Improved Bounds of Integer Solution Counts via Volume and Extending to Mixed-Integer Linear Constraints

Cunjing Ge[1], Armin Biere[2]

[1]Nanjing University, China

[2]University of Freiburg, Germany

NANJING UNIVERSITY

Outline

- **Background**
- Our Approach
  - The Framework of Bounds Approximation
  - Sampling Method
  - Extend Bounds to Mixed-Integer Cases
- Evaluation

Linear Constraints

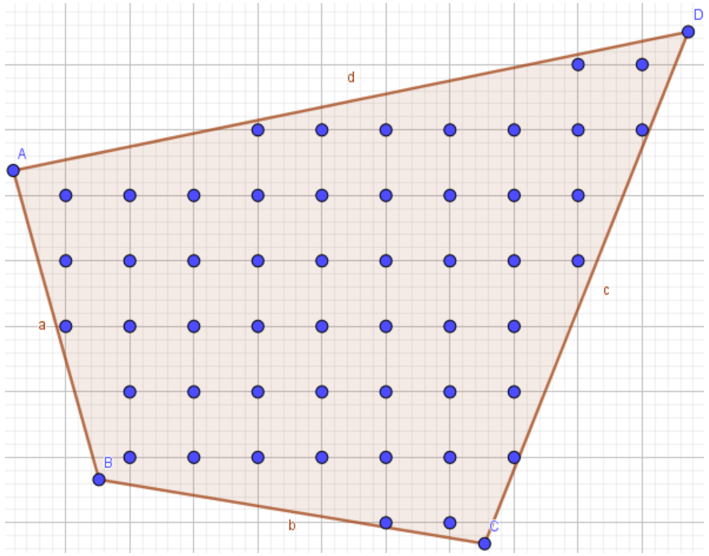
- A LC (linear constraint) is in the form:

$$a_1x_1 + \cdots + a_nx_n \; op \; a_0,$$

where x_i are numeric variables, a_i are constants, and $op \in \{<, \leq, >, \geq, =\}$.

- A set of LCs is in the form: $A\vec{x} \leq \vec{b}$.

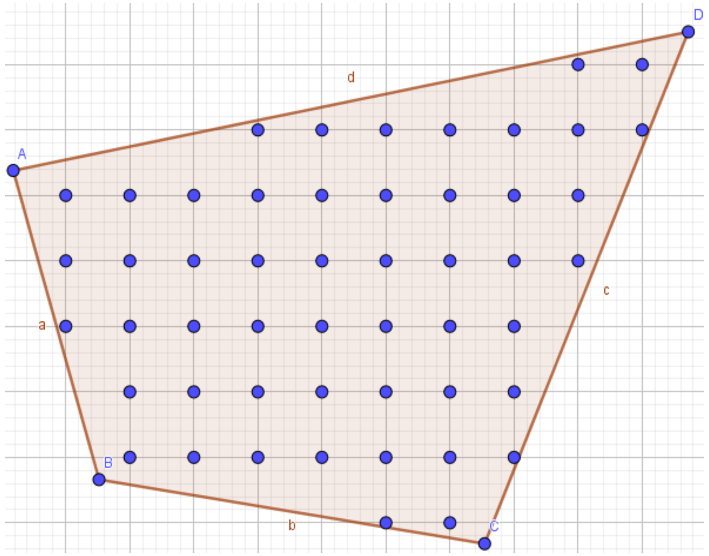
Linear Constraints



- A LC corresponds to a hyperspace.
- A set of LCs corresponds to a convex polytope.

Solution Counting

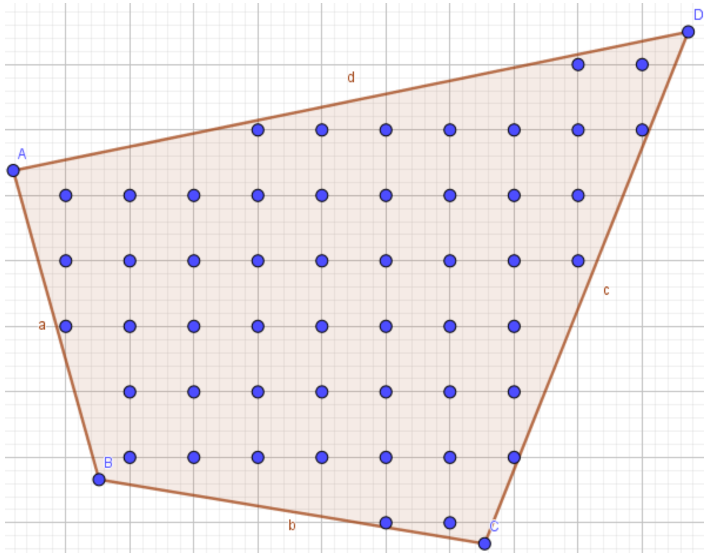
Real domains:
Volume of the polytope



Integer domains:
Counts of the inner integer points

Solution Counting

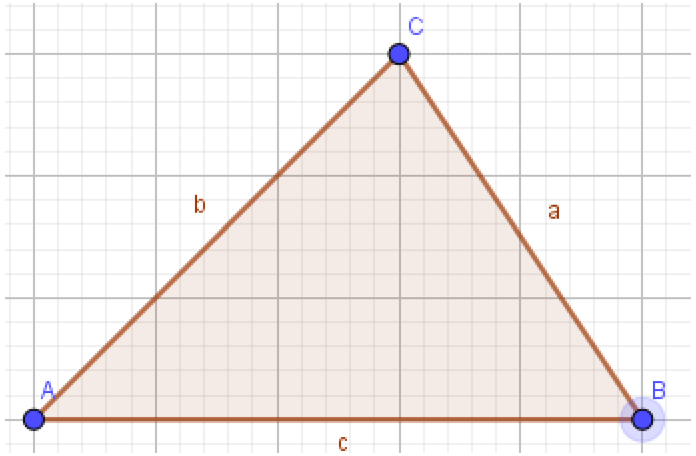
Real domains:
Volume of the polytope



Integer domains:
Counts of the inner integer points

- Let *vol(P)* denote the volume of *P*.
- Let *lat(P)* denote the number of integer points in *P*.

An Example

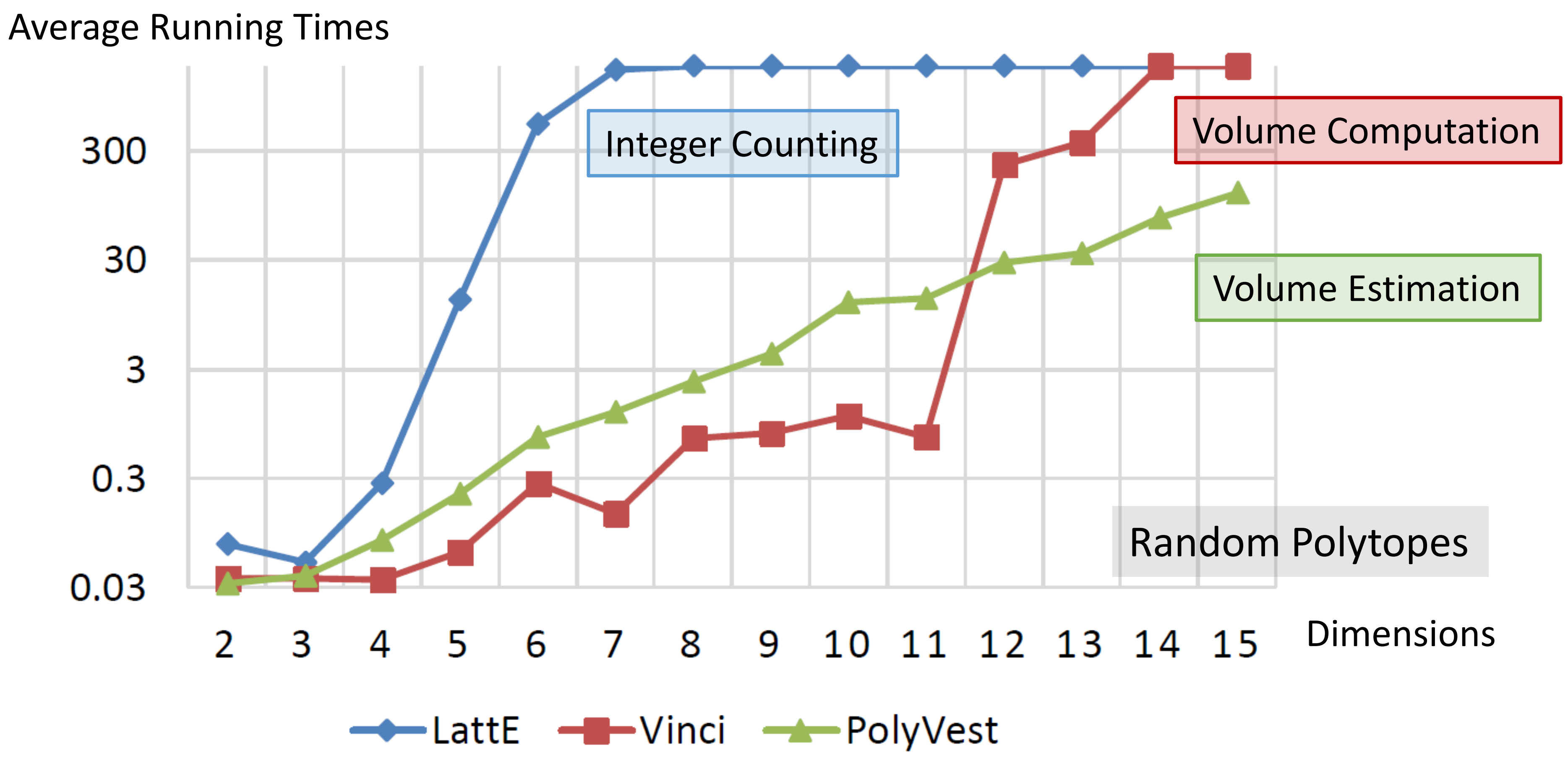


$$(a + b > c) \wedge (a + c > b) \wedge (b + c > a) \wedge$$
$$(1 \leq a \leq 32) \wedge (1 \leq b \leq 32) \wedge (1 \leq c \leq 32).$$

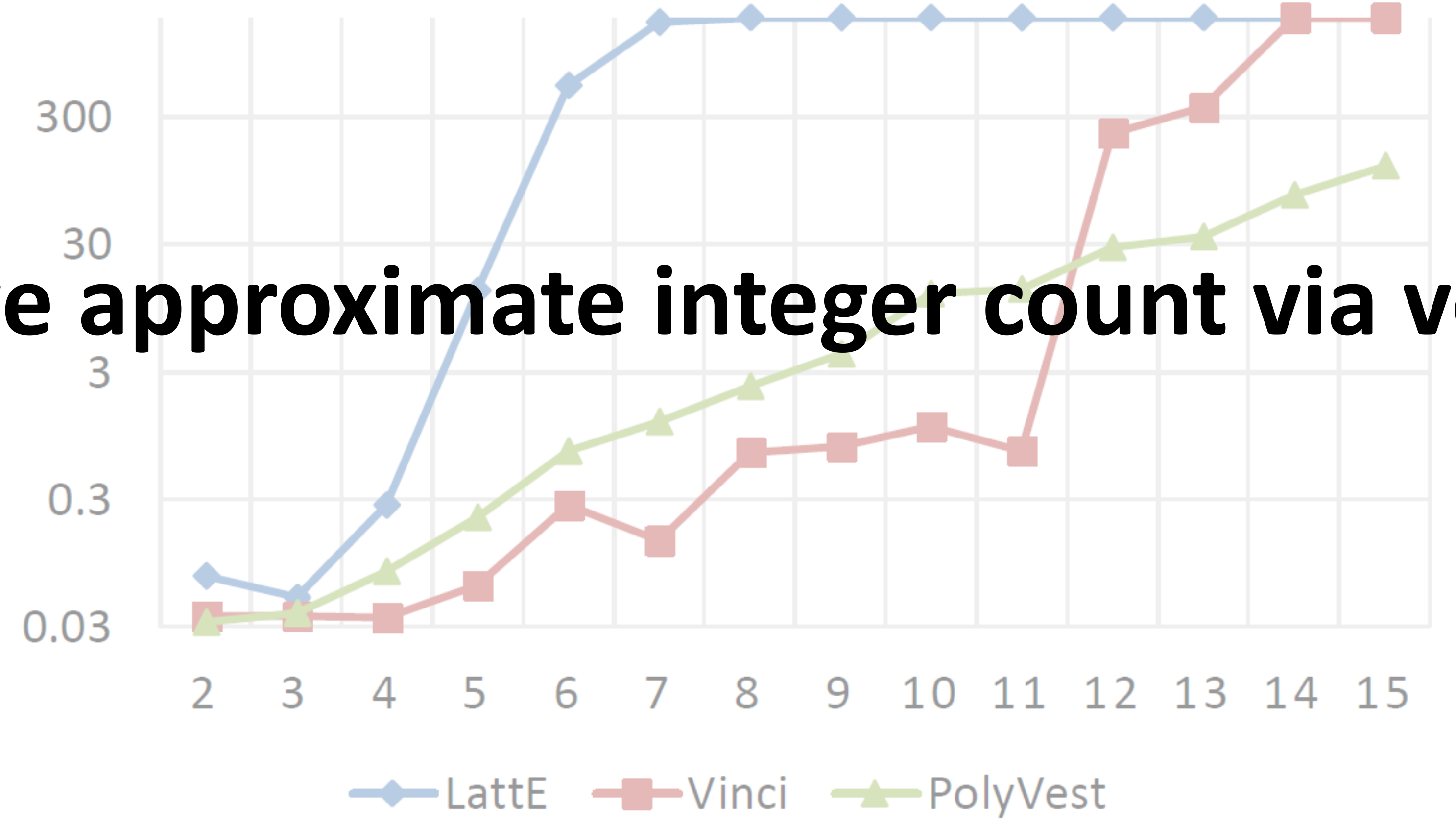
- How many assignments that can form triangles?
  - 16400 integer assignments (integer solutions)
  - Assume a, b, c are reals, then the volume of above formula is $16291 \approx 16400$

- From more experiments, the difference is usually very small.

Integer Count vs. Volume

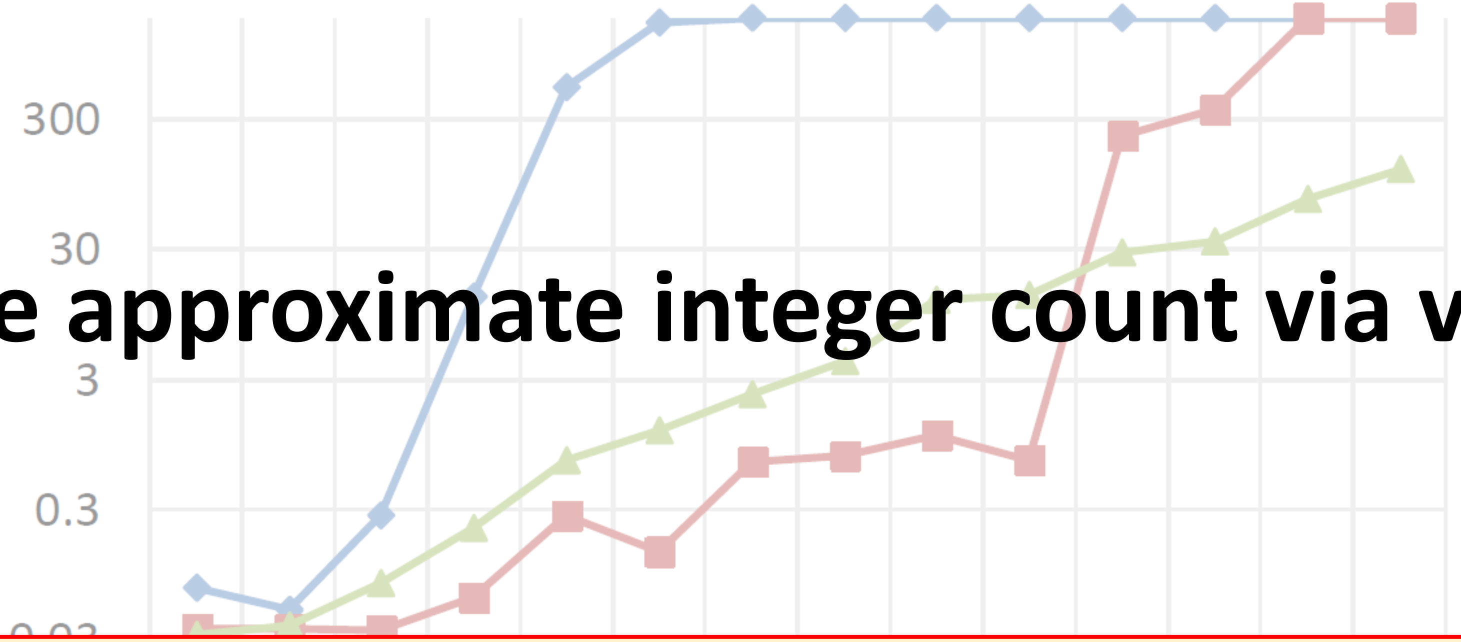


Integer Count vs. Volume



Can we approximate integer count via volume?

Integer Count vs. Volume

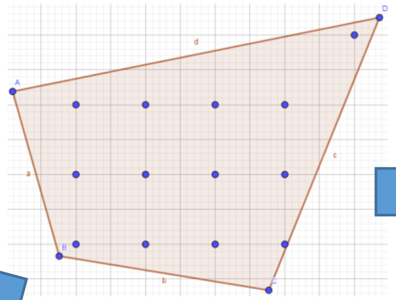


Can we approximate integer count via volume?

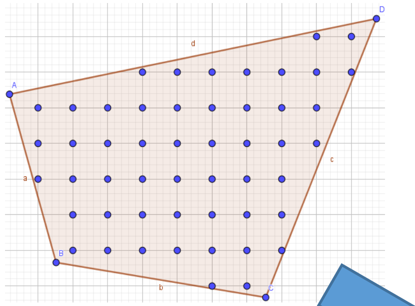
C. Ge, F. Ma, X. Ma, F. Zhang, P. Huang, and J. Zhang. Approximating integer solution counting via space quantification for linear constraints. In Sarit Kraus, editor, Proc. of IJCAI, pages 1697–1703. ijcai.org, 2019. doi:10.24963/ijcai.2019/235.

Integer Count vs. Volume

Integer Counting
- LattE
- barvinok



Discretization



Volume Computation
- Vinci

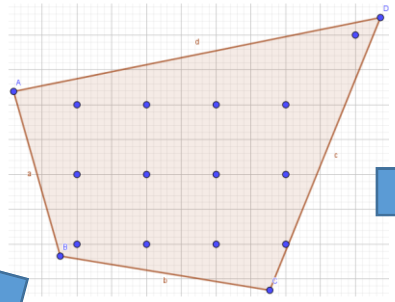
Volume Estimation
- PolyVest

Integer Solution Count

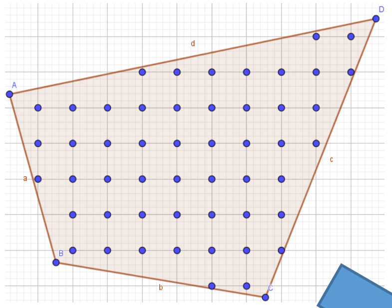
Solution Space Volume

Integer Count vs. Volume

Integer Counting
- LattE
- barvinok



Discretization



Volume Computation
- Vinci

Volume Estimation
- PolyVest

Integer Solution Count

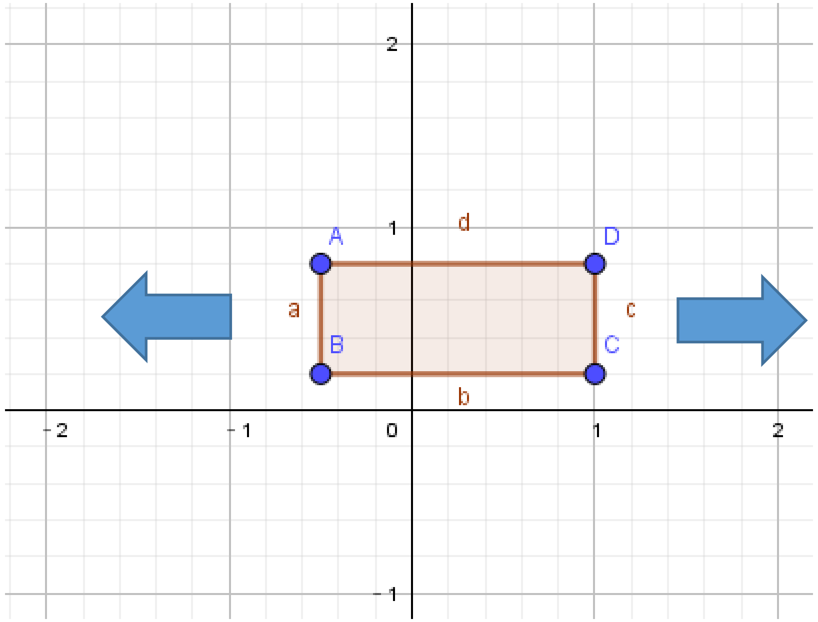
Discretization trick ❌

Obtaining a good approximation is difficult, and sometimes impossible.

Solution Space Volume

Integer Count vs. Volume

lat(ABCD) = 0
vol(ABCD) = 0.9

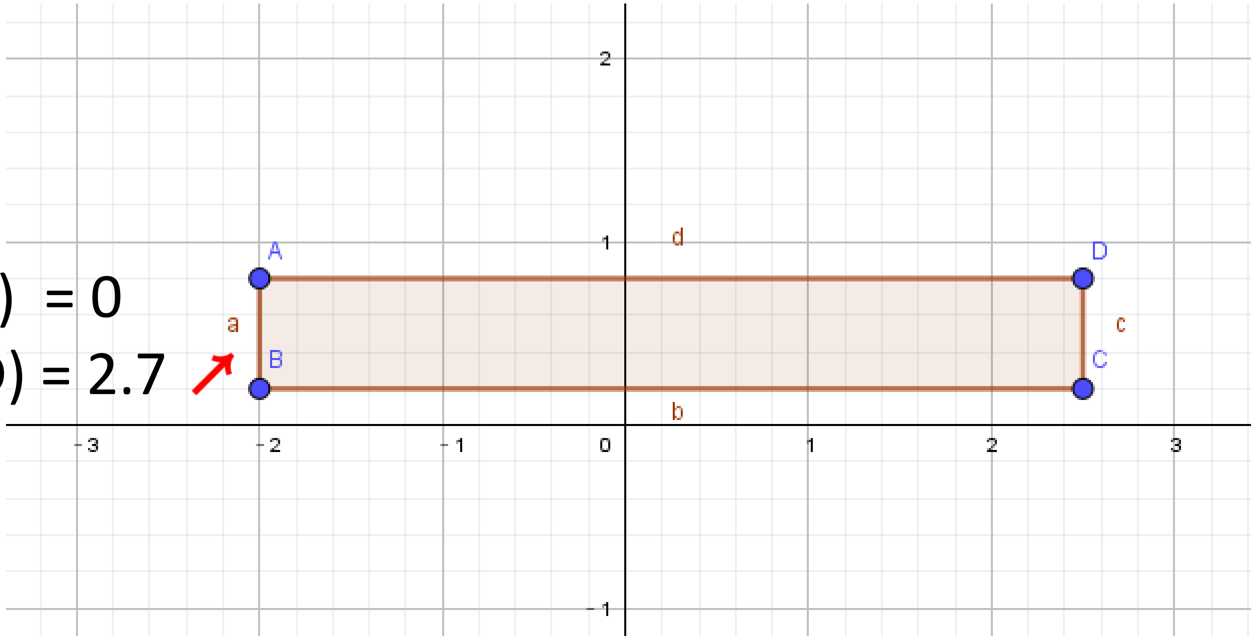


Stretch this “thin” rectangle

Integer Count vs. Volume

lat(ABCD) = 0

vol(ABCD) = 2.7

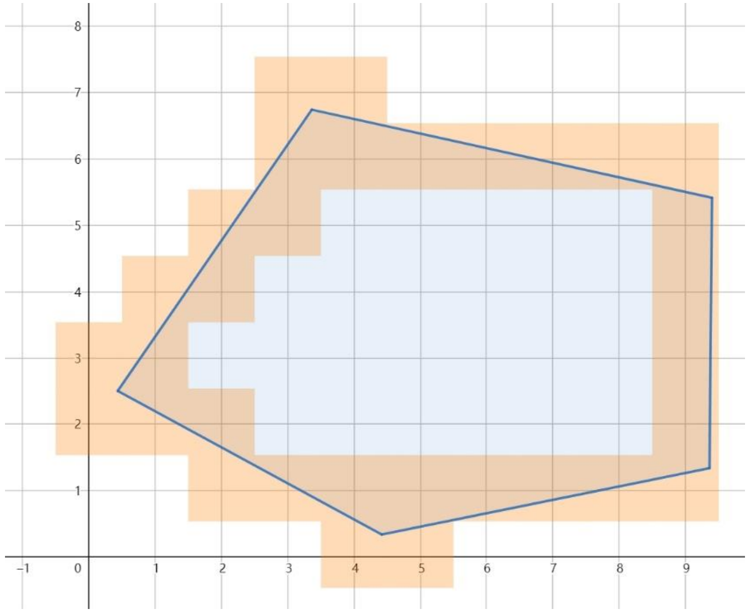


Obtaining a good approximation is impossible in this case.

Preliminaries

- **Definition:** An **Integer-cube** is a unique unit-cube all of whose centers are integer points.
  - Let *C(P)* denote the set of integer-cubes which intersect with *P*.
  - Similarly, we define *C(B(P))*.

$C(P)$ = orange + grey squares
$C(B(P))$ = orange squares



Map each integer-cube to a integer point.

Integer point ↔ Integer-cube

Count ↔ Volume

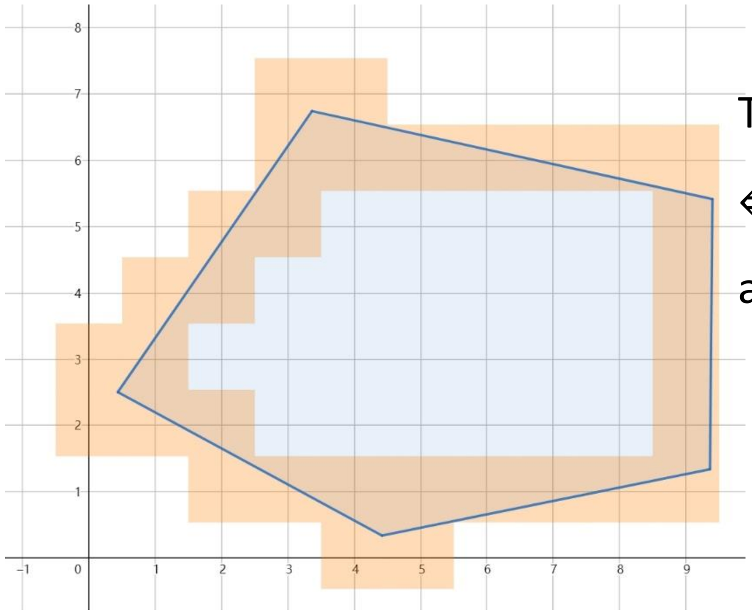
Previous Results

How to compute?

Theorem 1. $\mathrm{vol}(P) - \left|C\left(B(P)\right)\right| \le \mathrm{lat}(P) \le \mathrm{vol}(P) + \left|C\left(B(P)\right)\right|.$

- **Intuition:** The difference between $\mathrm{vol}(P)$ and $\mathrm{l}at(P)$ is caused by points (integer-cubes) that are close to the boundary of *P*.

Orange area is the key



Theorem 1 $\Leftrightarrow |\mathrm{vol}(P) - \mathrm{lat}(P)| \le \left|C\left(B(P)\right)\right|$

$\Leftrightarrow |C(P)| - |C(B(P)| \le \mathrm{lat}(P) \le |C(P)|$

and $|C(P)| - |C(B(P)| \le \mathrm{vol}(P) \le |C(P)|$

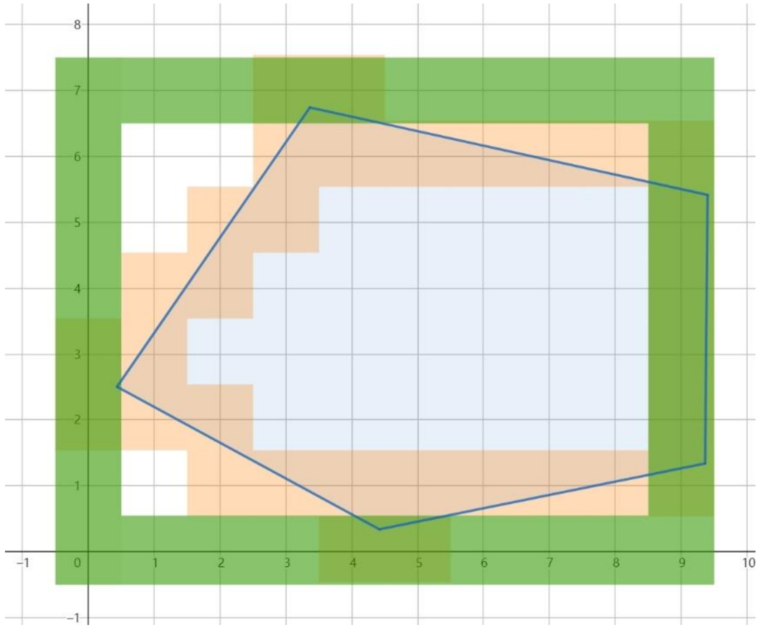
Previous Results

Theorem 2. $\left|C\left(B(P)\right)\right| \leq 2\sum_{i=1}^{n}\prod_{i\neq j}\left(M_j(P) - m_j(P)\right),$

where $m_i(P) = \lceil \min\{x_i | \boldsymbol{x} \in P\} - 1 \rceil,\ \ M_i(P) = \lfloor \max\{x_i | \boldsymbol{x} \in P\} + 1 \rfloor.$

Too loose

Intuition: Mapping orange squares to green squares (cubes close to the surfaces of an outside cuboid).



Contributions

- **A novel algorithm for approximating $\left|\boldsymbol{C}\left(\boldsymbol{B}(\boldsymbol{P})\right)\right|$.**
  - The approximation will be closer to $\left|C\left(B(P)\right)\right|$ than the previous method.
  - The guarantee of the approximation is provided and proved.

- **Extend Theorem 1 to the mixed-integer cases.**

Outline

- Background
- **Our Approach**
  - **The Framework of Bounds Approximation**
  - Sampling Method
  - Extend Bounds to Mixed-Integer Cases
- Evaluation

The Framework of Bounds Approximation

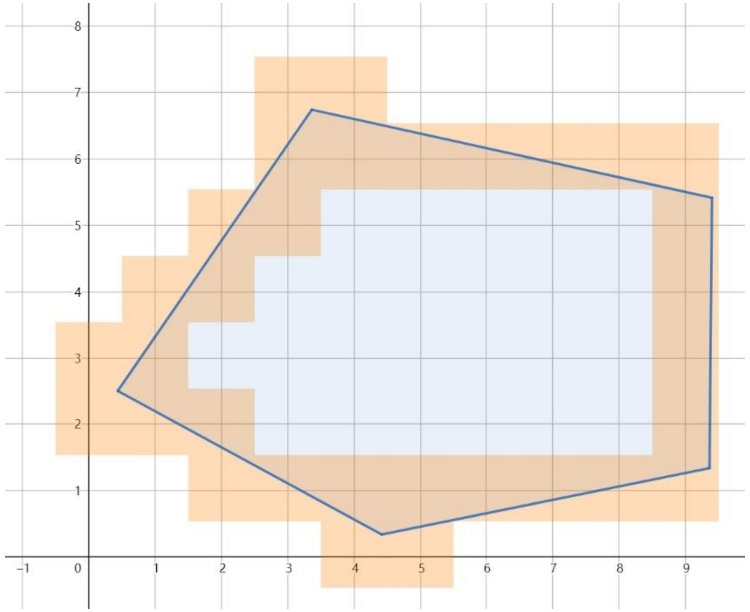
The sampling method will be discussed later

$$\bigcup_{\kappa \in C(P)} \kappa$$

- Generate a set of real sample points uniformly in $\mathrm{union}(C(P))$.
- Count the number of samples that lie in P and $\mathrm{union}(C(B(P))$, i.e., $|X \cap P|$ and $|X \cap \mathrm{union}(C(B(P))|$.

Let X denote the set of sample points

$C(P)$ = orange + grey squares
$C(B(P))$ = orange squares



- $\frac{|C(P)|}{\mathrm{vol}(P)} \approx \frac{|X|}{|X \cap P|}$
- $\frac{|C(B(P))|}{\mathrm{vol}(P)} \approx \frac{|X \cap \mathrm{union}(C(B(P))|}{|X \cap P|}$

The Framework of Bounds Approximation

- Let $\hat{r}_1 = \frac{|X|}{|X \cap P|}$ and $\hat{r}_2 = 1 - \frac{|X \cap \text{union}(C(BP))|}{|X|}$.

- $\frac{|C(P)|}{\text{vol}(P)} \approx \frac{|X|}{|X \cap P|}$
- $\frac{|C(B(P))|}{\text{vol}(P)} \approx \frac{|X \cap \text{union}(C(B(P))|}{|X \cap P|}$

Theorem 3. Given a set of sample points X in $\text{union}(C(P))$. Then

$$|C(P)| - \left|C\big(B(P)\big)\right| = \text{vol}(P) \cdot \lim_{|X| \to \infty} \frac{\hat{r}_2}{\hat{r}_1}$$

Lower bound $\text{lb}(P)$ of $\text{lat}(P)$

and

$$|C(P)| = \text{vol}(P) \cdot \lim_{|X| \to \infty} \frac{1}{\hat{r}_1}$$

Upper bound $\text{ub}(P)$ of $\text{lat}(P)$

The stopping criterion

- Since sampling points is a Bernoulli trial, then $\hat{r}_1$ and $\hat{r}_2$ are approximations of proportion of binomial distributions.

The well-known normal approximation confidence interval.

- According to the binomial confidence interval, we know that r_1 lies in interval $\left[\hat{r}_1 - z_{1-\frac{\delta}{4}}\sqrt{\frac{\hat{r}_1(1-\hat{r}_1)}{|X|}}, \hat{r}_1 + z_{1-\frac{\delta}{4}}\sqrt{\frac{\hat{r}_1(1-\hat{r}_1)}{|X|}}\right]$ with probability at least $1-\delta/2$.

Let $\hat{e}_1 = z_{1-\frac{\delta}{4}}\sqrt{\frac{\hat{r}_1(1-\hat{r}_1)}{|X|}}$ and $\hat{e}_2 = z_{1-\frac{\delta}{4}}\sqrt{\frac{\hat{r}_2(1-\hat{r}_2)}{|X|}}$.

The stopping criterion

Recall that $\hat{e}_1 = z_{1-\frac{\delta}{4}}\sqrt{\frac{\hat{r}_1(1-\hat{r}_1)}{|X|}}$ and $\hat{e}_2 = z_{1-\frac{\delta}{4}}\sqrt{\frac{\hat{r}_2(1-\hat{r}_2)}{|X|}}$.

- r_1 and r_2 lie in interval $[\hat{r}_1 - \hat{e}_1, \hat{r}_1 + \hat{e}_1]$ and $[\hat{r}_2 - \hat{e}_2, \hat{r}_2 + \hat{e}_2]$ with probability at least $1 - \delta/2$, respectively.
- Then $r_1 \geq \hat{r}_1 - \hat{e}_1$ and $r_2 \leq \hat{r}_2 + \hat{e}_2$ with probability at least $1 - \delta/4$.
- Thus $\frac{r_2}{r_1} \leq \frac{\hat{r}_2+\hat{e}_2}{\hat{r}_1-\hat{e}_1}$ with probability at least $1 - \delta/2$.
- Finally, $\frac{\hat{r}_2-\hat{e}_2}{\hat{r}_1+\hat{e}_1} \leq \frac{r_2}{r_1} \leq \frac{\hat{r}_2+\hat{e}_2}{\hat{r}_1-\hat{e}_1}$ and $\frac{1}{\hat{r}_1+\hat{e}_1} \leq \frac{1}{r_1} \leq \frac{1}{\hat{r}_1-\hat{e}_1}$ with probability $1 - \delta$.

The stopping criterion

Theorem 4. If $|X| \geq z^2_{1-\frac{\delta}{4}} \cdot (\frac{1}{\varepsilon} \cdot \sqrt{\frac{1-r_2}{r_2}} + \frac{1-\varepsilon}{\varepsilon} \cdot \sqrt{\frac{1-r_1}{r_1}})^2$ and $|X| \geq z^2_{1-\frac{\delta}{4}} \cdot (\frac{1+\varepsilon}{\varepsilon})^2 \cdot \frac{1-r_1}{r_1}$, then $\mathrm{Prob}(|\frac{\hat{r}_2}{\hat{r}_1} - \frac{r_2}{r_1}| \leq \varepsilon \cdot \frac{r_2}{\hat{r}_1}) \geq 1-\delta$ and $\mathrm{Prob}(|\frac{1}{\hat{r}_1} - \frac{1}{r_1}| \leq \varepsilon \cdot \frac{1}{\hat{r}_1}) \geq 1-\delta$.

Outline

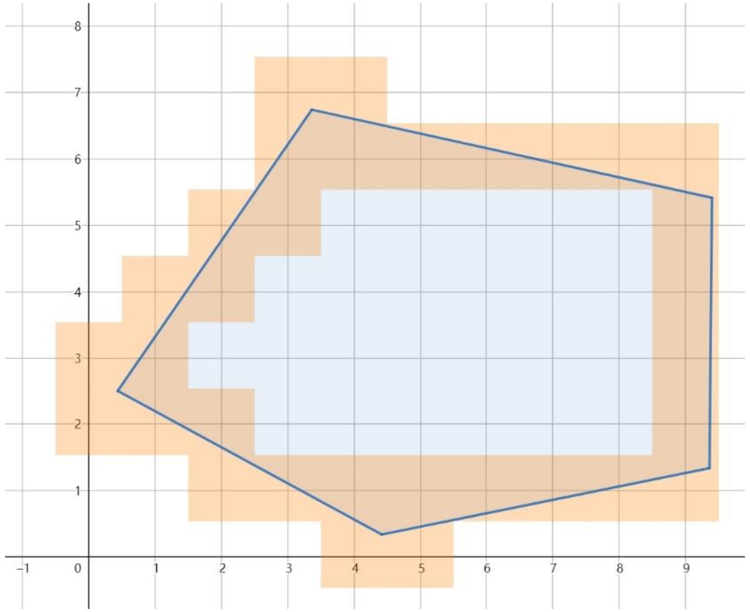
- Background
- **Our Approach**
  - The Framework of Bounds Approximation
  - **Sampling Method**
  - Extend Bounds to Mixed-Integer Cases
- Evaluation

Sampling

Q: How to generate sample points nearly uniformly in $\mathrm{union}(C(P))$?

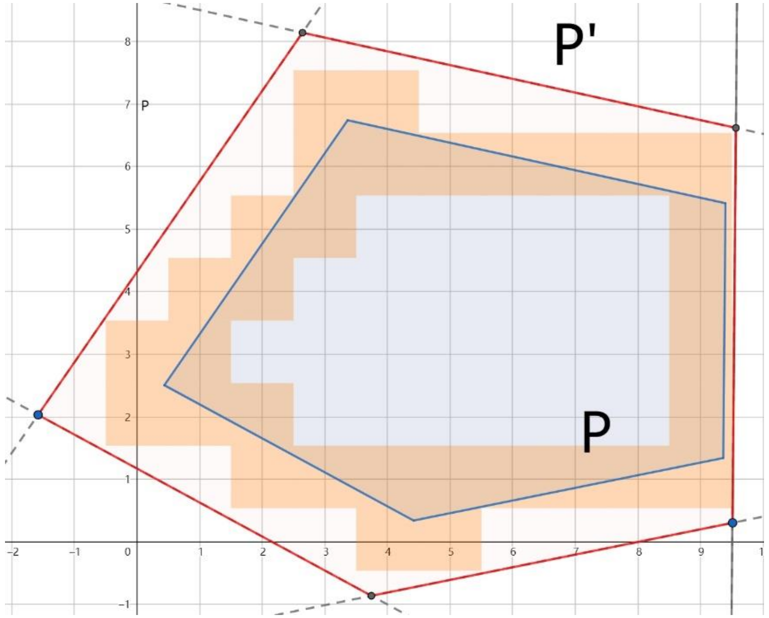
A: Rejection sampling + Nearly uniform sampler on polytopes.

Sampling in orange and grey area (**non-convex**).



Sampling

- Step 1: Find a larger polytope P' s.t. $\mathrm{union}(C(P)) \subset P'$.
- Step 2: Generate points in P' nearly uniformly.
- Step 3: Reject those points outside $\mathrm{union}(C(P))$.

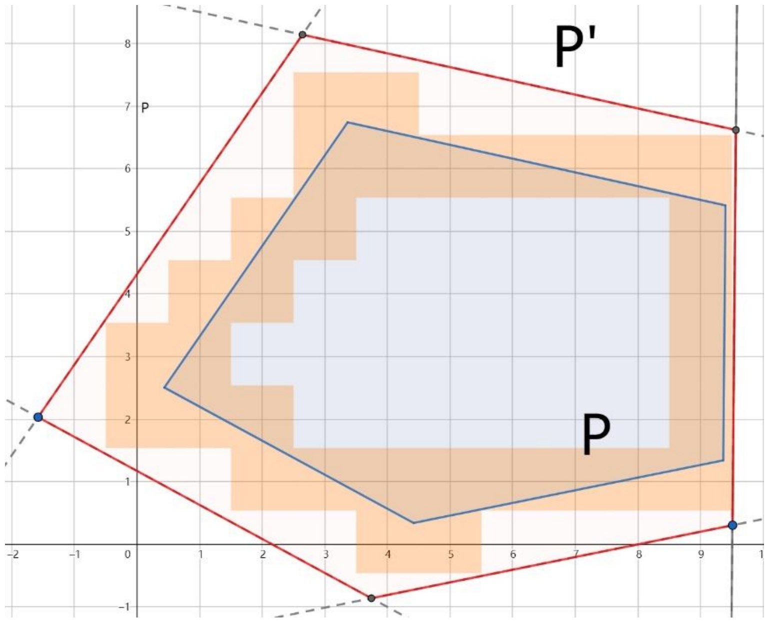


Sampling

- Step 1: Find a larger polytope P' s.t. $\mathrm{union}(C(P)) \subset P'$.
- Shift surfaces of P to obtain P', and P' should as small as possible.

The larger P', the lower probability of acceptance (lower performance).

Finding such parallel surfaces is an optimization problem.

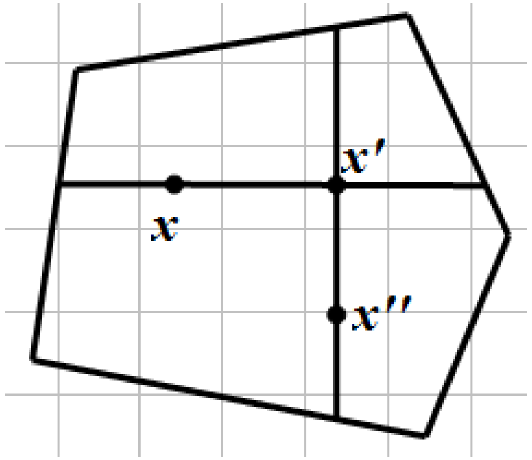


Let $v_k = \max\{\vec{A}_k\vec{x} \mid -1 \le x_i \le 1, i = 1, \ldots, n\}$. Then $\vec{A}_k\vec{x} \le b_k + v_k$ is the new LC of P'.

P' generated in this way should contain $\mathrm{union}\big(C(P)\big)$.

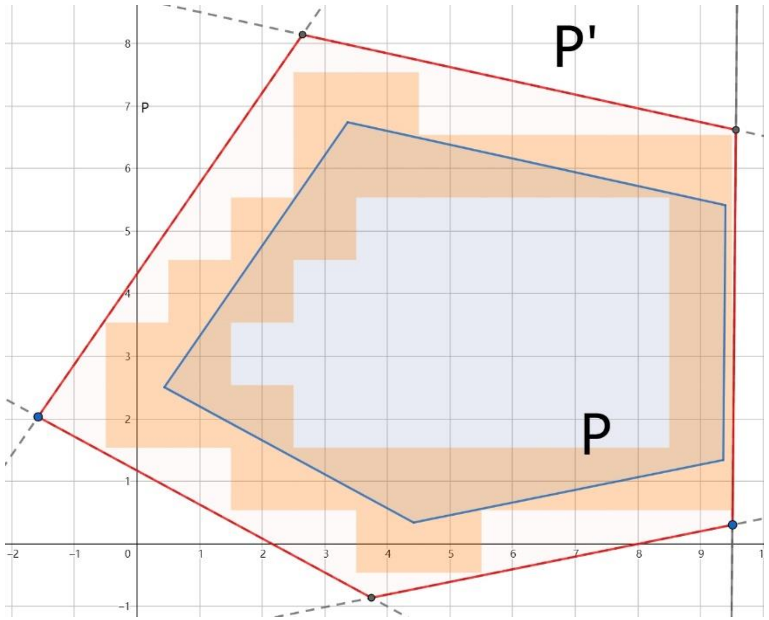
Sampling

- Step 2: Generate points in P' nearly uniformly.
- Coordinate Directions Hit-and-run method
  - Limiting distribution is uniform.
  - Commonly used in approximating polytopes volume.



Sampling

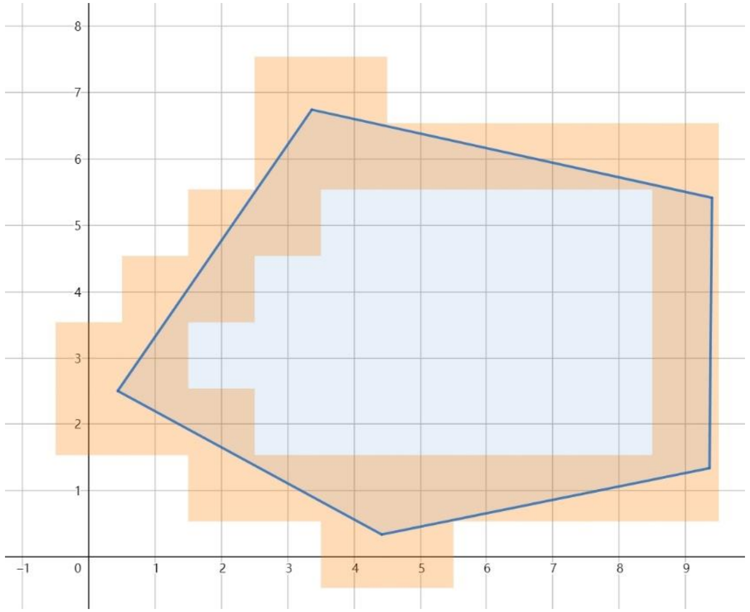
- Step 3: Reject those points outside $\mathrm{union}(C(P))$.
- How to efficiently check whether a point in $\mathrm{union}(C(P))$?



The Framework of Bounds Approximation

- Generate a set of real sample points uniformly in $\text{union}(C(P))$.
- Count the number of samples that lie in P and $\text{union}(C(B(P))$ to approximate $|C(P)|$ and$|C(P)| - |C(B(P))|$ eventually.

- $\dfrac{|C(P)|}{\text{vol}(P)} \approx \dfrac{|X|}{|X \cap P|}$
- $\dfrac{|C(B(P))|}{\text{vol}(P)} \approx \dfrac{|X \cap \text{union}(C(B(P))|}{|X \cap P|}$



Recall that by Theorem 1,

$$|C(P)| - |C(B(P)| \leq \text{lat}(P) \leq |C(P)|,$$

$$|C(P)| - |C(B(P)| \leq \text{vol}(P) \leq |C(P)|.$$

Outline

- Background
- **Our Approach**
  - The Framework of Bounds Approximation
  - Sampling Method
  - **Extend Bounds to Mixed-Integer Cases**
- Evaluation

Extend Bounds to Mixed-Integer Cases

- Mixed-Integer Linear Constraints (MILC) are constraints whose variables include not only reals but also integers.

- Without loss of generality, a set of MILCs F can be written in the form $A\vec{x} = A_1\vec{x}_I + A_2\vec{x}_R \leq \vec{b}$.

Let $n_I = |\vec{x}_I|$ and $n_R = |\vec{x}_R|$. Obviously, $n = n_I + n_R$.

Extend Bounds to Mixed-Integer Cases

- Let $\text{integral}(F)$ denote the integral on $\mathcal{M}(F)$. In detail

$$\text{integral}(F) = \sum_{\vec{\alpha}_I \in \mathcal{M}_I(F)} \text{vol}(F\{\vec{x}_I = \vec{\alpha}_I\}).$$

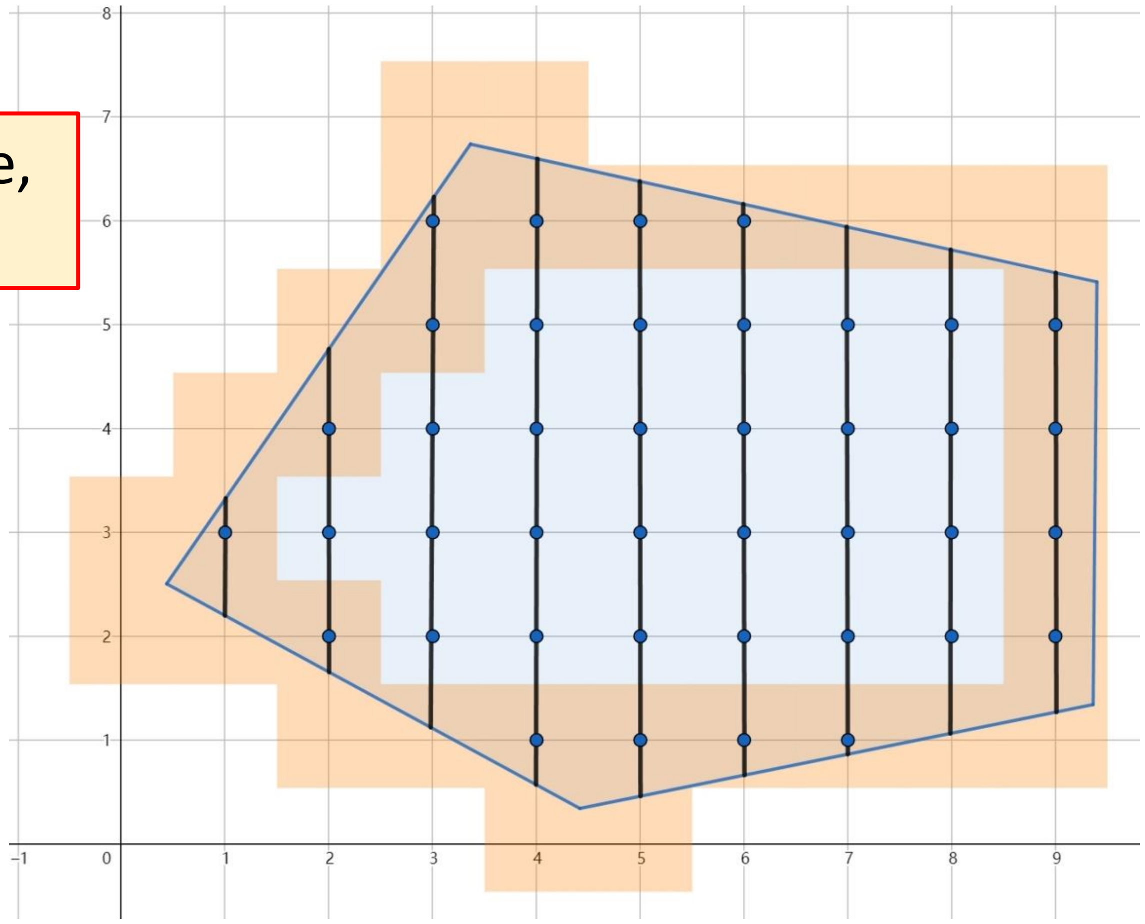
Let $\mathcal{M}(F) = \{\vec{x} \in \mathbb{Z}^{n_I} \times \mathbb{R}^{n_R}: A_1\vec{x}_I + A_2\vec{x}_R \leq \vec{b}\}$ denote the solution space of F.

Let $\mathcal{M}_I(F) \subset \mathbb{Z}^{n_I}$ and $\mathcal{M}_R(F) \subset \mathbb{R}^{n_R}$ denote the projection from $\mathcal{M}(F)$ to variables over $\vec{x}_I$ and $\vec{x}_R$ respectively.

Let $F\{\vec{x}_I = \vec{\alpha}_I\}$ denote the remaining constraints of F by assigning $\vec{\alpha}_I$ to $\vec{x}_I$.

Extend Bounds to Mixed-Integer Cases

- x is an integer variable,
- y is a real variable.



integral(F) is the sum of lengths of black lines parallel to y-axis.

Extend Bounds to Mixed-Integer Cases

Theorem 5. $|C(P)| - |C(B(P)| \leq \mathrm{integral}(F) \leq |C(P)|$.

It indicates that our algorithm can be directly applied for solving the mixed-integer cases, i.e., $\mathrm{lb}(P) \leq \mathrm{integral}(F) \leq \mathrm{ub}(P)$.

Outline

- Background
- Our Approach
  - The Framework of Bounds Approximation
  - Sampling Method
  - Extend Bounds to Mixed-Integer Cases
- **Evaluation**

Benchmarks

- **Random Polytopes**
  - Contains randomly number of integer and real variables.
  - The details of generating random polytopes can be found in our paper.
- **Instances from program analysis**
  - We adopted the application benchmarks from previous works.
  - They were generated by analyzing 7 programs (‘cubature’, ‘gjk’, ‘http-parser’, ‘muFFT’, ‘SimpleXML’, ‘tcas’ and ‘timeout’) ranging from 0.4k to 7.7k lines of source code via a symbolic execution bug-finding tool.
  - There are 3803 SMT(LIA) (linear integer arithmetic) formulas in total.

Experimental results on random polytopes (mixed-integer cases)

Benchmarks			ExactMI		Vinci		PolyVest		Bounds by MixIntCount			
n	n_I	$\bar{S}$	integral(F)	t (s)	vol(P)	t (s)	$\hat{\text{vol}}(P)$	t (s)	$\frac{\text{ub}(P)}{\text{vol}(P)}$	$\frac{\text{lb}(P)}{\text{vol}(P)}$	$\lvert X \rvert$	t (s)
4	1	948	1.44E+10	4.99	1.43E+10	0.007	1.45E+10	0.072	1.055	0.914	281	0.004
	2	874	6.07E+08	508	6.07E+08	0.006	6.15E+08	0.074	1.172	0.820	1177	0.006
	3	1099	—	—	1.47E+11	0.006	1.52E+11	0.067	1.037	0.919	156	0.003
5	1	369	1.64E+09	1.89	1.64E+09	0.012	1.66E+09	0.257	1.479	0.635	2791	0.011
	2	945	—	—	8.70E+12	0.019	8.59E+12	0.256	1.050	0.935	189	0.005
	3	1086	—	—	6.13E+13	0.018	6.17E+13	0.249	1.062	0.923	291	0.005
6	1	715	5.55E+09	18.3	5.55E+09	0.053	5.48E+09	0.755	2.653	0.281	5243	0.023
	2	1438	—	—	1.34E+17	0.067	1.39E+17	0.780	1.034	0.925	113	0.005
	3	229	—	—	5.91E+09	0.063	5.92E+09	0.751	1.795	0.464	3824	0.019
7	1	1101	2.47E+17	113	2.47E+17	0.659	2.50E+17	1.90	1.125	0.848	810	0.008
	2	980	—	—	2.41E+17	0.655	2.44E+17	2.05	1.136	0.867	855	0.010
	3	727	—	—	6.19E+15	0.658	6.13E+15	2.04	1.171	0.813	1180	0.011
8	1	703	6.57E+14	615	6.57E+14	7.64	6.44E+14	4.56	2.036	0.415	4643	0.035
	2	1251	—	—	6.83E+20	7.91	6.83E+20	4.17	1.099	0.873	628	0.009
	3	556	[illegible]	—	5.71E+15	8.24	5.68E+15	4.62	1.695	0.527	3706	0.030
9	1	475	—	—	—	—	2.37E+13	9.05	[illegible]	[illegible]	[illegible]	[illegible]
	2	1250	—	—	—	—	3.82E+22	8.46	[illegible]	[illegible]	[illegible]	[illegible]
	3	138	—	—	[illegible]	[illegible]	1.47E+10	9.71	[illegible]	[illegible]	[illegible]	[illegible]
12	1	252	—	—	[illegible]	[illegible]	2.56E+15	57.4	[illegible]	[illegible]	[illegible]	[illegible]
	2	439	—	—	[illegible]	[illegible]	8.18E+19	54.9	7.063	0.070	45424	0.700
	3	1408	—	—	—	—	1.43E+32	52.3	1.123	0.845	842	0.025
15	1	958	—	—	—	—	5.85E+32	235	2.120	0.408	5102	0.157
	2	1598	—	—	—	—	5.10E+38	228	1.292	0.766	1950	0.076
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	224	1.469	0.635	2902	0.105

Baseline

Volume computation and approximation

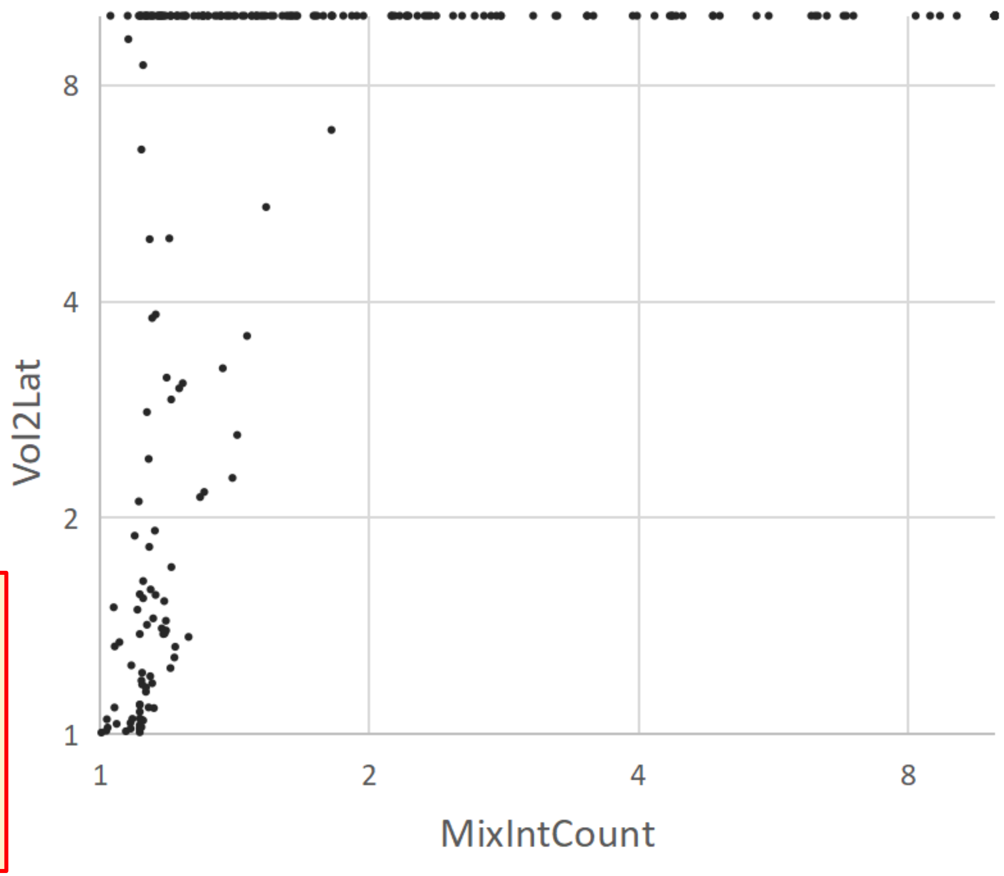
Approximate lb(P) and ub(P), s.t., lb$(P) \leq$ integral$(F) \leq$ ub(P), based on vol(P)

Benchmarks			ExactMI		Vinci		PolyVest		Bounds by MixIntCount			
n	n_I	$\bar{S}$	integral(F)	t (s)	vol(P)	t (s)	$\hat{\text{vol}}(P)$	t (s)	$\frac{\text{ub}(P)}{\text{vol}(P)}$	$\frac{\text{lb}(P)}{\text{vol}(P)}$	$\lvert X\rvert$	t (s)
4	1	948	1.44E+10	4.99	1.43E+10	0.007	1.45E+10	0.072	1.055	0.914	281	0.004
	2	874	6.07E+08	508	6.07E+08	0.006	6.15E+08	0.074	1.172	0.820	1177	0.006
	3	1099	—	—	1.47E+11	0.006	1.52E+11	0.067	1.037	0.919	156	0.003
5	1	369	1.64E+09	1.89	1.64E+09	0.012	1.66E+09	0.257	1.479	0.635	2791	0.011
	2	945	—	—	8.70E+12	0.019	8.59E+12	0.256	1.050	0.935	189	0.005
	3	1086	—	—	6.13E+13	0.018	6.17E+13	0.249	1.062	0.923	291	0.005
6	1	715	5.55E+09	18.3	5.55E+09	0.053	5.48E+09	0.755	2.653	0.281	5243	0.023
	2	1438	—	—	1.34E+17	0.067	1.39E+17	0.780	1.034	0.925	113	0.005
	3	229	—	—	5.91E+09	0.063	5.92E+09	0.751	1.795	0.464	3824	0.019
7	1	1101	2.47E+17	113	2.47E+17	0.659	2.50E+17	1.90	1.125	0.848	810	0.008
	2	980	—	—	2.41E+17	0.655	2.44E+17	2.05	1.136	0.867	855	0.010
	3	727	—	—	6.19E+15	0.658	6.13E+15	2.04	1.171	0.813	1180	0.011
8	1	703	6.57E+14	615	6.57E+14	7.64	6.44E+14	4.56	2.036	0.415	4643	0.035
	2	1251	—	—	6.83E+20	7.91	6.83E+20	4.17	1.099	0.873	628	0.009
	3	556	—	—	5.71E+15	8.24	5.68E+15	4.62	1.695	0.527	3706	0.030
9	1	475	—	—	—	—	2.37E+13	9.05	7.560	0.064	49295	0.390
	2	1250	—	—	—	—	3.82E+22	8.46	1.152	0.836	1016	0.015
							1.47E+10	9.71	34.50	0.000	100000	0.792
							2.56E+15	57.4	43.24	0.001	100000	1.517
							8.18E+19	54.9	7.063	0.070	45424	0.700
							1.43E+32	52.3	1.123	0.845	842	0.025
15	1	958	—	—	—	—	5.85E+32	235	2.120	0.408	5102	0.157
	2	1598	—	—	—	—	5.10E+38	228	1.292	0.766	1950	0.076
								224	1.469	0.635	2902	0.105

- Bounds are mostly useful.
- The overhead is negligible compared to the cost of volume computation or approximation.

Experimental results on random polytopes (mixed-integer cases)

Experimental results on application instances (pure integer cases)



Comparison about the tightness of bounds $\frac{\mathrm{ub}(P)}{\mathrm{lb}(P)}$ (the smaller the better).

Thanks!

Homepage: gecunjing.github.io

E-mail: gecunjing@nju.edu.cn

Tools and Benchmarks: https://github.com/bearben/MixIntCount