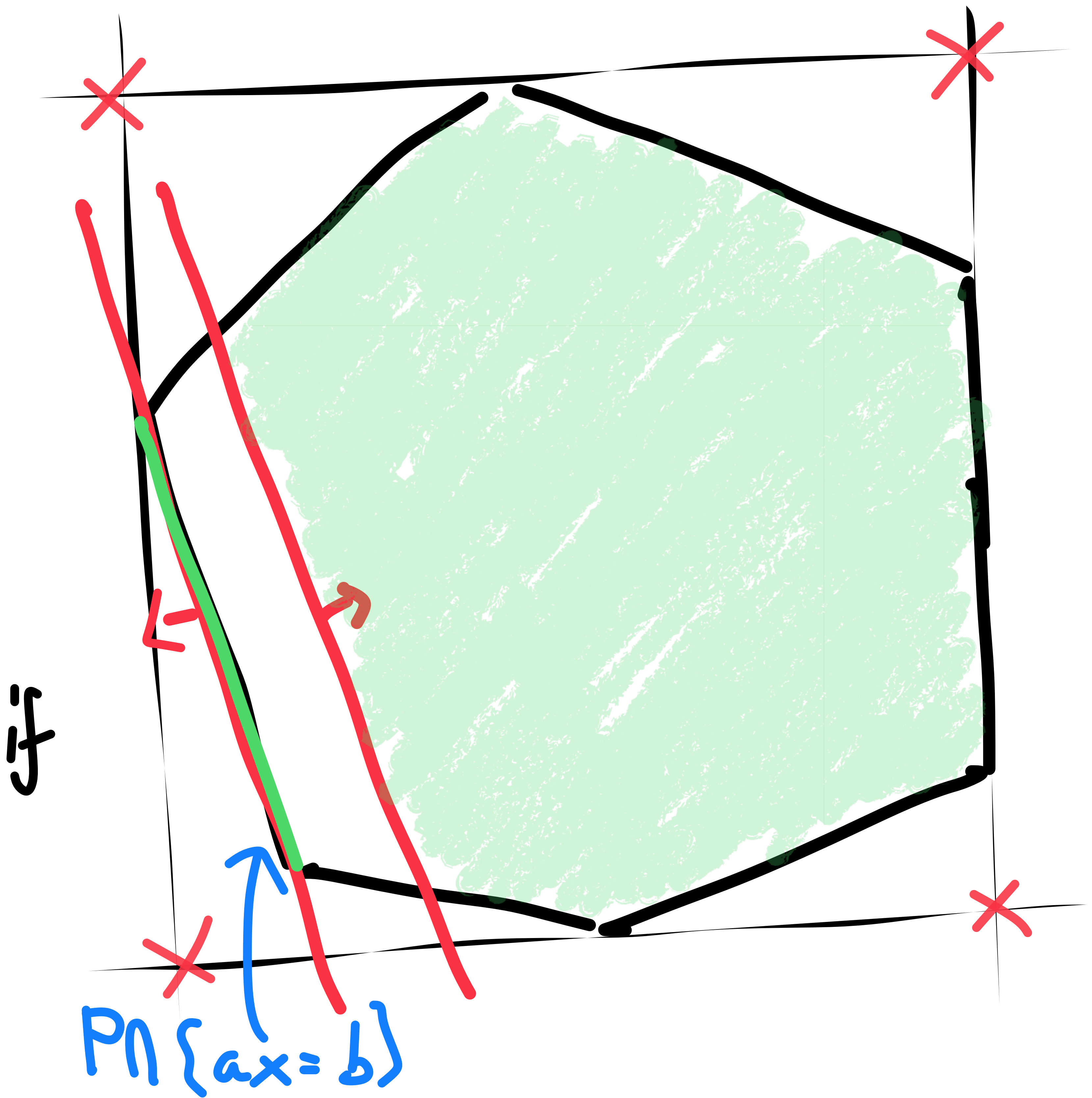
Thm: Every SP^* proof can be quasipolynomially translated into CP

Pf: a) CP = pathlike SP = facelike SP

Pathlike SP: A query $(ax \leq b-1, ax \geq b)$ is pathlike if $P \cap \{ax \leq b-1\} = \emptyset$ or $P \cap \{ax \geq b\} = \emptyset$

Facelike SP: A query $(ax \leq b-1, ax \geq b)$ is facelike if $P \cap \{ax < b-1\} = \emptyset$ or $P \cap \{ax > b\} = \emptyset$

ie $P \cap \{ax \leq b-1\}$ or $P \cap \{ax \geq b\}$ is a face.



Stabbing Planes vs Cutting Planes

Thm: Every SP^* proof can be quasipolynomially translated into CP

Pf: a) CP = pathlike SP = facelike SP

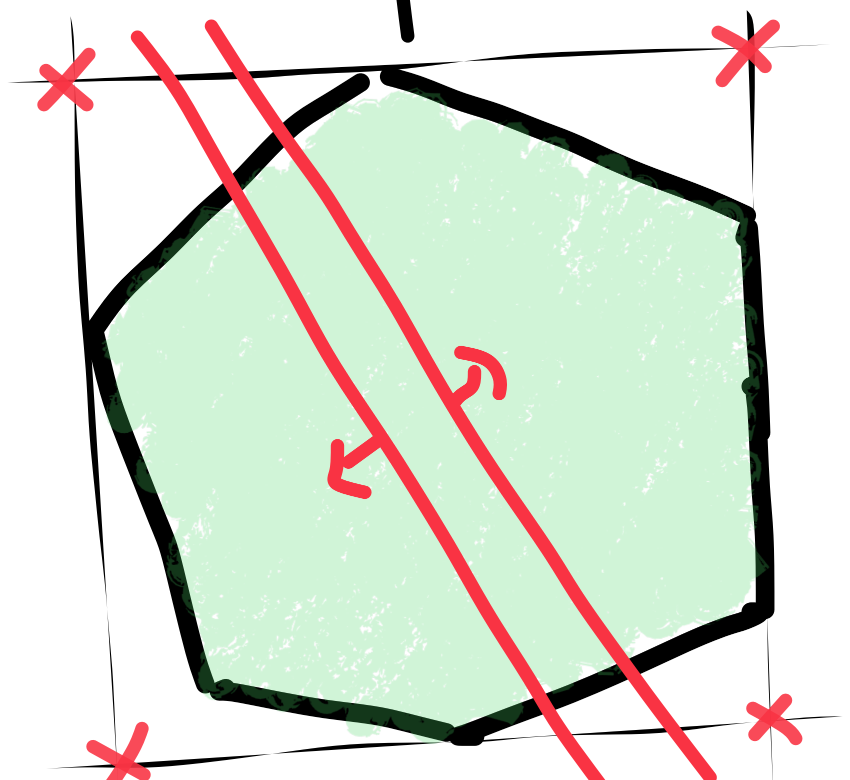
b) Any SP^* proof can be made facelike with a quasi-poly blowup in size.

Stubbing Planes vs Cutting Planes

Thm: Every SP* proof can be quasipolynomially translated into CP

Pf: a) CP = pathlike SP = facelike SP

b) Any SP* proof can be made facelike with a quasi-poly blowup in size.

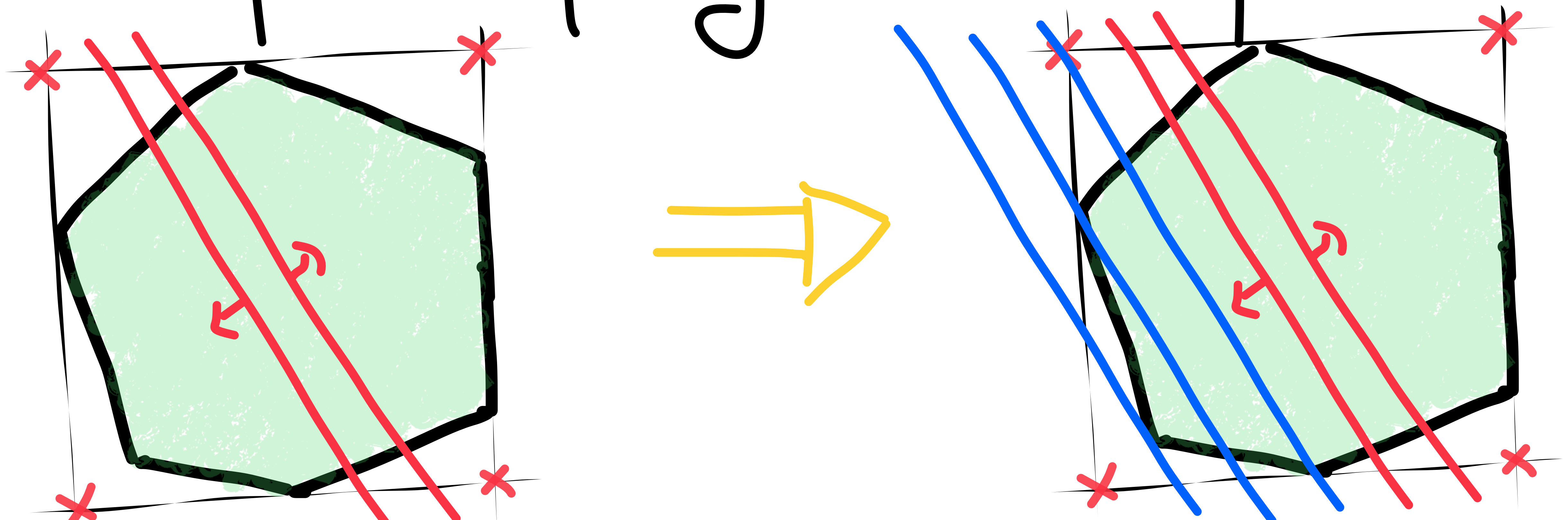


Stubbing Planes vs Cutting Planes

Thm: Every SP* proof can be quasipolynomially translated into CP

Pf: a) CP = pathlike SP = facelike SP

b) Any SP* proof can be made facelike with a quasi-poly blowup in size.



Stabbing Planes vs Cutting Planes

Thm: Every SP^* proof can be quasipolynomially translated into CP

Cor: Exponential lower bounds on SP^*

Stabbing Planes vs Cutting Planes

Thm: Every SP^* proof can be quasipolynomially translated into CP

Cor: Exponential lower bounds on SP^*

Cor: Cutting Planes has quasi-poly size proofs of any system of linear equations over a finite field

Depth of CP Proofs of Tseitin

- Both (ours and [DT20]) CP proofs of Tseitin are quasi-polynomially deep

Depth of CP Proofs of Tseitin

- Both (ours and [DT20]) CP proofs of Tseitin are quasi-polynomially deep
- Any CNF can be refuted in depth n

Depth of CP Proofs of Tseitin

- Both (ours and [DT20]) CP proofs of Tseitin are quasi-polynomially deep
- Any CNF can be refuted in depth n

Conjecture: There is a family of CNFs $\{F_n\}$ which has quasi-poly CP proofs, but any quasi-poly CP proof requires superlinear depth

Towards the Conjecture

Conjecture: There is a family of CNFs $\{F_n\}$ which has quasi-poly CP proofs, but any quasi-poly CP proof requires superlinear depth

Towards the Conjecture

Conjecture: There is a family of CNFs $\{F_n\}$ which has quasi-poly CP proofs, but any quasi-poly CP proof requires superlinear depth

▷ Requires a better understanding of CP depth

Towards the Conjecture

Conjecture: There is a family of CNFs $\{F_n\}$ which has quasi-poly CP proofs, but any quasi-poly CP proof requires superlinear depth

- Requires a better understanding of CP depth
- A new geometric techique for CP depth bounds

Towards the Conjecture

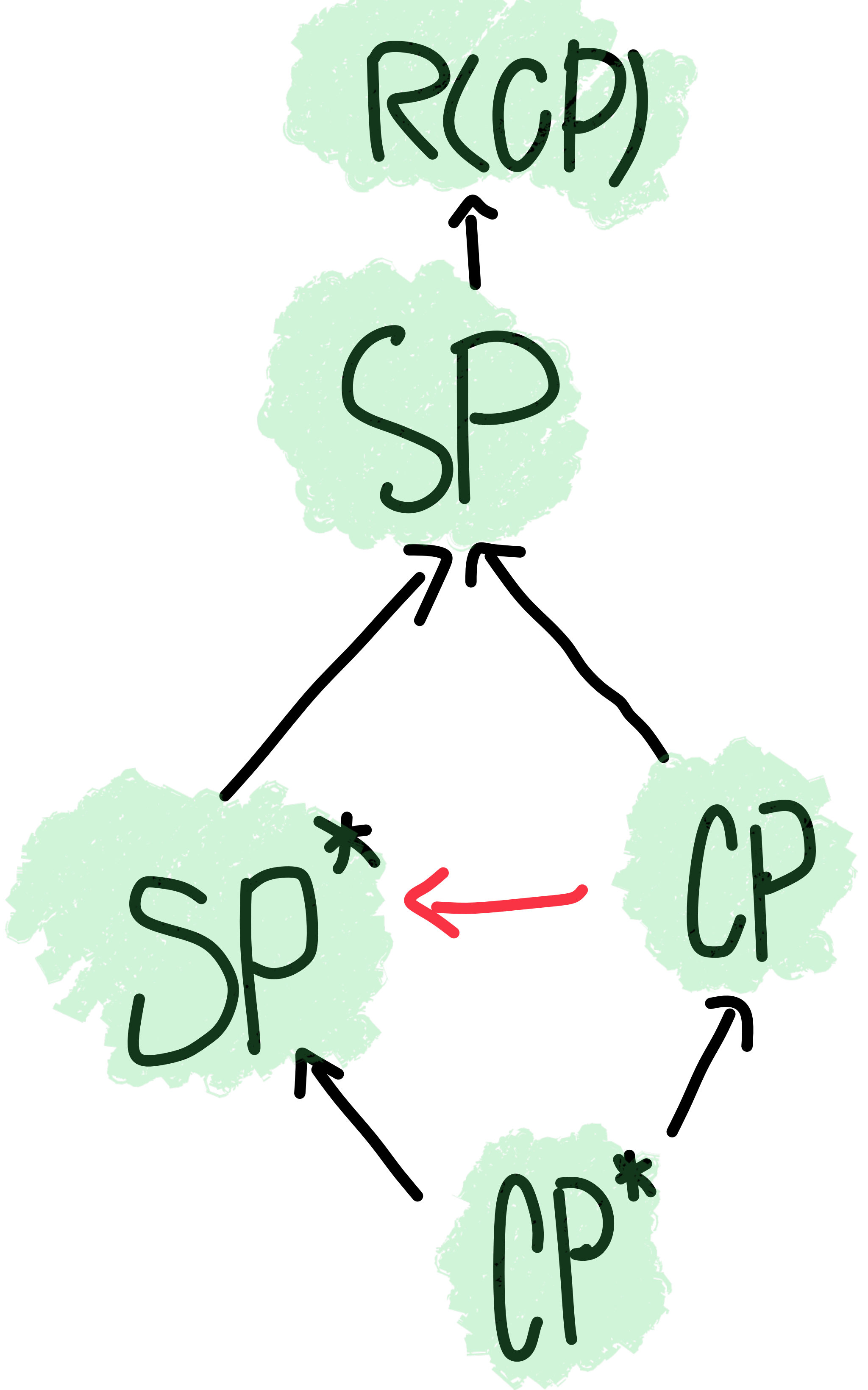
Conjecture: There is a family of CNFs $\{F_n\}$ which has quasi-poly CP proofs, but any quasi-poly CP proof requires superlinear depth

- Requires a better understanding of CP depth
- A new geometric techique for CP depth bounds

Thm: Tseitin requires $\Omega(n)$ depth to refute in Semantic CP

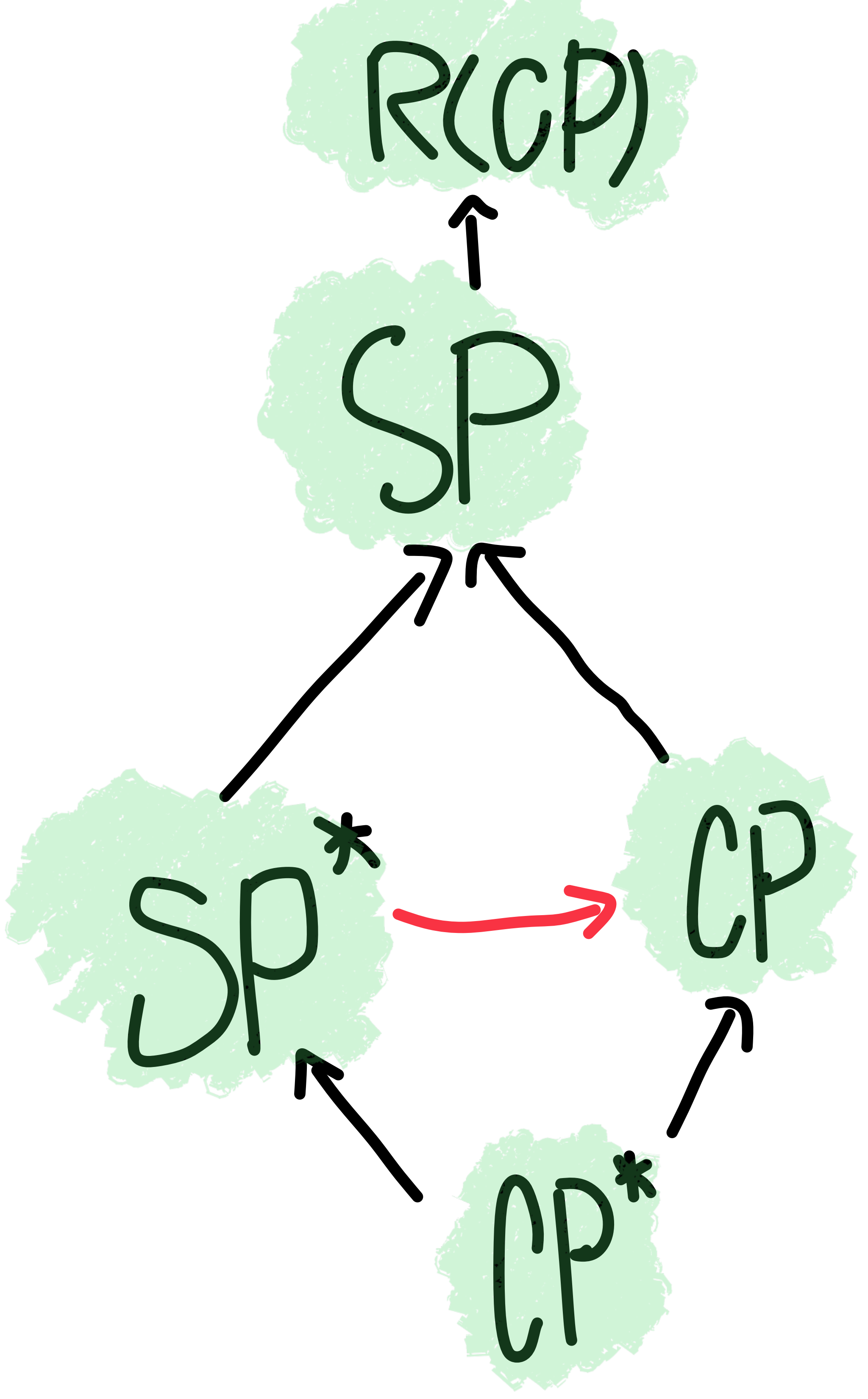
Open Problems

- Can CP p-simulate SP* ?



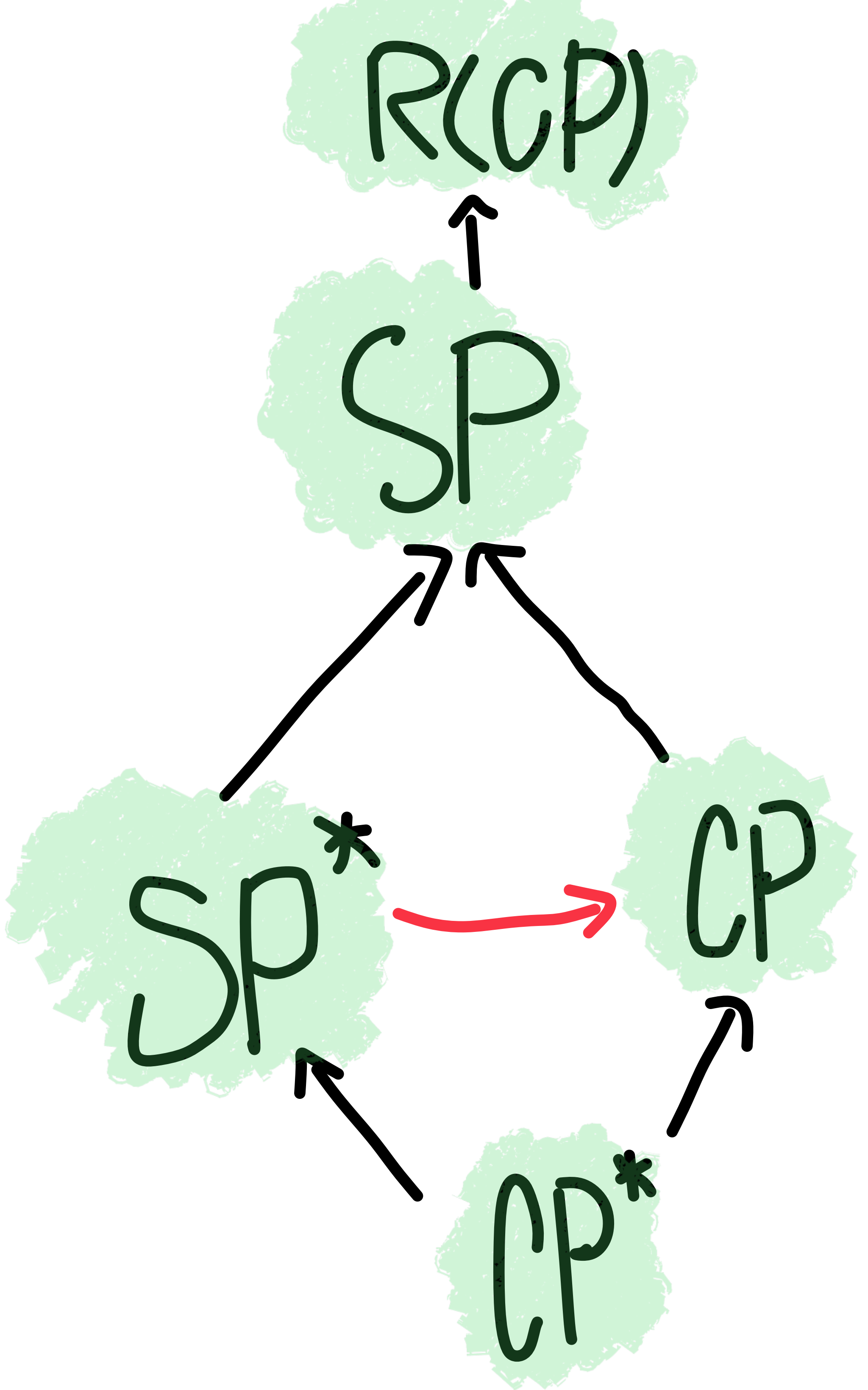
Open Problems

- Can CP p-simulate SP^*?
- Can CP (quasipolynomially) simulate SP?



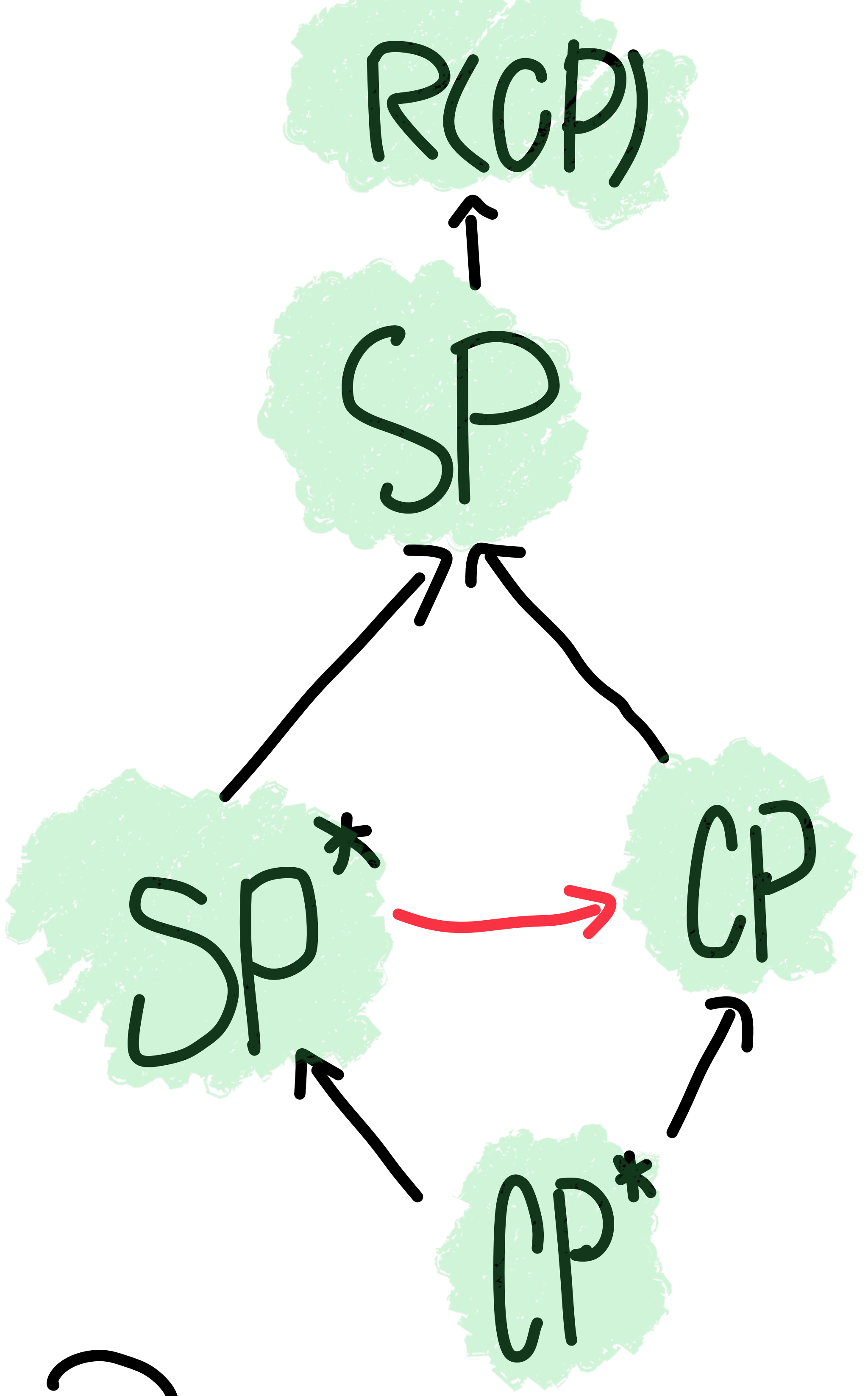
Open Problems

- Can CP p-simulate SP^*?
- Can CP (quasipolynomially) simulate SP?
- Can CP^* (quasipolynomially) simulate SP^*?



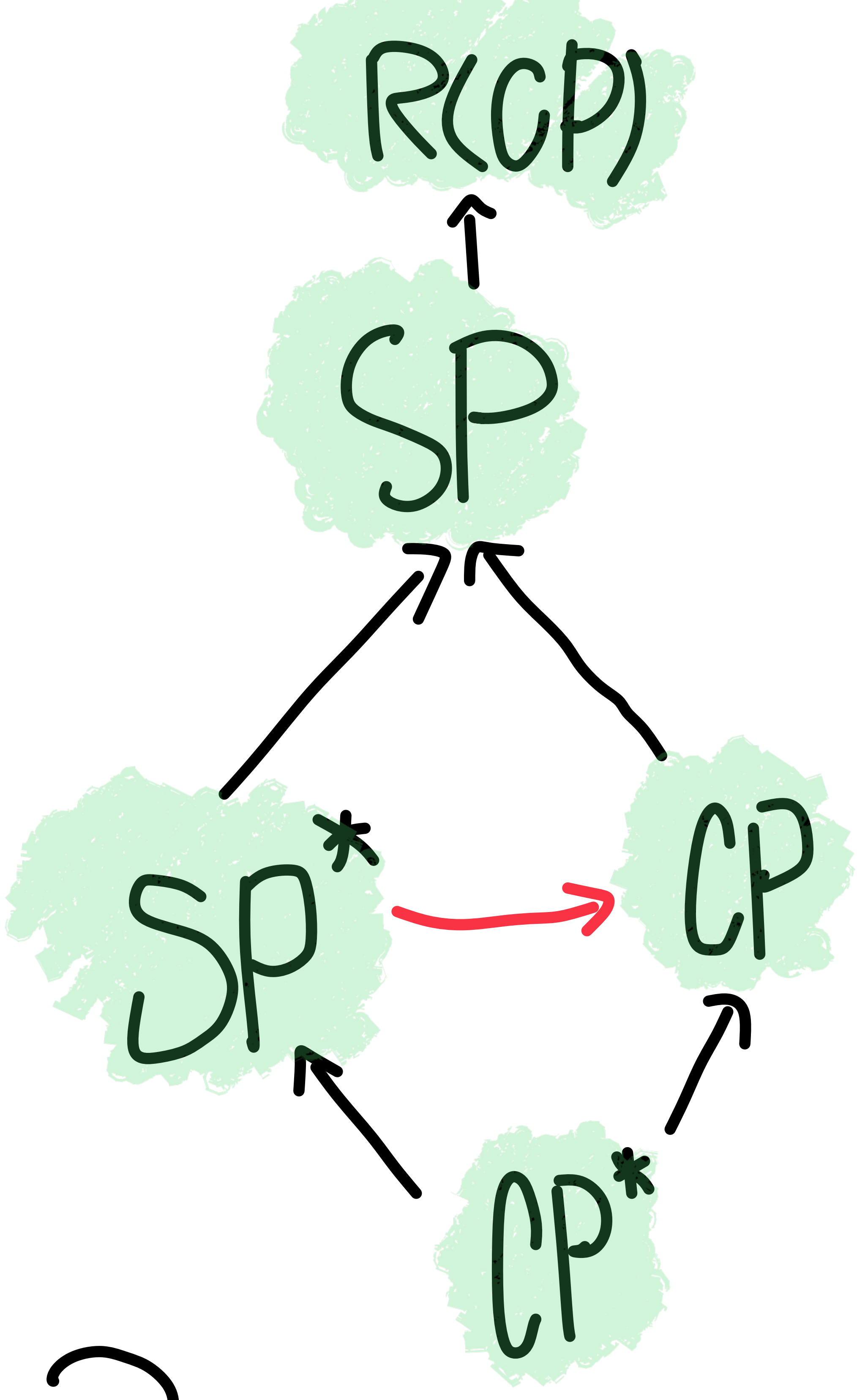
Open Problems

- Can CP p-simulate SP*?
- Can CP (quasipolynomially) simulate SP?
- Can CP* (quasipolynomially) simulate SP*?
- Can SP or CP simulate dag-like SP (R(CP))?



Open Problems

- Can CP p-simulate SP^*?
- Can CP (quasipolynomially) simulate SP?
- Can CP^* (quasipolynomially) simulate SP^*?
- Can SP or CP simulate dag-like SP (R(CP))?
  - → CP cannot p-simulate R(CP) [ABE02]



Open Problems

- Can CP p-simulate SP^*?
- Can CP (quasipolynomially) simulate SP?
- Can CP^* (quasipolynomially) simulate SP^*?
- Can SP or CP simulate dag-like SP (R(CP))?
  - → CP cannot p-simulate R(CP) [ABE02]
- Resolve the Conjecture