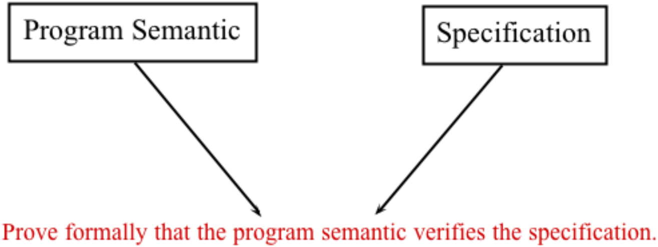
Polyhedron Over-approximation for Complexity Reduction in Static Analysis

Yassamine Seladji[1] and Zheng Qu[2].

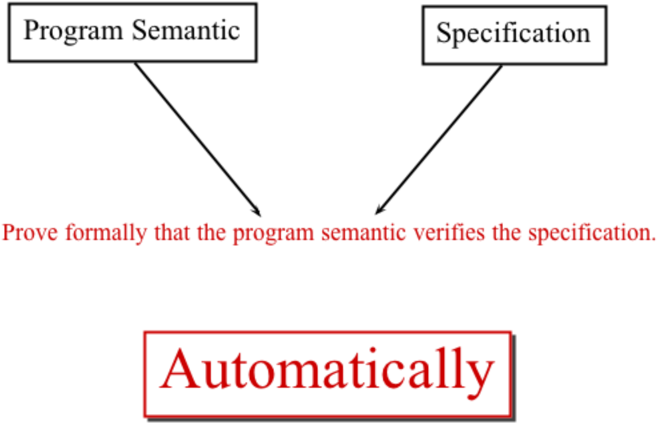
1 Department of Computer Science, The University of Tlemcen.
2 Department of Mathematics, The University of Hong Kong.

The context

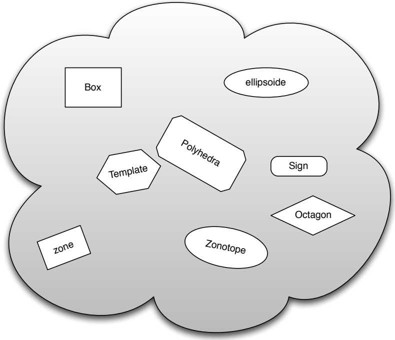
Program verification



The context
Program verification

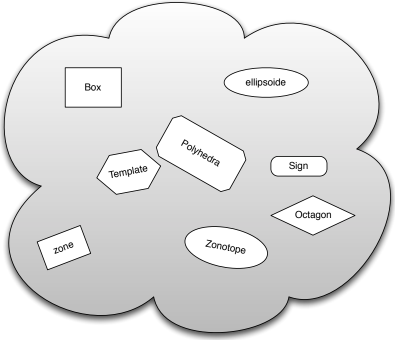


The Context
Program verification



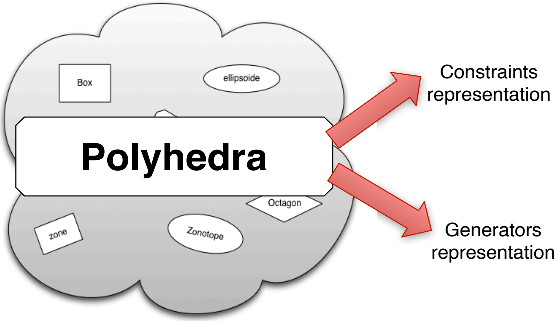
The Context

Program verification



The Context

Program verification



The Context

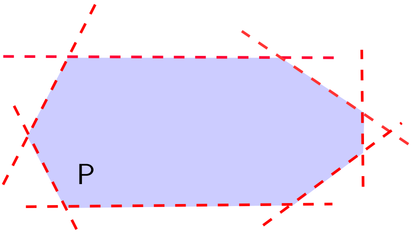
- The constraints representation :

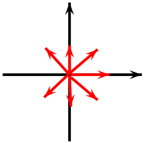
$$\mathbb{P}^n = \bigcap_{i \in [1,n]} \{x \in \mathbb{R}^n : \langle a_i, x \rangle \leq b_i\}$$

The Context

- The constraints representation :

$$\mathbb{P}^n = \bigcap_{i \in [1,n]} \{x \in \mathbb{R}^n : \langle a_i, x \rangle \leq b_i\}$$

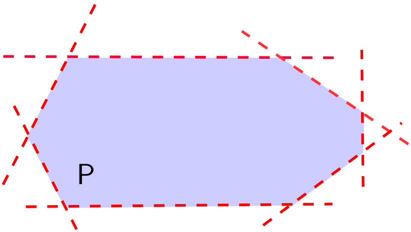


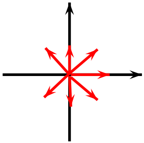


The Context

- The constraints representation :

$$\mathbb{P}^n = \bigcap_{i \in [1,n]} \{x \in \mathbb{R}^n : \langle a_i, x \rangle \leq b_i\}$$





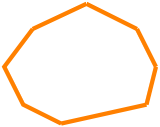
The redundancies elimination process can be time consuming.

The problem formulation

We want to approximate $\mathbb{P}^n$ by keeping only $k \leq n$ constraints, such that : $\mathbb{P}^n \subseteq \mathbb{P}^k$.

The problem formulation

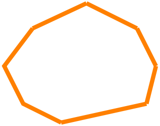
We want to approximate $\mathbb{P}^n$ by keeping only $k \leq n$ constraints, such that : $\mathbb{P}^n \subseteq \mathbb{P}^k$.



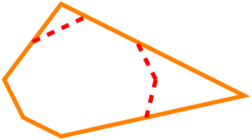
$$P^n := \bigcap_{i=1}^{n} \{x \in \mathbb{R}^n : \langle a_i, x \rangle \leq b_i\},$$

The problem formulation

We want to approximate $\mathbb{P}^n$ by keeping only $k \leq n$ constraints, such that : $\mathbb{P}^n \subseteq \mathbb{P}^k$.



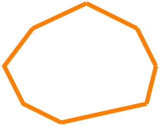
$$P^n := \bigcap_{i=1}^{n} \{x \in \mathbb{R}^n : \langle a_i, x \rangle \leq b_i\},$$



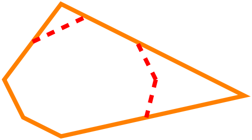
$$P^k := \bigcap_{i=1}^{k} \{x \in \mathbb{R}^n : \langle a_i, x \rangle \leq b_i\},$$

The problem formulation

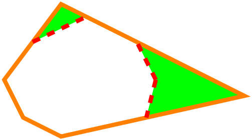
We want to approximate $\mathbb{P}^n$ by keeping only $k \leq n$ constraints, such that : $\mathbb{P}^n \subseteq \mathbb{P}^k$.



$$P^n := \bigcap_{i=1}^{n} \{x \in \mathbb{R}^n : \langle a_i, x \rangle \leq b_i\},$$



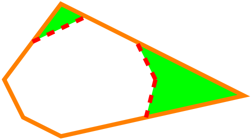
$$P^k := \bigcap_{i=1}^{k} \{x \in \mathbb{R}^n : \langle a_i, x \rangle \leq b_i\},$$



$$vol(P^k) - vol(P^n),$$

The problem formulation

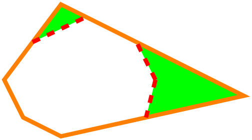
We want to approximate $\mathbb{P}^n$ by keeping only $k \leq n$ constraints, such that : $\mathbb{P}^n \subseteq \mathbb{P}^k$, with $\mathbb{P}^k$ is the best approximation of $\mathbb{P}^n$.



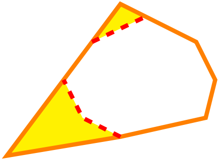
$$vol(P_1^k) - vol(P^n),$$

The problem formulation

We want to approximate $\mathbb{P}^n$ by keeping only $k \leq n$ constraints, such that : $\mathbb{P}^n \subseteq \mathbb{P}^k$, with $\mathbb{P}^k$ is the best approximation of $\mathbb{P}^n$.



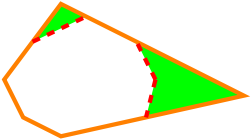
$$vol(P_1^k) - vol(P^n),$$



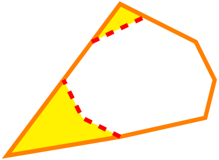
$$vol(P_2^k) - vol(P^n),$$

The problem formulation

We want to approximate $\mathbb{P}^n$ by keeping only $k \leq n$ constraints, such that : $\mathbb{P}^n \subseteq \mathbb{P}^k$, with $\mathbb{P}^k$ is the best approximation of $\mathbb{P}^n$.



$vol(P_1^k) - vol(P^n)$,



$vol(P_2^k) - vol(P^n)$,

If $vol(P_1^k) \leq vol(P_2^k)$ then (P_1^k) is the best approximation of (P^n)

The problem formulation

The best approximation of P^n is the polyhedron with minimum volume associated with a subset with cardinality k.
We need to solve the following combinatorial optimization problem :

$$\min_{\substack{S \subset [n] \\ |S| = k}} vol(P^K) - vol(P^n)$$

The discrete approximation

The following combinatorial optimization problem is known to be a hard problem :

$$\min_{\substack{S \subset [n] \\ |S|=k}} vol(P^K) - vol(P^n)$$

The discrete approximation

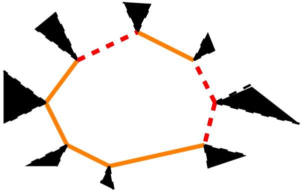
The following combinatorial optimization problem is known to be a hard problem :

$$\min_{\substack{S \subset [n] \\ |S| = k}} vol(P^K) - vol(P^n)$$

The discrete approximation :

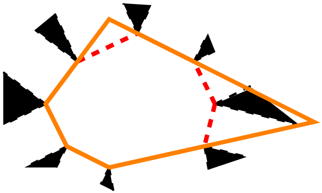
- Define the volume difference approximation.
- Define distance functions.
- Solve the K-median problem.

The volume difference approximation



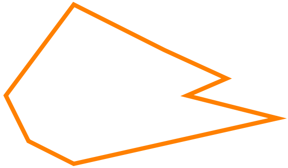
$$N_P{}^n(x) = \left\{ \sum_{i=1}^{n} y_i a_i : y_i \geq 0, y_i(\langle a_i, x \rangle - b_i) = 0 \right\}.$$

The volume difference approximation



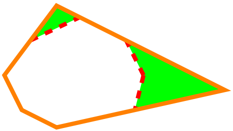
$$P^k \setminus \left(\bigcup_{x \in F_{n-2}(P^n)} \{x + N_P{}^n(x)\} \right),$$

The volume difference approximation

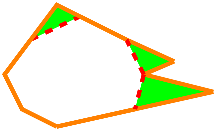


$$O(P^k) = P^k \setminus \left(\bigcup_{x \in F_{n-2}(P^n)} \{x + N_P{}^n(x)\} \right),$$

The volume difference approximation

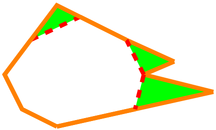


$vol(P^k \backslash P^n)$



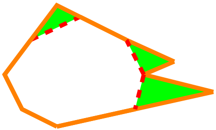
$$vol(O(P^k) \backslash P^n) = \int_{bd(P^n)} \min_{j \in k} d_j(x) dx$$

The volume difference approximation



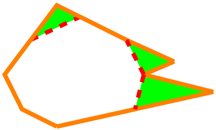
- $vol(O(P^k)\backslash P^n) = \int_{bd(P^n)} \min_{j \in k} d_j(x) dx$

The volume difference approximation



- $vol(O(P^k)\backslash P^n) = \int_{bd(P^n)} \min_{j\in k} d_j(x)dx$
- $$\min_{\substack{S\subset[n]\\|S|=k}} \int_{bd(P^n)} \min_{j\in k} d_j(x)dx$$

The volume difference approximation



- $vol(O(P^k)\backslash P^n) = \int_{bd(P^n)} \min_{j\in k} d_j(x)dx$
- $$\min_{\substack{S\subset[n]\\|S|=k}} \int_{bd(P^n)} \min_{j\in k} d_j(x)dx$$
- approximate

$$\min_{\substack{S\subset[n]\\|S|=k}} vol(P^K) - vol(P^n)$$

The integration approximation

- We generate a discrete set X of representative points.
- X is a subset of points uniformly distributed[1] over $bd(P^n)$.

1. The running Shake and Bake Algorithm

The integration approximation

- We generate a discrete set X of representative points.
- X is a subset of points uniformly distributed [1] over $bd(P^n)$.

$$\sum_{x \in X} \min_{j \in [n]} d_j(x) \simeq \int_{bd(P^n)} \min_{j \in [n]} d_j(x)$$

1. The running Shake and Bake Algorithm

The discrete approximation

The initial combinatorial optimization problem :

$$\min_{\substack{S \subset [n] \\ |S| = k}} vol(P^K) - vol(P^n)$$

The discrete approximation

The initial combinatorial optimization problem :

$$\min_{\substack{S\subset[n]\\|S|=k}} vol(P^K) - vol(P^n)$$

Its discrete approximation problem :

$$\min_{\substack{S\subset[n]\\|S|=k}} \sum_{x\in X} \min_{j\in[n]} d_j(x)$$

The discrete approximation

The initial combinatorial optimization problem :

$$\min_{\substack{S \subset [n] \\ |S|=k}} vol(P^K) - vol(P^n)$$

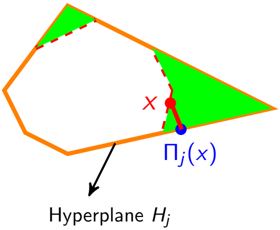
Its discrete approximation problem :

$$\min_{\substack{S \subset [n] \\ |S|=k}} \sum_{x \in X} \min_{j \in [n]} d_j(x)$$

X be a discrete subset of P and $d(\cdot,\cdot) : [m] \times \mathbb{R}^n \to \mathbb{R}_+$ be a distance function.

The distance functions

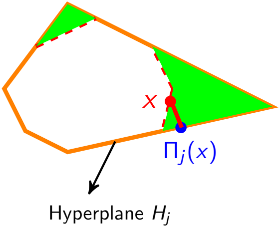
- The projective distance :



$$\Pi_j(x) := \arg\min\{\|x - y\| : y \in H_j\}.$$

The distance functions

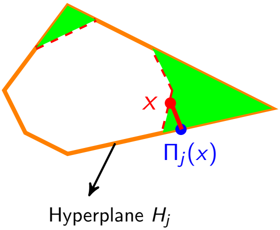
- The projective distance :



$$\Pi_j(x) := \arg\min\{\|x - y\| : y \in H_j\}.$$

$$p_j(x) = b_j - \langle a_j, x \rangle, \;\; j \in [n], x \in bd(P^n).$$

- The projective distance :



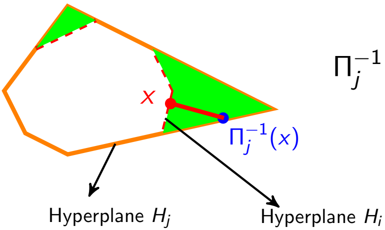
$$\Pi_j(x) := \arg\min\{\|x - y\| : y \in H_j\}.$$

$$p_j(x) = b_j - \langle a_j, x\rangle, \;\; j \in [n], x \in bd(P^n).$$

Where : $bd(P^n) = \bigcup_{i=1}^{n} (P^n \bigcap H_i)$ with $H_i := \{x \in \mathbb{R}^n : \langle a_i, x\rangle = b_i\}$

The distance functions

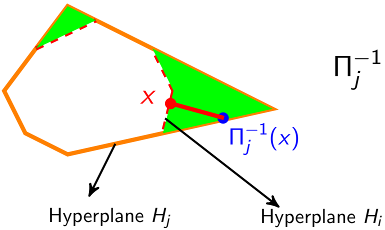
- The inverse projective distance :



$$\Pi_j^{-1}(x) := \{y \in H_j : \Pi_i(y) = x\}.$$

The distance functions

- The inverse projective distance :

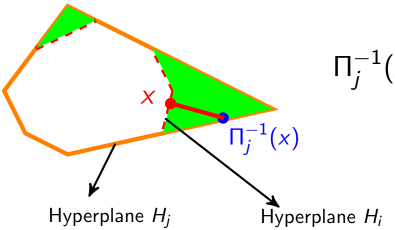


$$\Pi_j^{-1}(x) := \{y \in H_j : \Pi_i(y) = x\}.$$

$$\delta_j(x) := \frac{b_j - \langle a_j, x\rangle}{\max\left(\langle a_i, a_j\rangle, 0\right)}, \;\; j \in [n], x \in ri(bd(P^n)) \cap H_i.$$

The distance functions

- The inverse projective distance :



$$\Pi_j^{-1}(x) := \{y \in H_j : \Pi_i(y) = x\}.$$

$$\delta_j(x) := \frac{b_j - \langle a_j, x\rangle}{\max\left(\langle a_i, a_j\rangle, 0\right)}, \;\; j \in [n], x \in ri(bd(P^n)) \cap H_i.$$

Where :

$$ri(bd(P^n)) = \bigcup_{i=1}^{n} \left(P^n \bigcap H_i \backslash \left(\bigcup_{j \neq i} H_j \right) \right).$$

The discrete approximation

$$\min_{\substack{S \subset [n] \\ |S| = k}} \sum_{x \in X} \min_{j \in [n]} d_j(x)$$

- It's a k-median problem.
- X is the set of cities.
- The hyperplanes $H_1, \cdots, H_m$ is the set facilities .
- $d(j, x)$ is the distance between a city $x \in X$ and a facility H_j.
- We apply the algorithm of Jain and Vazirani [2], for approximately solving k-median problems in polynomial time.

2. V. V. Vazirani, Approximation algorithms, Springer-Verlag, 2001

The experimentation

The experimentation results are based on three criterion :

- the quality criterion used to evaluate the accuracy.
- the efficiency criterion used to evaluate the execution time.
- The impact of the K parameter depends on the number of the kept constraints K .

The experimentation : the quality criterion

The quality criterion evaluates the added over-approximation.

Program		P_N		P_K		
Name	$\|V\|$	N	volume	K	Inverse Projective	Projective
filter2	4	332	35.91	221	39.03	36.62
Dampened_oscillator	4	332	31.78066	221	36.6896	41.74
Harmonic_oscillator	6	332	243.07	221	272.71	300.58
lp_iir_9600_2	6	372	274	248	366	345 (271)
Linear_quadratic_gaussian	7	398	628.14	265	851.149	685.78
Butterworth_low_pass_filter	9	542	3293.69	361	4434.05	4286.32
Observer_based_controller	10	500	Unbounded	333	11389.06	14710.72
lp_iir_9600_4	10	500	Unbounded	333	682	613
lp_iir_9600_4_elliptic	10	500	Unbounded	333	3133	2892
lp_iir_9600_6_elliptic	14	692	Unbounded	461	643387	523225
bs_iir_9600_12000_10_chebyshev	22	1268	Unbounded	845	26264	18650

The experimentation : the quality criterion

The exemple : `filter2`

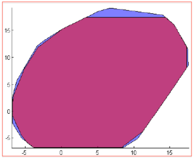


FIGURE – The projective distance

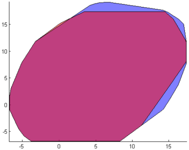


FIGURE – The inverse projective distance

The exemple : `Harmonic_oscillator`

FIGURE – The projective distance

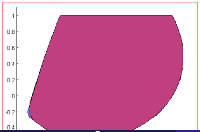
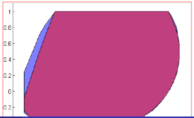


FIGURE – The inverse projective distance



The experimentation : the efficiency criterion

The efficiency criterion used to evaluate the execution time.

Program		Standard analysis		K-analysis		
Name	$\|V\|$	N	time	K	inverse Projective	Projective
filter2	4	332	0.586s	221	5m57s	2m44.108s
Dampened_oscillator	4	332	2.080s	221	5m52s	2m29s
Harmonic_oscillator	6	332	0.610s	221	5m45s	2m24.441s
lp_iir_9600_2	6	372	41.462s	248	6m52.283s	3m8.179s
Linear_quadratic_gaussian	7	398	37m5s	265	8m25s	3m38.310s
Butterworth_low_pass_filter	9	542	124m35s	361	16m30s	7m21.390s
Observer_based_controller	10	500	TO	333	13m46s	6m13.019s
lp_iir_9600_4	10	500	TO	333	13m52s	6m11.377s
lp_iir_9600_4_elliptic	10	500	TO	333	13m37s	6m17.650s
lp_iir_9600_6_elliptic	14	692	TO	461	27m18s	12m5.430s
bs_iir_9600_12000_10_chebyshev	22	1268	TO	845	TO	44m28.396s

The experimentation : the impact of the K parameter

Evaluate the impact of the K parameter using the quality of the efficiency criterion.

Program	lp_iir_9600_2				Butterworth_low_pass_filter			
N	372				542			
K	50	150	248	300	100	200	300	361
volume	2029	450	345	315	1.9×10^5	2.4×10^4	6.4×10^3	4.286×10^3
Execution Time	3m10.6s	3m10.2s	3m8s	3m9s	7m36s	7m32s	7m23s	7m21s

Future work

- How to choose a relevant K taking into account the space dimension.
- What is the impact of the initial number of constraints on the result precision.

Thank you for your attention.

Questions