

Finding Bounds on Ehrhart Quasi-Polynomials

Harald Devos Jan Van Campenhout Dirk Stroobandt

ELIS/PARIS
Ghent University

Seventh ACES Symposium
September 17–18 2007, Edegem

Outline

1 Introduction

- What are (Ehrhart) quasi-polynomials?
- Where do they arise?
- Why do we need bounds on quasi-polynomials?

2 How do we find bounds?

- Continuous versus discrete domain extrema of polynomials
- Converting quasi-polynomials into polynomials

3 Conclusions and Future Work



Outline

1 Introduction

- What are (Ehrhart) quasi-polynomials?
- Where do they arise?
- Why do we need bounds on quasi-polynomials?

2 How do we find bounds?

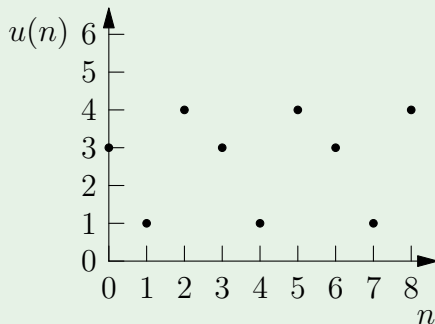
- Continuous versus discrete domain extrema of polynomials
- Converting quasi-polynomials into polynomials

3 Conclusions and Future Work



Periodic Numbers

Example



$$u(n) = [3, 1, 4]_n$$

Periodic Numbers

Definition

Let n be a discrete variable, i.e. $n \in \mathbb{Z}$. A 1-dimensional periodic number is a function that depends periodically on n .

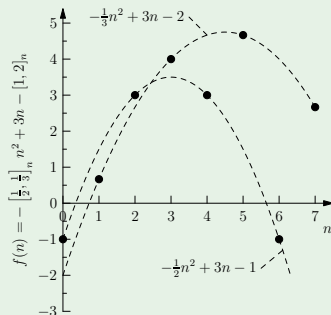
$$u(n) = [u_0, u_1, \dots, u_{d-1}]_n = \begin{cases} u_0 & \text{if } n \equiv 0 \pmod{d} \\ u_1 & \text{if } n \equiv 1 \pmod{d} \\ \vdots & \\ u_{d-1} & \text{if } n \equiv d-1 \pmod{d} \end{cases}$$

d is called the period.

Quasi-Polynomials

Example

$$\begin{aligned} f(n) &= - \left[\frac{1}{2}, \frac{1}{3} \right]_n n^2 + 3n - [1, 2]_n \\ &= \begin{cases} -\frac{1}{3}n^2 + 3n - 2 & \text{if } n \equiv 0 \pmod{2} \\ -\frac{1}{2}n^2 + 3n - 1 & \text{if } n \equiv 1 \pmod{2} \end{cases} \end{aligned}$$



Quasi-Polynomials

Definition

A polynomial in a variable x is a linear combination of powers of x :

$$f(x) = \sum_{i=0}^g c_i x^i$$

Quasi-Polynomials

Definition

A polynomial in a variable x is a linear combination of powers of x :

$$f(x) = \sum_{i=0}^g c_i x^i$$

Definition

A quasi-polynomial in a variable x is a polynomial expression with periodic numbers as coefficients:

$$f(n) = \sum_{i=0}^g u_i(n) n^i$$

with $u_i(n)$ periodic numbers.

Outline

1 Introduction

- What are (Ehrhart) quasi-polynomials?
- Where do they arise?
- Why do we need bounds on quasi-polynomials?

2 How do we find bounds?

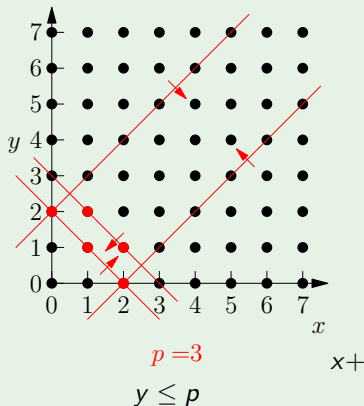
- Continuous versus discrete domain extrema of polynomials
- Converting quasi-polynomials into polynomials

3 Conclusions and Future Work



Where do Quasi-Polynomials arise?

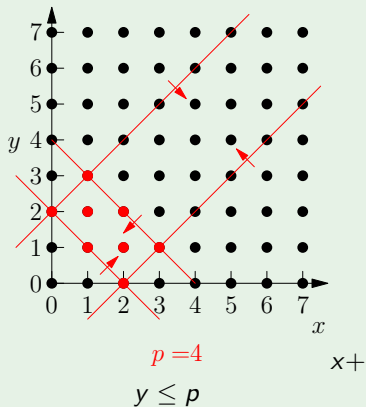
Example



$$\frac{p}{3} \quad \frac{f(p)}{5}$$

Where do Quasi-Polynomials arise?

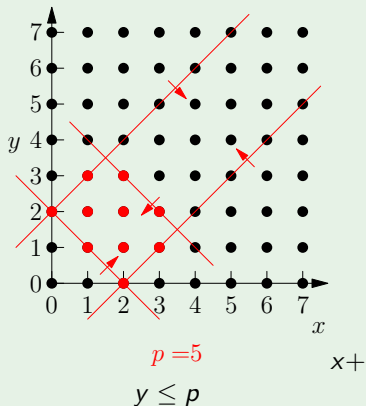
Example



p	$f(p)$
3	5
4	8

Where do Quasi-Polynomials arise?

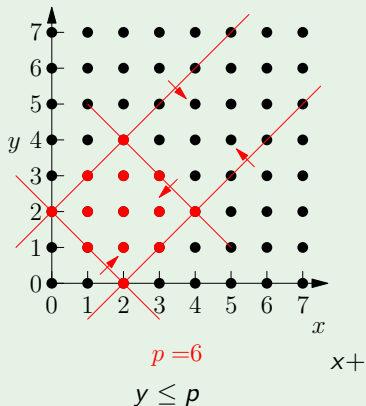
Example



p	$f(p)$
3	5
4	8
5	10

Where do Quasi-Polynomials arise?

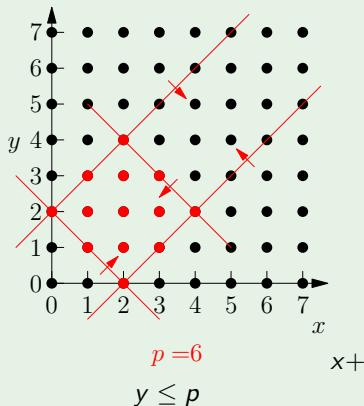
Example



p	$f(p)$
3	5
4	8
5	10
6	13

Where do Quasi-Polynomials arise?

Example



p	$f(p)$
3	5
4	8
5	10
6	13

$$\frac{5}{2}p + \left[-2, \frac{-5}{2} \right]_p$$

Where do Quasi-Polynomials arise?

- The number of integer points in a **parametric polytope** P_p of dimension n is expressed as a piecewise a quasi-polynomial of degree n in p (Claus and Loechner).
- More general **polyhedral counting problems**:
Systems of linear inequalities combined with $\vee, \wedge, \neg, \forall, \text{ or } \exists$ (Presburger formulas).
- Many problems in **static program analysis** can be expressed as polyhedral counting problems.

Outline

1 Introduction

- What are (Ehrhart) quasi-polynomials?
- Where do they arise?
- Why do we need bounds on quasi-polynomials?

2 How do we find bounds?

- Continuous versus discrete domain extrema of polynomials
- Converting quasi-polynomials into polynomials

3 Conclusions and Future Work



Why do we need bounds on quasi-polynomials?

Some problems in static program analysis need bounds on quasi-polynomials.

Example

Number of live elements = quasi-polynomial



Memory usage = maximum over all execution points

Outline

1 Introduction

- What are (Ehrhart) quasi-polynomials?
- Where do they arise?
- Why do we need bounds on quasi-polynomials?

2 How do we find bounds?

- Continuous versus discrete domain extrema of polynomials
- Converting quasi-polynomials into polynomials

3 Conclusions and Future Work



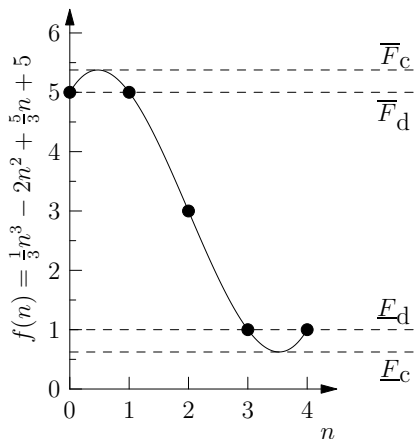
Continuous vs. Discrete domain extrema of polynomials

Discrete domain \Rightarrow evaluate in each point

Not possible for

- parametric domains
- large domains (NP-complete)

Continuous vs. Discrete domain extrema of polynomials



- The relative difference is smaller for
 - ▶ larger intervals
 - ▶ lower degree
- \Rightarrow Continuous-domain extrema can be used as approximation of discrete-domain extrema.

Outline

1 Introduction

- What are (Ehrhart) quasi-polynomials?
- Where do they arise?
- Why do we need bounds on quasi-polynomials?

2 How do we find bounds?

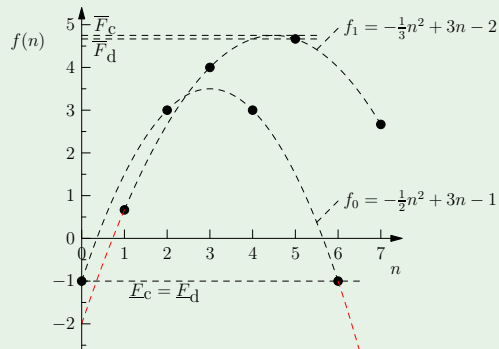
- Continuous versus discrete domain extrema of polynomials
- Converting quasi-polynomials into polynomials

3 Conclusions and Future Work



How: Mod Classes

Example



Good for

- small period
- large domains

How: Other Methods

Finding Bounds on Ehrhart Quasi-Polynomials

Harald Devos, Jan Van Campenhout
and Dik Stroobandt
Ghent University, ELIS/PARIS.



Harald.Devos@UGent.be

Sint-Pietersnieuwestraat 41, 9000 Gent, Belgium

<http://www.elis.ugent.be/paris>

What are Quasi-Polynomials ?

A **quasi-polynomial** (or **quasi-polynomial**) is a function $f: \mathbb{N} \rightarrow \mathbb{R}$ such that $f(x) = \sum_{i=0}^d a_i x^i + b(x)$ where $b(x)$ is a periodic function.

$$f(x) = \sum_{i=0}^d a_i x^i + b(x)$$

$$b(x) = \sum_{i=0}^d a_i x^i + b(x)$$

A **quasi-polynomial** is a polynomial expression with periodic coefficients.

$$f(x) = \sum_{i=0}^d a_i x^i + b(x)$$

For example

$$f(x) = \sum_{i=0}^d a_i x^i + b(x)$$



Fig.1: Quasi-polynomial

Periodic numbers can be expressed using Fourier or trigonometric expressions instead of the unit vector.

$$b(x) = \sum_{i=0}^d a_i x^i + b(x)$$

The example above can be written as:

$$f(x) = \sum_{i=0}^d a_i x^i + b(x)$$

The definition of periodic number and quasi-polynomial can be extended to the multivariate case.

Other methods

- Good evaluation in each point of the domain.
- Not possible for large or parametric domains.
- Bad degree (small number of points) for each Fourier solution because of periodicity (E. Weibel).
- Good for: high dimensional value (off-diagonal) depending on a single variable.



Add Var

Mod Classes

- Techniques (Mod Classes) with constant.
- New variables eliminated.
- Addition modulo range to value of f .
- Good for small periods and large domains.



Fig.2: Mod Classes method.

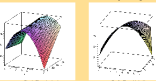


Fig.3: Mod Classes method.

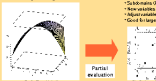


Fig.4: Mod Classes method.

Split Periods

- Subdividing Periods with constant.
- New variables eliminated.
- Algebraic variable range to value of f .
- Good for large periods.



Fig.5: Split Periods method.

Conclusions and Future work

- Different methods perform better in accuracy-complexity trade-off for different situations.
- Influenced by the domain (size, period, degree, number of dimensions and factorial expressions...)
- A hybrid method should be considered.
- Select for each quasi-polynomial in each dimension (quasi-polynomial) independently which method to apply.
- Constructive methods combined with stochastic search with non-convex cost.

Harald Devos is supported by I.W.T. grant 060008

Continuous- vs. discrete-domain extrema of polynomials

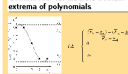


Fig.6: Continuous vs. discrete domain extrema of polynomials.

- Related difference: EE is smaller for larger domains or lower degrees.
- For small domains evaluation at all points is so an little cost and gives the exact answer.
- For large domains the cost of evaluation becomes a good approximation of the cost of the exact answer.
- For multi-variate polynomials, perfect evaluation is for a selection of the variables, if possible.

Experiments

- Memory usage of a matrix multiplication (x200) 32-bit/64-bit.
- Number of nodes.

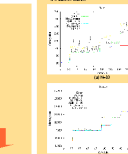


Fig.7: Comparison of methods: number of memory usage for matrix multiplication as a function of the problem.

Other methods

- needed for large periods
- offer trade-off between accuracy and computation time
- see poster

Conclusions and Future Work

- Bounds on quasi-polynomials useful for static program analysis
- Different methods fit different situations (period, degree, domain size).

- Outlook
 - ▶ A hybrid method should be constructed.
 - ▶ Parametric bounds on parameterized quasi-polynomials