Derivation of Process Networks and the use of Bernstein Expansion to compute FIFO Sizes

Sven Verdoolaege

Katholieke Universiteit Leuven, Belgium

July 3, 2009

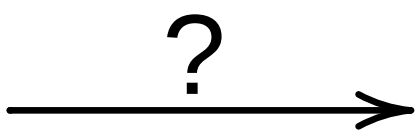
Outline

1. Introduction and Motivation
2. Derivation of Process Networks
   - Polytope Model
   - Dataflow Analysis
   - Parametric Counting
   - Parametric Maximization
3. Bernstein Expansion
   - Interval
   - Polytopes
   - Simplification
   - Incremental Computation
4. Conclusion

Motivation

Goal: map sequential programs to parallel hardware (MPSoC)

```
 1 for( i=0; i<2*Nrw; i++ )
     for( j=0; j<2*Ncl; j++ )
       a[i][j] = ReadImage();

   for( i=0; i<Nrw; i++ ) {
 6   // 1D DWT in vertical direction with subsampling
     for( j=0; j<2*Ncl; j++ ) {
       tmpLine = (i==Nrw-1) ? a[2*i][j] : a[2*i+2][j];
       Hf[j] = high_flt_vert( a[2*i][j], a[2*i+1][j], tmpLine );

11     tmp = (i==0) ? Hf[j] : oldHf[j];
       low_flt_vert( tmp,a[2*i][j],Hf[j],&oldHf[j],&Lf[j] );
     }

     // 1D DWT in horizontal direction with subsampling ---------
16   for( j=0; j<Ncl; j++ ) {
       tmp = (j==Ncl-1) ? Lf[2*j] : Lf[2*j+2];
       HL[i][j] = high_flt_hor( Lf[2*j], Lf[2*j+1], tmp );

       tmp = (j==0) ? HL[i][j] : HL[i][j-1];
21     LL[i][j] = low_flt_hor( tmp, Lf[2*j],  HL[i][j] );
     }

     // 1D DWT in horizontal direction with subsampling ---------
     for( j=0; j<Ncl; j++ ) {
26     tmp = (j==Ncl-1) ? Hf[2*j] : Hf[2*j+2];
       HH[i][j] = high_flt_hor( Hf[2*j], Hf[2*j+1], tmp );

       tmp = (j == 0) ? HH[i][j] : HH[i][j-1];
       LH[i][j] = low_flt_hor( tmp, Hf[2*j],  HH[i][j] );
31   }
   }
   // The Outputs  ------------------------------------------------
   for( i=0; i<Nrw; i++)
     for( j=0; j<Ncl; j++) {
36     Sink( LL[i][j] );
       Sink( HL[i][j] );
       Sink( LH[i][j] );
       Sink( HH[i][j] );
     }
```





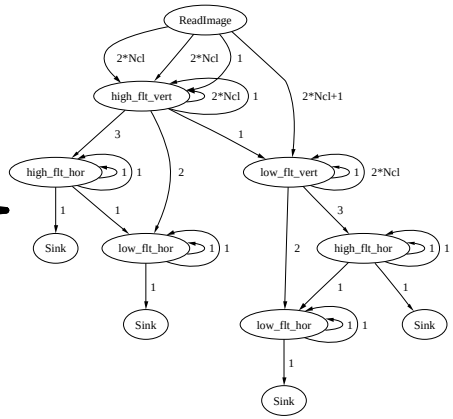
Motivation

Goal: map sequential programs to parallel hardware (MPSoC)

```
 1 for( i=0; i<2*Nrw; i++ )
     for( j=0; j<2*Ncl; j++ )
       a[i][j] = ReadImage();

   for( i=0; i<Nrw; i++ ) {
 6   // 1D DWT in vertical direction with subsampling
     for( j=0; j<2*Ncl; j++ ) {
       tmpLine = (i==Nrw-1) ? a[2*i][j] : a[2*i+2][j];
       Hf[j] = high_flt_vert( a[2*i][j], a[2*i+1][j], tmpLine );

11     tmp = (i==0) ? Hf[j] : oldHf[j];
       low_flt_vert( tmp,a[2*i][j],Hf[j],&oldHf[j],&Lf[j] );
     }

     // 1D DWT in horizontal direction with subsampling ---------
16   for( j=0; j<Ncl; j++ ) {
       tmp = (j==Ncl-1) ? Lf[2*j] : Lf[2*j+2];
       HL[i][j] = high_flt_hor( Lf[2*j], Lf[2*j+1], tmp );

       tmp = (j==0) ? HL[i][j] : HL[i][j-1];
21     LL[i][j] = low_flt_hor( tmp, Lf[2*j],  HL[i][j] );
     }

     // 1D DWT in horizontal direction with subsampling ---------
     for( j=0; j<Ncl; j++ ) {
26     tmp = (j==Ncl-1) ? Hf[2*j] : Hf[2*j+2];
       HH[i][j] = high_flt_hor( Hf[2*j], Hf[2*j+1], tmp );

       tmp = (j == 0) ? HH[i][j] : HH[i][j-1];
       LH[i][j] = low_flt_hor( tmp, Hf[2*j],  HH[i][j] );
31   }
   }
   // The Outputs  ------------------------------------------------
   for( i=0; i<Nrw; i++)
     for( j=0; j<Ncl; j++) {
36     Sink( LL[i][j] );
       Sink( HL[i][j] );
       Sink( LH[i][j] );
       Sink( HH[i][j] );
     }
```

?



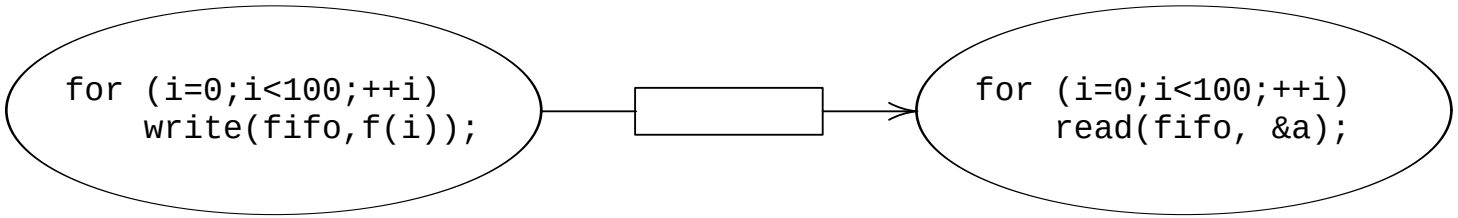


process network

Kahn Process Networks

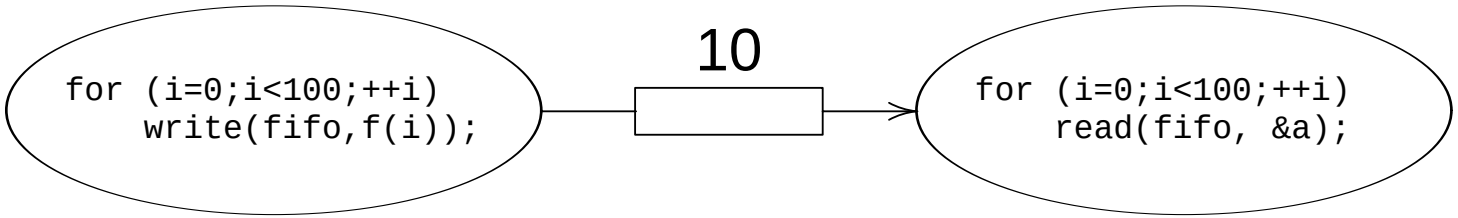
- Parallel specification
  $\Rightarrow$ easier to map onto parallel hardware
- Consists of
  - independent processes
  - communication channels
    $\Rightarrow$ unbounded FIFOs / memories
  - synchronization: blocking read + blocking write

Kahn Process Networks



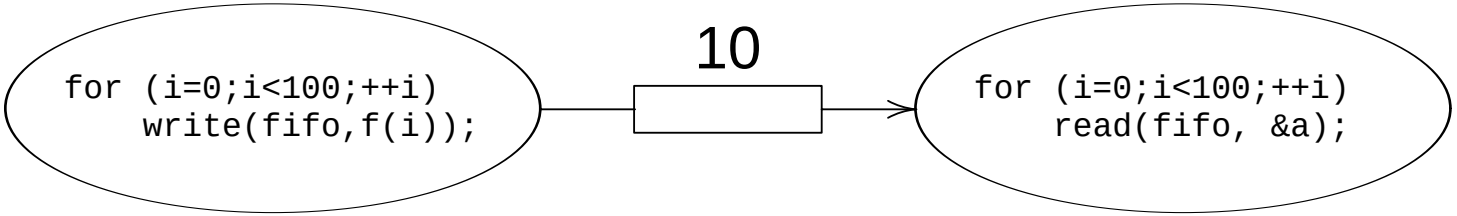
- Parallel specification
  $\Rightarrow$ easier to map onto parallel hardware
- Consists of
  - independent processes
  - communication channels
    $\Rightarrow$ unbounded FIFOs / memories
  - synchronization: blocking read + blocking write

Kahn Process Networks



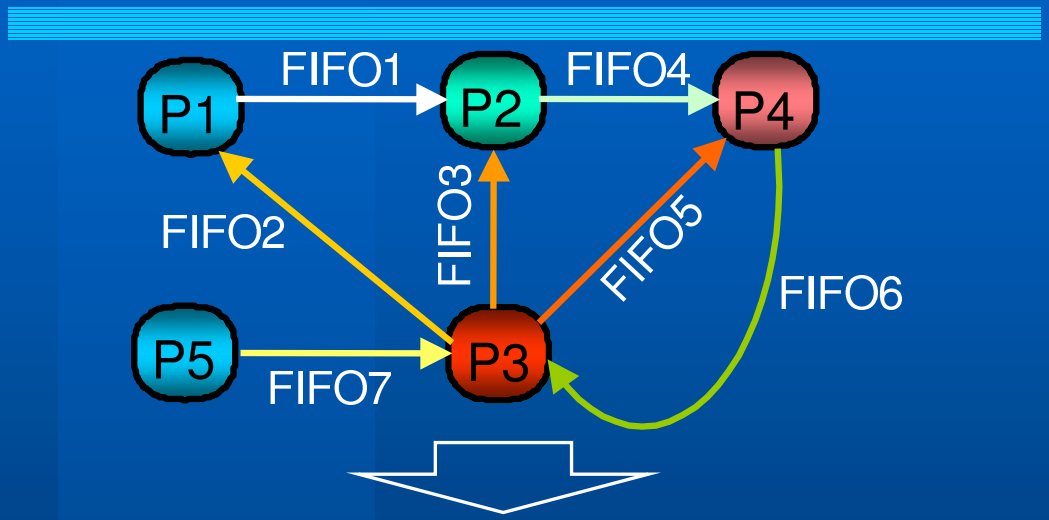
- Parallel specification
  $\Rightarrow$ easier to map onto parallel hardware
- Consists of
  - independent processes
  - communication channels
    $\Rightarrow$ unbounded FIFOs / memories
  - synchronization: blocking read + blocking write

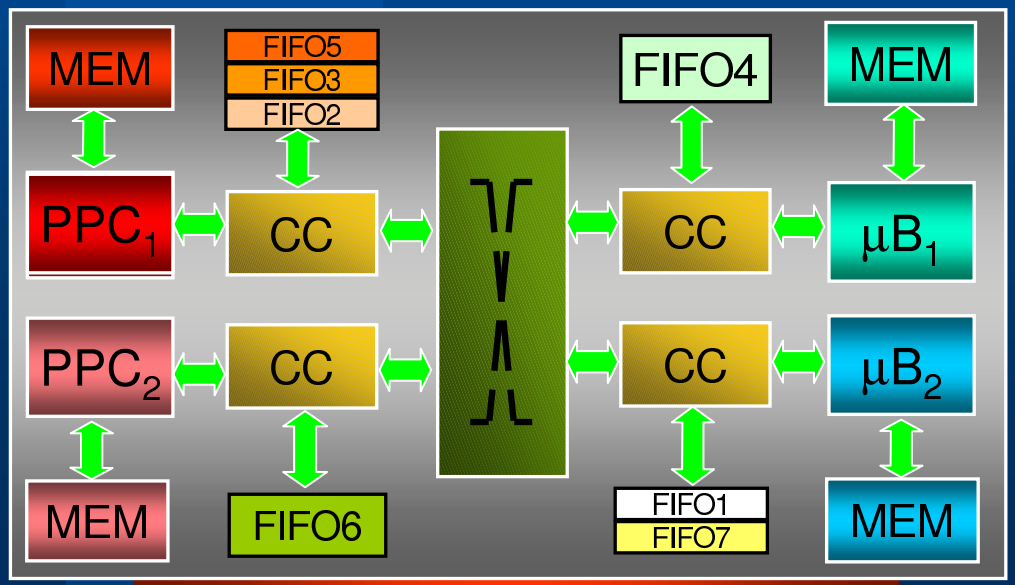
Kahn Process Networks



- Parallel specification
  $\Rightarrow$ easier to map onto parallel hardware
- Consists of
  - independent processes
  - communication channels
    $\Rightarrow$ unbounded FIFOs / memories
  - synchronization: blocking read + blocking write

MPSoCs Programming and Mapping





- Programming model: KPN
- Example of a MPSoC platform:
  - 2 PowerPC processors
  - 2 MicroBlaze processors
  - 4 Communication memories (CM)
  - 4 Communication controllers (CC)
  - Crossbar component
- Mapping:
  - The processes are mapped onto the processors using standard GCC compiler.
  - The FIFOs are mapped onto the CMs depending on the process mapping.
  - CCs take care about the inter processor communication and synchronization

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis

  Given a read from an array element, what was the last write to that array element?

- counting

  Given a communication channel, how many tokens are in the channel at any given time?

- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis

  Given a read from an array element, what was the last write to that array element?

- counting

  Given a communication channel, how many tokens are in the channel at any given time?

- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis

  Given a read from an array element, what was the last write to that array element?

- counting

  Given a communication channel, how many tokens are in the channel at any given time?

- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Polytope Model

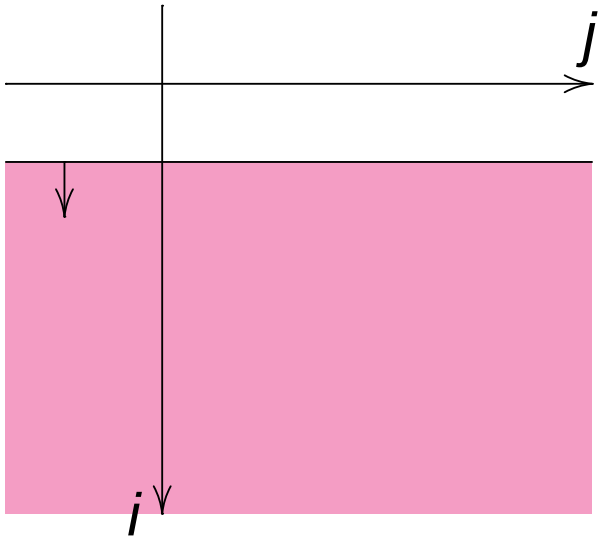
```
#define N 5
for (i = 1; i <= N; ++i)
    for (j = 1; j <= i; ++j)
        a[i][j]=
```

Assumptions on sequential code:

- iterators are integers
- loops with affine bounds
- affine conditions
- affine index expressions

Polytope Model

```
#define N 5
for (i = 1; i <= N; ++i)
    for (j = 1; j <= i; ++j)
        a[i][j]=
```



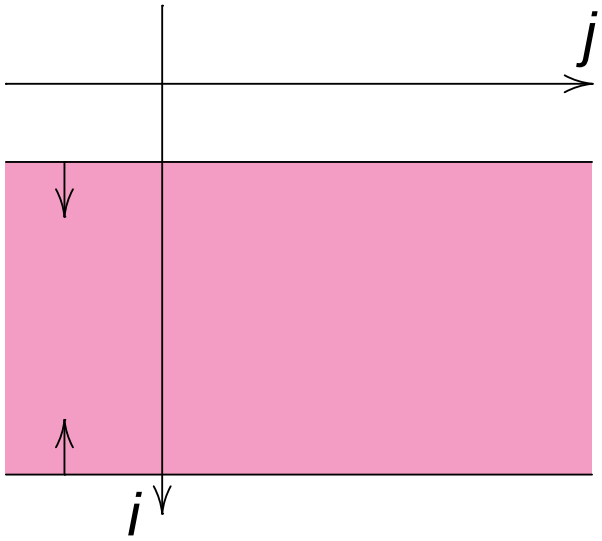
Iteration domain: $P = \{ [i,j] \mid i \geq 1 \}$

Assumptions on sequential code:

- iterators are integers
- loops with affine bounds
- affine conditions
- affine index expressions

Polytope Model

```
#define N 5
for (i = 1; i <= N; ++i)
    for (j = 1; j <= i; ++j)
        a[i][j]=
```



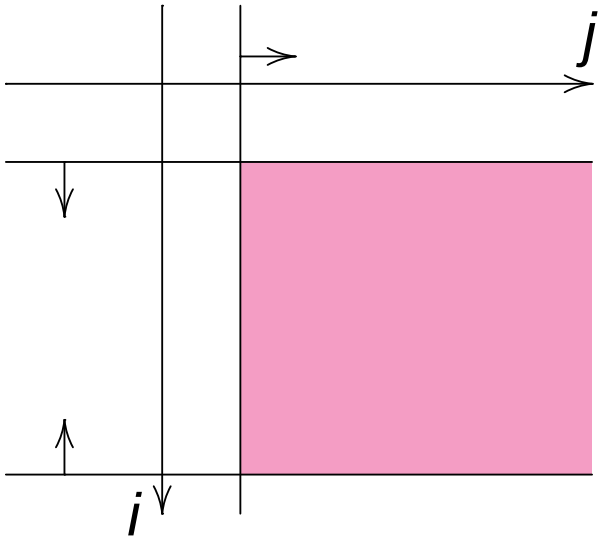
Iteration domain: $P = \{ [i,j] \mid i \geq 1 \wedge i \leq N \}$

Assumptions on sequential code:

- iterators are integers
- loops with affine bounds
- affine conditions
- affine index expressions

Polytope Model

```
#define N 5
for (i = 1; i <= N; ++i)
    for (j = 1; j <= i; ++j)
        a[i][j]=
```



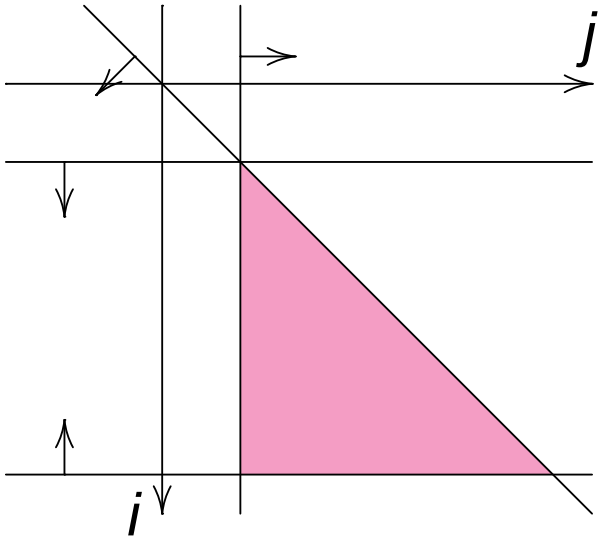
Iteration domain: $P = \{ [i,j] \mid i \geq 1 \wedge i \leq N \wedge j \geq 1 \quad \}$

Assumptions on sequential code:

- iterators are integers
- loops with affine bounds
- affine conditions
- affine index expressions

Polytope Model

```
#define N 5
for (i = 1; i <= N; ++i)
    for (j = 1; j <= i; ++j)
        a[i][j]=
```



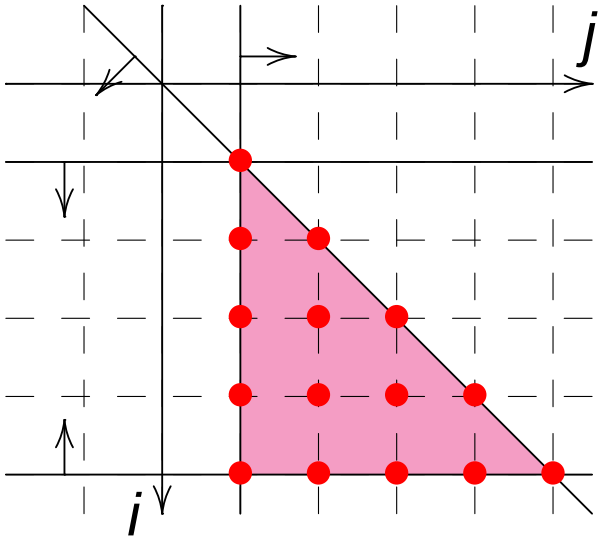
Iteration domain: $P = \{ [i,j] \mid i \geq 1 \wedge i \leq N \wedge j \geq 1 \wedge j \leq i \}$

Assumptions on sequential code:

- iterators are integers
- loops with affine bounds
- affine conditions
- affine index expressions

Polytope Model

```
#define N 5
for (i = 1; i <= N; ++i)
    for (j = 1; j <= i; ++j)
        a[i][j]=
```



Iteration domain: $P = \{ [i,j] \mid i \geq 1 \wedge i \leq N \wedge j \geq 1 \wedge j \leq i \}$

Assumptions on sequential code:

- iterators are integers
- loops with affine bounds
- affine conditions
- affine index expressions

$\mathcal{H}$-Parametric Polytopes

```
for (i = 1; i <= N; ++i)
    for (j = 1; j <= i; ++j)
        a[i][j]=
```

$$P(N) = \{ (i,j) \mid i \geq 1 \wedge i \leq N \wedge j \geq 1 \wedge j \leq i \}$$

$\Rightarrow$ a set of polytopes, one for each value of N

parametric vertices:

$$P = \text{conv}\left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} N \\ 1 \end{bmatrix}, \begin{bmatrix} N \\ N \end{bmatrix} \right\}$$

In general: different (active) vertices on different parts of the parameter space (chamber decomposition)

$\mathcal{H}$-Parametric Polytopes

```
for (i = 1; i <= N; ++i)
    for (j = 1; j <= i; ++j)
        a[i][j]=
```

$$P(N) = \{ (i,j) \mid i \geq 1 \wedge i \leq N \wedge j \geq 1 \wedge j \leq i \}$$

$\Rightarrow$ a set of polytopes, one for each value of N

parametric vertices:

$$P = \text{conv} \left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} N \\ 1 \end{bmatrix}, \begin{bmatrix} N \\ N \end{bmatrix} \right\}$$

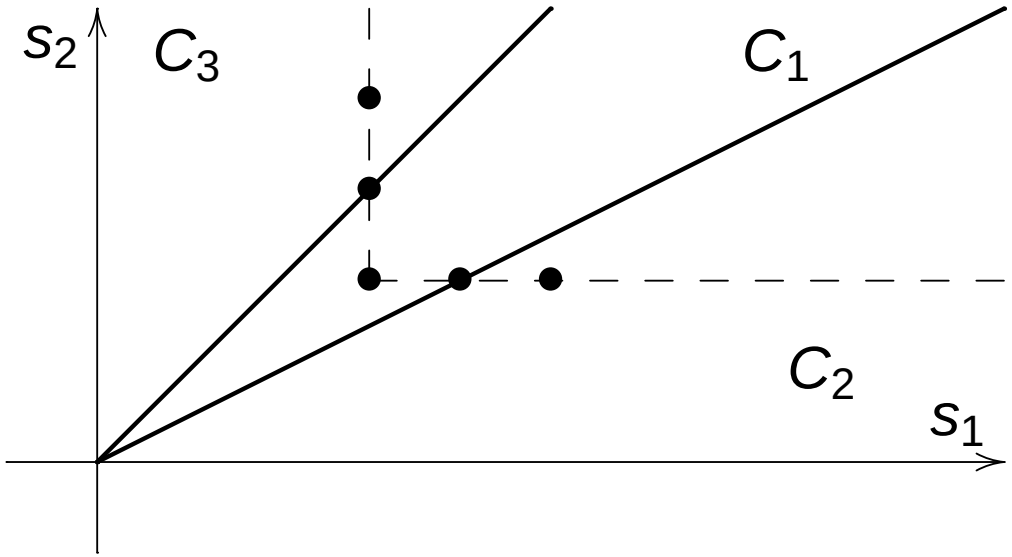
In general: different (active) vertices on different parts of the parameter space (chamber decomposition)

Chamber Decomposition

$$P(\mathbf{s}) = \left\{ \mathbf{t} \in \mathbb{Q}^2 \mid -s_1 + 2s_2 + t_1 - 2t_2 \geq 0 \wedge s_1 - s_2 - t_1 + t_2 \geq 0 \wedge t_1 \geq 0 \wedge t_2 \geq 0 \right\}$$

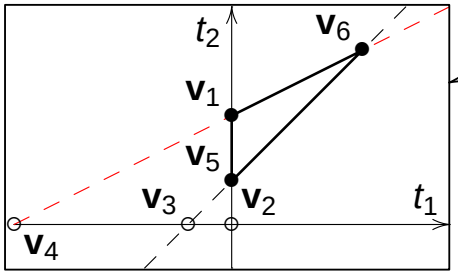
Chamber Decomposition

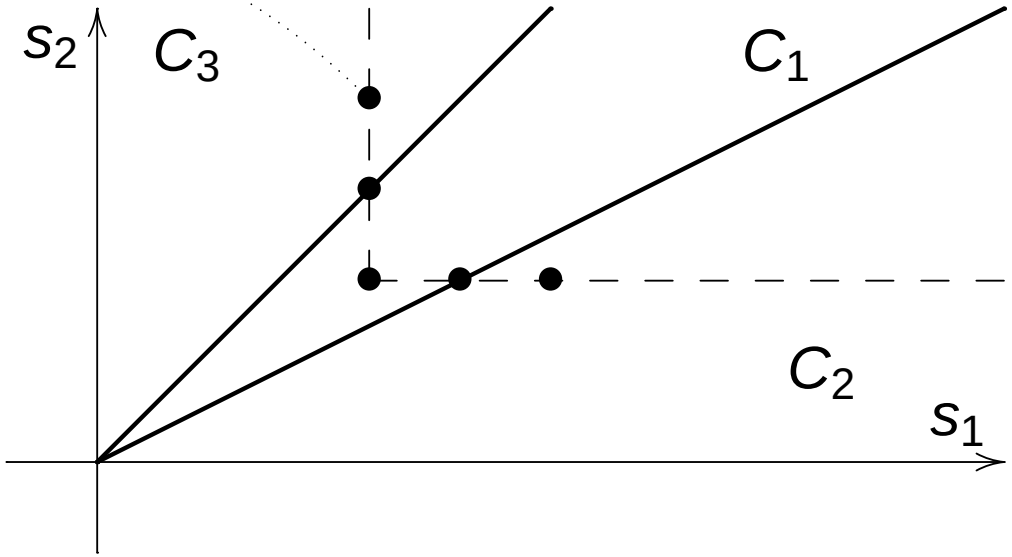
$$P(\mathbf{s}) = \left\{ \mathbf{t} \in \mathbb{Q}^2 \mid -s_1 + 2s_2 + t_1 - 2t_2 \geq 0 \wedge s_1 - s_2 - t_1 + t_2 \geq 0 \wedge t_1 \geq 0 \wedge t_2 \geq 0 \right\}$$



Chamber Decomposition

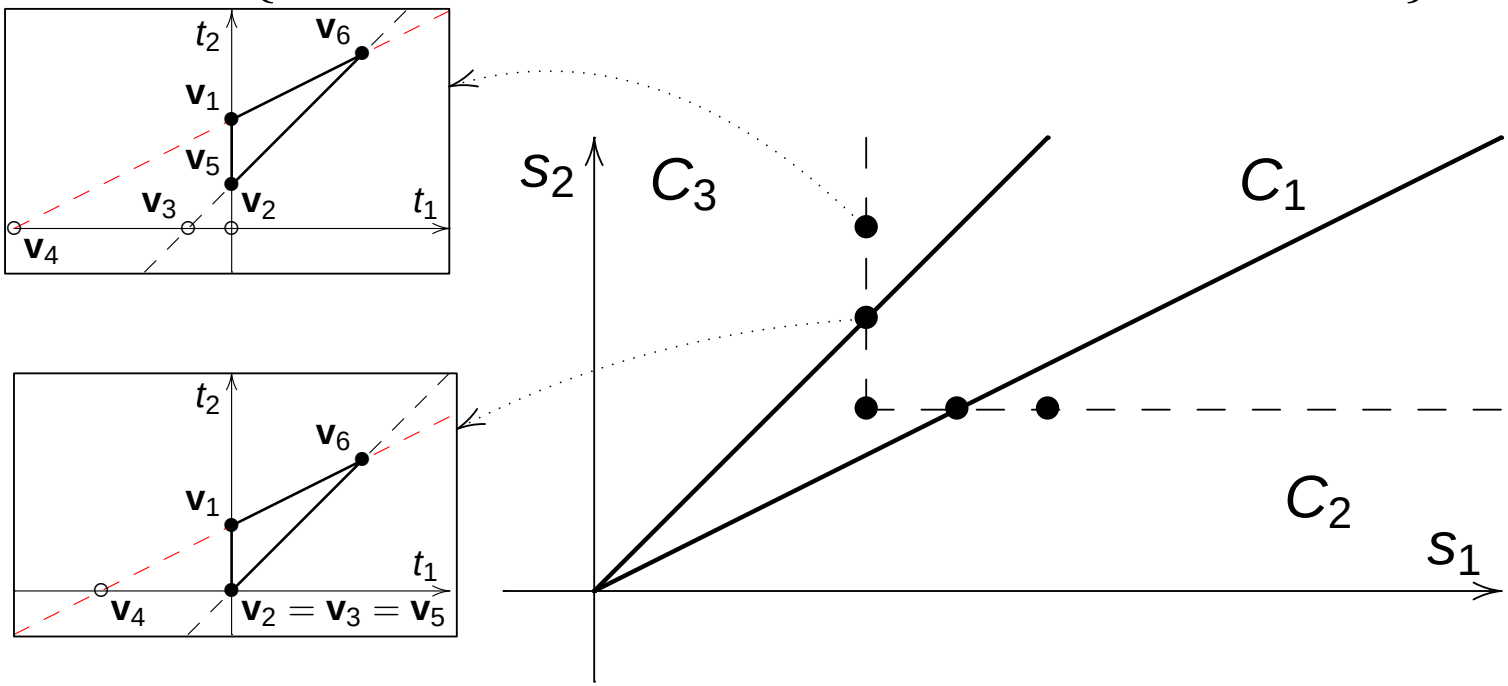
$$P(\mathbf{s}) = \left\{ \mathbf{t} \in \mathbb{Q}^2 \mid -s_1 + 2s_2 + t_1 - 2t_2 \geq 0 \wedge s_1 - s_2 - t_1 + t_2 \geq 0 \wedge t_1 \geq 0 \wedge t_2 \geq 0 \right\}$$





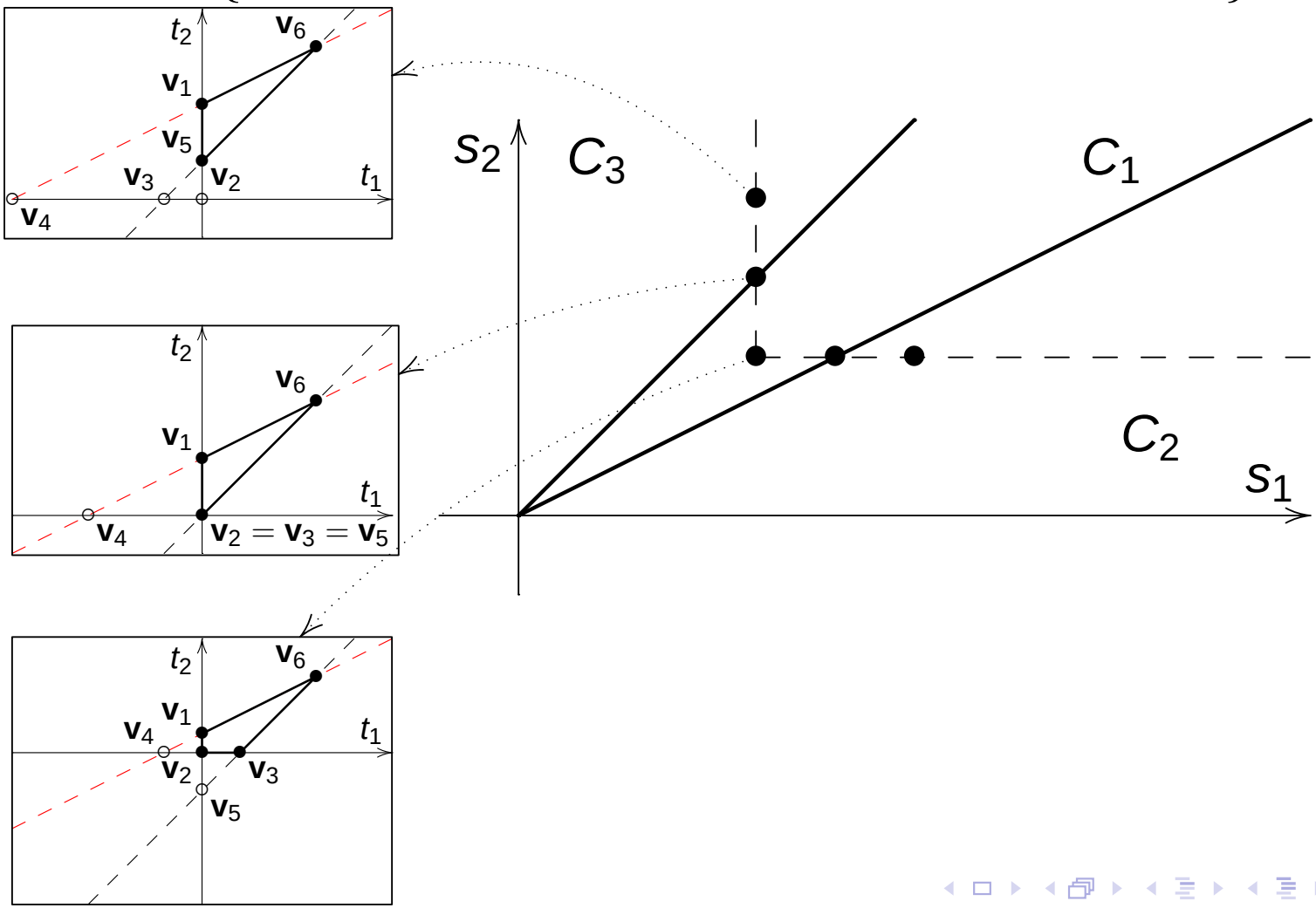
Chamber Decomposition

$$P(\mathbf{s}) = \left\{ \mathbf{t} \in \mathbb{Q}^2 \mid -s_1 + 2s_2 + t_1 - 2t_2 \geq 0 \wedge s_1 - s_2 - t_1 + t_2 \geq 0 \wedge t_1 \geq 0 \wedge t_2 \geq 0 \right\}$$



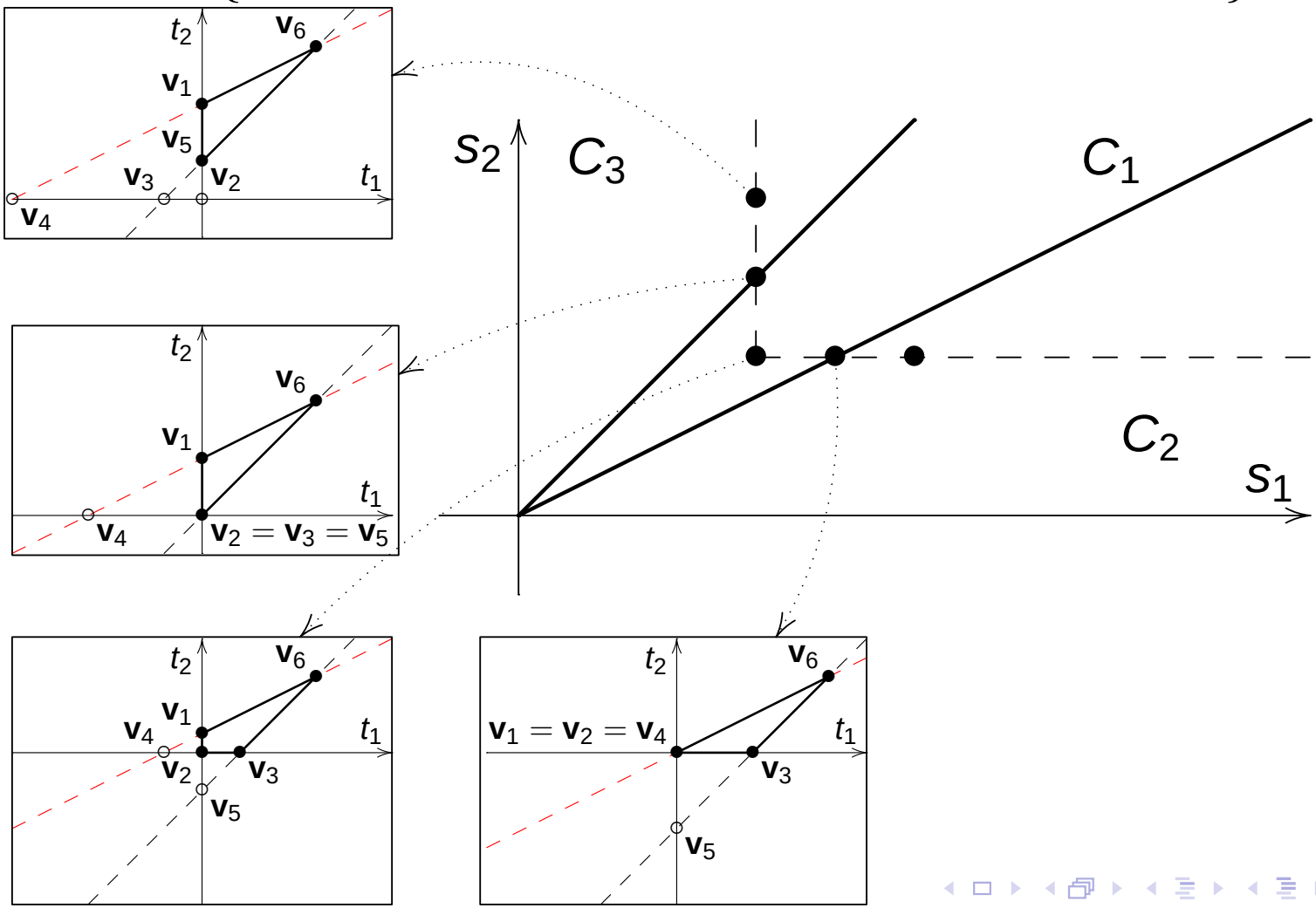
Chamber Decomposition

$$P(\mathbf{s}) = \left\{ \mathbf{t} \in \mathbb{Q}^2 \mid -s_1 + 2s_2 + t_1 - 2t_2 \geq 0 \wedge s_1 - s_2 - t_1 + t_2 \geq 0 \wedge t_1 \geq 0 \wedge t_2 \geq 0 \right\}$$



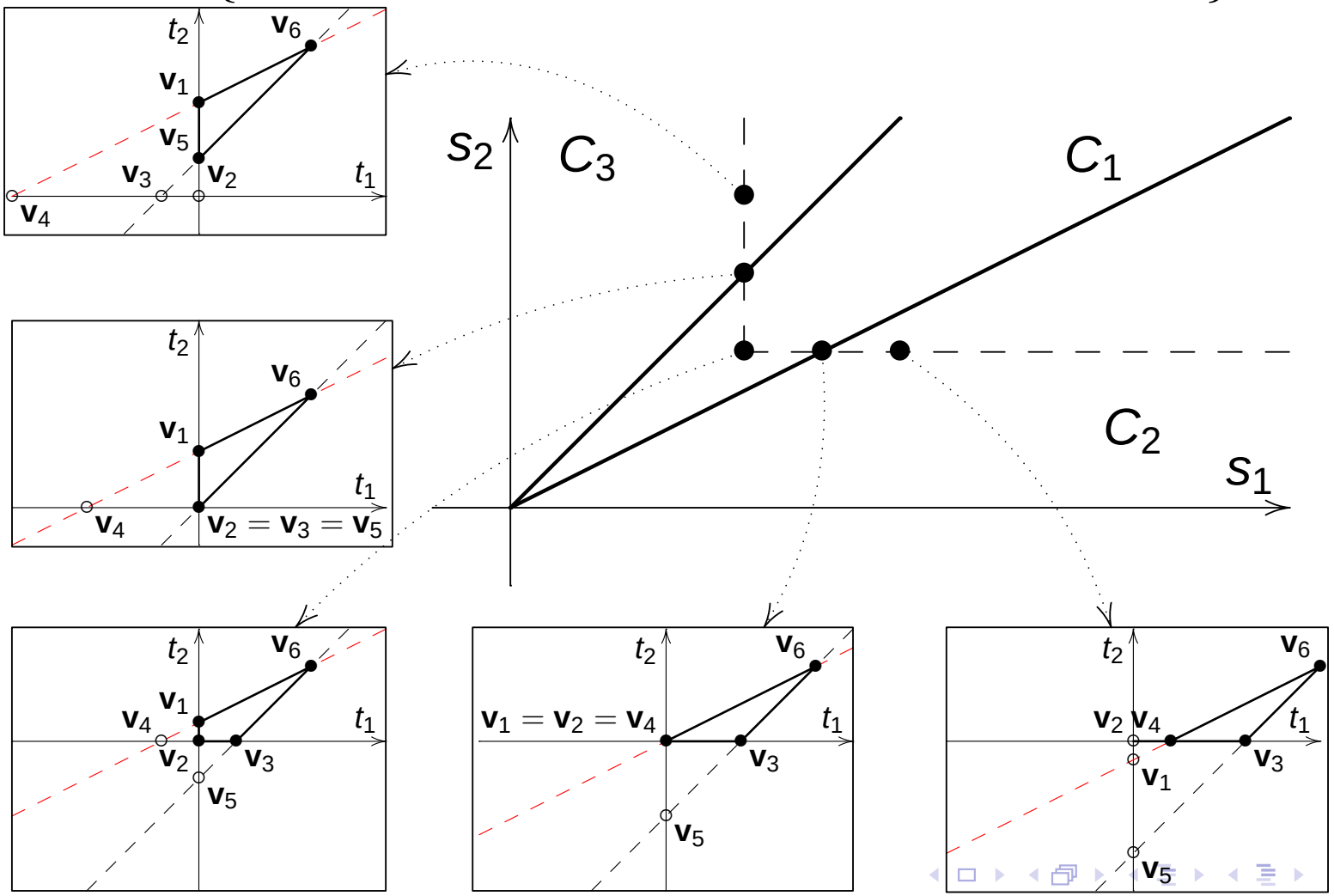
Chamber Decomposition

$$P(\mathbf{s}) = \left\{ \mathbf{t} \in \mathbb{Q}^2 \mid -s_1 + 2s_2 + t_1 - 2t_2 \geq 0 \wedge s_1 - s_2 - t_1 + t_2 \geq 0 \wedge t_1 \geq 0 \wedge t_2 \geq 0 \right\}$$



Chamber Decomposition

$$P(\mathbf{s}) = \left\{ \mathbf{t} \in \mathbb{Q}^2 \mid -s_1 + 2s_2 + t_1 - 2t_2 \geq 0 \wedge s_1 - s_2 - t_1 + t_2 \geq 0 \wedge t_1 \geq 0 \wedge t_2 \geq 0 \right\}$$



Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis

  Given a read from an array element, what was the last write to that array element?

- counting

  Given a communication channel, how many tokens are in the channel at any given time?

- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis

  Given a read from an array element, what was the last write to that array element?

- counting

  Given a communication channel, how many tokens are in the channel at any given time?

- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Example

```
int N, M;
#pragma parameter N 10 100
#pragma parameter M 10 100

int main(void)
{
    int i, j, k;
    int a[N][N];

    for (i = 0; i < N; ++i)
        for (j = 0; j < N; ++j)
            a[i][j] = Read();
    for (j = 0; j < N; ++j) {
        for (k = 0; k < N; ++k)
            if (M < N)
                Switch(a[j][k], a[M][k], &a[j][k], &a[M][k]);
        for (i = j+1; i < N; ++i)
           a[i][j] = Imp(a[i][j], a[j][j]);
        for (k = j+1; k < N; ++k)
            for (i = j+1; i < N; ++i)
                 a[i][k] = MulSubs(a[i][k],a[i][j],a[j][k]);
    }
    for (i = 0; i < N; ++i)
        for (j = 0; j < N; ++j)
            a[i][j] = Pass(a[i][j]);
}
```

Example

```
int N, M;
#pragma parameter N 10 100
#pragma parameter M 10 100

int main(void)
{
    int i, j, k;
    int a[N][N];

    for (i = 0; i < N; ++i)
        for (j = 0; j < N; ++j)
            a[i][j] = Read();
    for (j = 0; j < N; ++j) {
        for (k = 0; k < N; ++k)
            if (M < N)
                Switch(a[j][k], a[M][k], &a[j][k], &a[M][k]);
        for (i = j+1; i < N; ++i)
           a[i][j] = Imp(a[i][j], a[j][j]);
        for (k = j+1; k < N; ++k)
            for (i = j+1; i < N; ++i)
                 a[i][k] = MulSubs(a[i][k],a[i][j],a[j][k]);
    }
    for (i = 0; i < N; ++i)
        for (j = 0; j < N; ++j)
            a[i][j] = Pass(a[i][j]);
}
```

Dataflow Analysis

Given a read from an array element, what was the last write to that array element?

Step 1: last write through given write access

```
for (j = 0; j < N; ++j)
  for (k = j+1; k < N; ++k)
    for (i = j+1; i < N; ++i)
      a[i][k] = MulSubs(a[i][k],a[i][j],a[j][k]);
```

Context: $10 \leq M \leq 100 \wedge 10 \leq N \leq 100$

Iteration domain:

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$

$$\text{lexmax}\{(j_w,k_w,i_w) \in S \mid (j_w,k_w,i_w) \prec (j_r,k_r,i_r) \wedge (j_r,k_r,i_r) \in S \wedge (k_w,i_w) = (k_r,i_r)\}$$

Dataflow Analysis

Given a read from an array element, what was the last write to that array element?

Step 1: last write through given write access

```
for (j = 0; j < N; ++j)
  for (k = j+1; k < N; ++k)
    for (i = j+1; i < N; ++i)
      a[i][k] = MulSubs(a[i][k],a[i][j],a[j][k]);
```

Context: $10 \leq M \leq 100 \wedge 10 \leq N \leq 100$

Iteration domain:

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$

$$\text{lexmax}\{(j_w,k_w,i_w) \in S \mid (j_w,k_w,i_w) \prec (j_r,k_r,i_r) \wedge (j_r,k_r,i_r) \in S \wedge (k_w,i_w) = (k_r,i_r)\}$$

Dataflow Analysis

Step 1: last write through given write access

Iteration domain:

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$

$$\text{lexmax}\{(j_w,k_r,i_r) \in S \mid (j_w,k_r,i_r) \prec (j_r,k_r,i_r) \wedge (j_r,k_r,i_r) \in S\}$$

$$\text{lexmax}\{(j_w,k_r,i_r) \in S \mid (j_w,k_r) = (j_r,k_r) \wedge i_r < i_r \wedge (j_r,k_r,i_r) \in S\} \vee$$
$$\text{lexmax}\{(j_w,k_r,i_r) \in S \mid j_w = j_r \wedge k_r < k_r \wedge (j_r,k_r,i_r) \in S\} \vee$$
$$\text{lexmax}\{(j_w,k_r,i_r) \in S \mid j_w < j_r \wedge (j_r,k_r,i_r) \in S\}$$

Lexicographic maximum of parametric polytope

Variables: write iterators

Parameters: read iterators and structural parameters

Dataflow Analysis

Step 1: last write through given write access

Iteration domain:

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$

$$\text{lexmax}\{(j_w,k_r,i_r) \in S \mid (j_w,k_r,i_r) \prec (j_r,k_r,i_r) \wedge (j_r,k_r,i_r) \in S\}$$

$$\begin{aligned}
&\text{lexmax}\{(j_w,k_r,i_r) \in S \mid (j_w,k_r) = (j_r,k_r) \wedge i_r < i_r \wedge (j_r,k_r,i_r) \in S\} \vee \\
&\text{lexmax}\{(j_w,k_r,i_r) \in S \mid j_w = j_r \wedge k_r < k_r \wedge (j_r,k_r,i_r) \in S\} \vee \\
&\text{lexmax}\{(j_w,k_r,i_r) \in S \mid j_w < j_r \wedge (j_r,k_r,i_r) \in S\}
\end{aligned}$$

Lexicographic maximum of parametric polytope

Variables: write iterators

Parameters: read iterators and structural parameters

Dataflow Analysis

Step 1: last write through given write access

Iteration domain:

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$

$$\text{lexmax}\{(j_w,k_r,i_r) \in S \mid (j_w,k_r,i_r) \prec (j_r,k_r,i_r) \wedge (j_r,k_r,i_r) \in S\}$$

$$\begin{aligned}
&\text{lexmax}\{(j_w,k_r,i_r) \in S \mid (j_w,k_r) = (j_r,k_r) \wedge i_r < i_r \wedge (j_r,k_r,i_r) \in S\} \vee \\
&\text{lexmax}\{(j_w,k_r,i_r) \in S \mid j_w = j_r \wedge k_r < k_r \wedge (j_r,k_r,i_r) \in S\} \vee \\
&\text{lexmax}\{(j_w,k_r,i_r) \in S \mid j_w < j_r \wedge (j_r,k_r,i_r) \in S\}
\end{aligned}$$

Lexicographic maximum of parametric polytope

Variables: write iterators

Parameters: read iterators and structural parameters

Example

```
int N, M;
#pragma parameter N 10 100
#pragma parameter M 10 100

int main(void)
{
    int i, j, k;
    int a[N][N];

    for (i = 0; i < N; ++i)
        for (j = 0; j < N; ++j)
            a[i][j] = Read();
    for (j = 0; j < N; ++j) {
        for (k = 0; k < N; ++k)
            if (M < N)
                Switch(a[j][k], a[M][k], &a[j][k], &a[M][k]);
        for (i = j+1; i < N; ++i)
           a[i][j] = Imp(a[i][j], a[j][j]);
        for (k = j+1; k < N; ++k)
            for (i = j+1; i < N; ++i)
                 a[i][k] = MulSubs(a[i][k],a[i][j],a[j][k]);
    }
    for (i = 0; i < N; ++i)
        for (j = 0; j < N; ++j)
            a[i][j] = Pass(a[i][j]);
}
```

Example

```
int N, M;
#pragma parameter N 10 100
#pragma parameter M 10 100

int main(void)
{
    int i, j, k;
    int a[N][N];

    for (i = 0; i < N; ++i)
        for (j = 0; j < N; ++j)
            a[i][j] = Read();
    for (j = 0; j < N; ++j) {
        for (k = 0; k < N; ++k)
            if (M < N)
                Switch(a[j][k], a[M][k], &a[j][k], &a[M][k]);
        for (i = j+1; i < N; ++i)
           a[i][j] = Imp(a[i][j], a[j][j]);
        for (k = j+1; k < N; ++k)
            for (i = j+1; i < N; ++i)
                 a[i][k] = MulSubs(a[i][k],a[i][j],a[j][k]);
    }
    for (i = 0; i < N; ++i)
        for (j = 0; j < N; ++j)
            a[i][j] = Pass(a[i][j]);
}
```

Dataflow Analysis

Step 1: last write through given write access
Step 2: combination of results over all write accesses

```
for (j = 0; j < N; ++j)
  for (k = j+1; k < N; ++k)
    for (i = j+1; i < N; ++i)
      a[i][k] = MulSubs(a[i][k],a[i][j],a[j][k]);
```

Final result:

$$(j_w, k_w, i_w) = \begin{cases} (j_r - 1, k_r, i_r) & \text{if } N \leq M \\ (j_r - 1, k_r, i_r) & \text{if } M < N \wedge M < i_r \\ (j_r - 1, k_r, i_r) & \text{if } i_r < M < N \end{cases}$$

Dataflow Analysis

Step 1: last write through given write access
Step 2: combination of results over all write accesses

```
for (j = 0; j < N; ++j)
  for (k = j+1; k < N; ++k)
    for (i = j+1; i < N; ++i)
      a[i][k] = MulSubs(a[i][k],a[i][j],a[j][k]);
```

Final result:

$$(j_w, k_w, i_w) = \begin{cases} (j_r - 1, k_r, i_r) & \text{if } N \leq M \\ (j_r - 1, k_r, i_r) & \text{if } M < N \wedge M < i_r \\ (j_r - 1, k_r, i_r) & \text{if } i_r < M < N \end{cases}$$

Dataflow Analysis

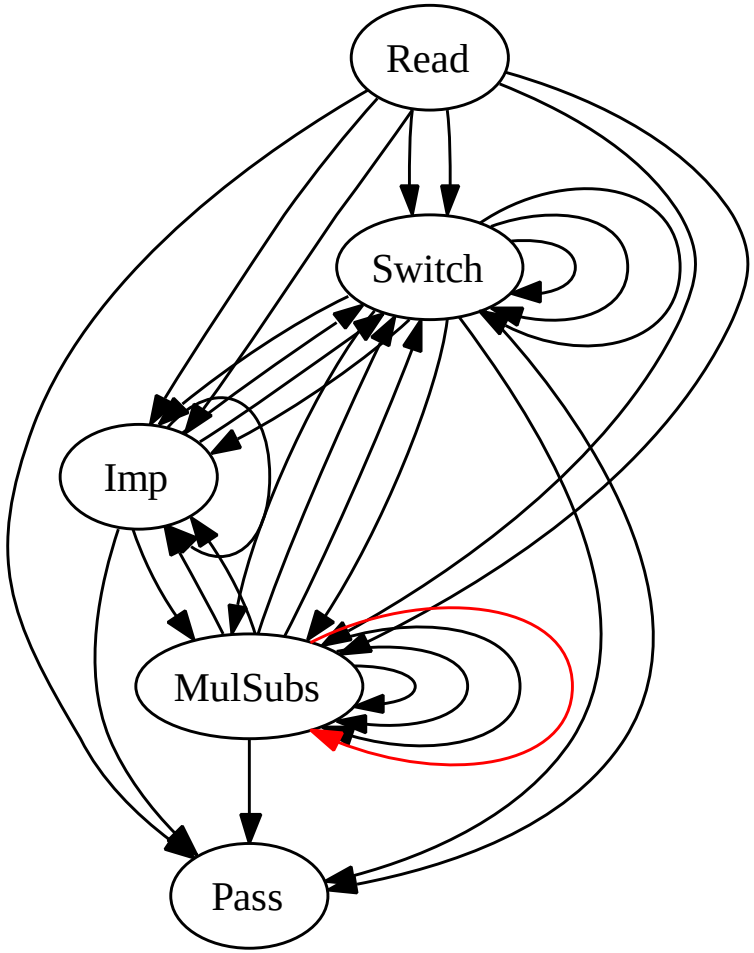
Step 1: last write through given write access
Step 2: combination of results over all write accesses

```
for (j = 0; j < N; ++j)
  for (k = j+1; k < N; ++k)
    for (i = j+1; i < N; ++i)
      a[i][k] = MulSubs(a[i][k],a[i][j],a[j][k]);
```

Final result:

$$(j_w, k_w, i_w) = \begin{cases} (j_r - 1, k_r, i_r) & \text{if } N \leq M \\ (j_r - 1, k_r, i_r) & \text{if } M < N \wedge M < i_r \\ (j_r - 1, k_r, i_r) & \text{if } i_r < M < N \end{cases}$$

Dataflow Analysis



Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis

  Given a read from an array element, what was the last write to that array element?

- counting

  Given a communication channel, how many tokens are in the channel at any given time?

- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis

  Given a read from an array element, what was the last write to that array element?

  Lexicographic maximum of parametric polytope (`PIP`)
  $\Rightarrow$ piecewise quasi-affine map
- counting

  Given a communication channel, how many tokens are in the channel at any given time?

- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis

  Given a read from an array element, what was the last write to that array element?

  Lexicographic maximum of parametric polytope (`PIP`)
  $\Rightarrow$ piecewise quasi-affine map
- counting

  Given a communication channel, how many tokens are in the channel at any given time?

- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Parametric Counting

Given a communication channel, how many tokens are in the channel at any given time?

Iteration domain:

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$

Dependence relation (partial result from dataflow analysis):

$$R = R_1 \cup R_2 \cup R_3$$

$$R_2 = \{((j_r - 1, k_r, i_r), (j_r, k_r, i_r)) \in S \times S \mid M < N \wedge M < i_r\}$$

Assume communication channel is a FIFO:
Number of elements in FIFO at (j,k,i) is number of writes before (j,k,i) minus number of reads before (j,k,i)

$$\#\{(j_w, k_w, i_w) \in \operatorname{dom} R \mid (j_w, k_w, i_w) \prec (j,k,i)\}$$
$$- \#\{(j_r, k_r, i_r) \in \operatorname{ran} R \mid (j_r, k_r, i_r) \prec (j,k,i)\}$$

Parametric Counting

Given a communication channel, how many tokens are in the channel at any given time?

Iteration domain:

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$

Dependence relation (partial result from dataflow analysis):

$$R = R_1 \cup R_2 \cup R_3$$

$$R_2 = \{((j_r - 1, k_r, i_r), (j_r, k_r, i_r)) \in S \times S \mid M < N \wedge M < i_r\}$$

Assume communication channel is a FIFO:

Number of elements in FIFO at (j,k,i) is number of writes before (j,k,i) minus number of reads before (j,k,i)

$$\#\{(j_w, k_w, i_w) \in \operatorname{dom} R \mid (j_w, k_w, i_w) \prec (j,k,i)\} - \#\{(j_r, k_r, i_r) \in \operatorname{ran} R \mid (j_r, k_r, i_r) \prec (j,k,i)\}$$

Parametric Counting: Order Predicate

$$\#\{(j_w, k_w, i_w) \in \operatorname{dom} R \mid (j_w, k_w, i_w) \prec (j, k, i)\} - \#\{(j_r, k_r, i_r) \in \operatorname{ran} R \mid (j_r, k_r, i_r) \prec (j, k, i)\}$$

Ordering of iterations

- fixed within the same process
- unknown at compile-time between different processes
  $\Rightarrow$ combine all processes into a single process
  $\Rightarrow$ perform counting in the combined process

Process Merging

Given: iteration domains of dimension $d_i \leq D$

1. Place in a common iteration space of dimension D
   (no change of order within an iteration domain)
   Heuristic: identify dimensions related through dependence relation
2. Add constant offset to all iteration domains
   (without violating any dependence)
   Heuristic: put dependent iterations “close” together

Process Merging: Common Iteration Space

```
for (i = 0; i <= 5; ++i)
    a[i] = ...
for (i = 0; i <= 5; ++i)
    for (j = 0; j <= 5; ++j)
        b[i][j] = a[i];
```

$$R = \{(i, i', j') \mid 0 \leq i \leq 5 \wedge 0 \leq i' \leq 5 \wedge 0 \leq j' \leq 5 \wedge i = i'\}$$

$\Rightarrow$ identify i and i'

$\Rightarrow$, not

Process Merging: Common Iteration Space

```
for (i = 0; i <= 5; ++i)
    a[i] = ...
for (i = 0; i <= 5; ++i)
    for (j = 0; j <= 5; ++j)
        b[i][j] = a[i];
```

$$R = \{(i, i', j') \mid 0 \leq i \leq 5 \wedge 0 \leq i' \leq 5 \wedge 0 \leq j' \leq 5 \wedge i = i'\}$$

$\Rightarrow$ identify i and i'

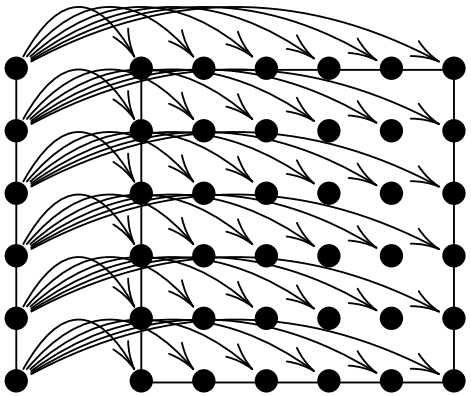
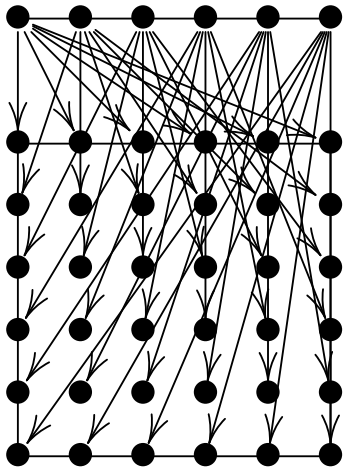
$\Rightarrow$, not

Process Merging: Common Iteration Space

```
for (i = 0; i <= 5; ++i)
    a[i] = ...
for (i = 0; i <= 5; ++i)
    for (j = 0; j <= 5; ++j)
        b[i][j] = a[i];
```

$$R = \{(i, i', j') \mid 0 \leq i \leq 5 \wedge 0 \leq i' \leq 5 \wedge 0 \leq j' \leq 5 \wedge i = i'\}$$

$\Rightarrow$ identify i and i'

$\Rightarrow$  , not 

Process Merging

Given: iteration domains of dimension $d_i \leq D$

1. Place in a common iteration space of dimension D
   (no change of order within an iteration domain)
   Heuristic: identify dimensions related through dependence relation
2. Add constant offset to all iteration domains
   (without violating any dependence)
   Heuristic: put dependent iterations “close” together

Process Merging

Given: iteration domains of dimension $d_i \leq D$

1. Place in a common iteration space of dimension D
   (no change of order within an iteration domain)
   Heuristic: identify dimensions related through dependence relation
2. Add constant offset to all iteration domains
   (without violating any dependence)
   Heuristic: put dependent iterations “close” together

Process Merging: Example

Original QR code and code equivalent to merged QR

```
for (j = 0; j < N; ++j)
  for (i = j; i < N; ++i)
    R[j][i] = Zero();
for (k = 0; k < K; ++k)
  for (j = 0; j < N; ++j)
    X[k][j] = ReadMatrix();
for (k = 0; k < K; ++k)
  for (j = 0; j < N; ++j) {
    Vectorize(R[j][j], X[k][j],
              &R[j][j], &X[k][j], &t);
  for (i = j+1; i < N; ++i)
    Rotate(R[j][i], X[k][i], t,
           &R[j][i], &X[k][i], &t);
  }
for (j = 0; j < N; ++j)
  for (i = j; i < N; ++i)
    WriteMatrix(R[j][i]);
```

```
for (a = 0; a <= K-1 a++)
  for (b = 0; b <= N-1; b++)
    for (c = b; c <= 2*N-2; c++) {
      if (a == 0 && c <= N-1)
        R[b][c] = Zero();
      if (b == 0 && N-1 <= c)
        X[a][c-N+1] = ReadMatrix();
      if (c == N-1)
        Vectorize(R[b][b], X[a][b],
                  &R[b][b], &X[a][b], &t);
      if (b <= c-N)
        Rotate(R[b][c-N+1],X[a][c-N+1],t,
               &R[b][c-N+1],&X[a][c-N+1],&t);
      if (a == K-1 & b <= c-N+1)
        WriteMatrix(R[b][c-N+1]);
    }
```

Parametric Counting

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$
$$R = R_1 \cup R_2 \cup R_3$$
$$R_2 = \{((j_r - 1, k_r, i_r), (j_r, k_r, i_r)) \in S \times S \mid M < N \wedge M < i_r\}$$

$$\#\{(j_w,k_w,i_w) \in \operatorname{dom} R \mid (j_w,k_w,i_w) \prec (j,k,i)\} - \#\{(j_r,k_r,i_r) \in \operatorname{ran} R \mid (j_r,k_r,i_r) \prec (j,k,i)\}$$

Number of integer points in parametric polytope

Variables: write/read iterators

Parameters: “inspection” iterators and structural parameters

$$\begin{cases} i - Nk - Ni + N^2 & \text{if } i \leq k \leq i+1 \wedge i \geq M+2 \\ -k + i^2 - 2Ni + 2i + N^2 - N + 1 & \text{if } k \geq i+2 \wedge i \geq M+2 \end{cases}$$

$$(\text{on } \operatorname{ran} R_2 \cap \{(j,k,i) \mid i = j+1\})$$

Parametric Counting

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$
$$R = R_1 \cup R_2 \cup R_3$$
$$R_2 = \{((j_r - 1, k_r, i_r), (j_r, k_r, i_r)) \in S \times S \mid M < N \wedge M < i_r\}$$

$$\#\{(j_w,k_w,i_w) \in \operatorname{dom} R \mid (j_w,k_w,i_w) \prec (j,k,i)\} - \#\{(j_r,k_r,i_r) \in \operatorname{ran} R \mid (j_r,k_r,i_r) \prec (j,k,i)\}$$

Number of integer points in parametric polytope

Variables: write/read iterators

Parameters: “inspection” iterators and structural parameters

$$\begin{cases} i - Nk - Ni + N^2 & \text{if } i \leq k \leq i+1 \wedge i \geq M+2 \\ -k + i^2 - 2Ni + 2i + N^2 - N + 1 & \text{if } k \geq i+2 \wedge i \geq M+2 \end{cases}$$

$$(\text{on } \operatorname{ran} R_2 \cap \{(j,k,i) \mid i = j+1\})$$

Parametric Counting

$$S = \{(j,k,i) \mid 0 \le j < N \wedge j+1 \le k < N \wedge j+1 \le i < N\}$$
$$R = R_1 \cup R_2 \cup R_3$$
$$R_2 = \{((j_r - 1, k_r, i_r), (j_r, k_r, i_r)) \in S \times S \mid M < N \wedge M < i_r\}$$

$$\#\{(j_w,k_w,i_w) \in \operatorname{dom} R \mid (j_w,k_w,i_w) \prec (j,k,i)\} - \#\{(j_r,k_r,i_r) \in \operatorname{ran} R \mid (j_r,k_r,i_r) \prec (j,k,i)\}$$

Number of integer points in parametric polytope

Variables: write/read iterators

Parameters: “inspection” iterators and structural parameters

$$\begin{cases} i - Nk - Ni + N^2 & \text{if } i \le k \le i+1 \wedge i \ge M+2 \\ -k + i^2 - 2Ni + 2i + N^2 - N + 1 & \text{if } k \ge i+2 \wedge i \ge M+2 \end{cases}$$

$$(\text{on } \operatorname{ran} R_2 \cap \{(j,k,i) \mid i = j+1\})$$

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis
  Lexicographic maximum of parametric polytope (`PIP`)
  $\Rightarrow$ piecewise quasi-affine map
- counting

  Given a communication channel, how many tokens are in the channel at any given time?

- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis
  Lexicographic maximum of parametric polytope (`PIP`)
  $\Rightarrow$ piecewise quasi-affine map
- counting

  Given a communication channel, how many tokens are in the channel at any given time?

  Number of elements in parametric polytope (`barvinok`)
  $\Rightarrow$ piecewise quasi-polynomial
- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis
  Lexicographic maximum of parametric polytope (`PIP`)
  $\Rightarrow$ piecewise quasi-affine map
- counting

  Given a communication channel, how many tokens are in the channel at any given time?

  Number of elements in parametric polytope (`barvinok`)
  $\Rightarrow$ piecewise quasi-polynomial
- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Maximization

Given a communication channel, what is the maximal number of tokens in the channel during the execution?

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$
$$R = R_1 \cup R_2 \cup R_3$$
$$R_2 = \{((j_r - 1, k_r, i_r), (j_r, k_r, i_r)) \in S \times S \mid M < N \wedge M < i_r\}$$

Tokens in FIFO at (j,k,i) (on $\operatorname{ran} R_2 \cap \{(j,k,i) \mid i = j+1\}$):

$$\begin{cases} i - Nk - Ni + N^2 & \text{if } i \leq k \leq i+1 \wedge i \geq M+2 \\ -k + i^2 - 2Ni + 2i + N^2 - N + 1 & \text{if } k \geq i+2 \wedge i \geq M+2 \end{cases}$$

Maximal number of tokens:

$$\max\Big(\max_{(j,k,i)\in T_1} i - Nk - Ni + N^2, \max_{(j,k,i)\in T_2} -k + i^2 - 2Ni + 2i + N^2 - N + 1\Big)$$

$$T_1 = \operatorname{ran} R_2 \cap \{(j,k,i) \mid i = j+1 \wedge i \leq k \leq i+1 \wedge i \geq M+2\}$$
$$T_2 = \operatorname{ran} R_2 \cap \{(j,k,i) \mid i = j+1 \wedge k \geq i+2 \wedge i \geq M+2\}$$

Maximization

Given a communication channel, what is the maximal number of tokens in the channel during the execution?

$$S = \{(j,k,i) \mid 0 \leq j < N \wedge j+1 \leq k < N \wedge j+1 \leq i < N\}$$
$$R = R_1 \cup R_2 \cup R_3$$
$$R_2 = \{((j_r - 1, k_r, i_r), (j_r, k_r, i_r)) \in S \times S \mid M < N \wedge M < i_r\}$$

Tokens in FIFO at (j,k,i) (on $\operatorname{ran} R_2 \cap \{(j,k,i) \mid i = j+1\}$):

$$\begin{cases} i - Nk - Ni + N^2 & \text{if } i \leq k \leq i+1 \wedge i \geq M+2 \\ -k + i^2 - 2Ni + 2i + N^2 - N + 1 & \text{if } k \geq i+2 \wedge i \geq M+2 \end{cases}$$

Maximal number of tokens:

$$\max(\max_{(j,k,i)\in T_1} i - Nk - Ni + N^2, \max_{(j,k,i)\in T_2} -k + i^2 - 2Ni + 2i + N^2 - N + 1)$$
$$T_1 = \operatorname{ran} R_2 \cap \{(j,k,i) \mid i = j+1 \wedge i \leq k \leq i+1 \wedge i \geq M+2\}$$
$$T_2 = \operatorname{ran} R_2 \cap \{(j,k,i) \mid i = j+1 \wedge k \geq i+2 \wedge i \geq M+2\}$$

Maximization

$$\max\left(\max_{(j,k,i)\in T_1} i-Nk-Ni+N^2,\ \max_{(j,k,i)\in T_2} -k+i^2-2Ni+2i+N^2-N+1\right)$$

$$T_1 = \{(j,k,i) \mid i = j+1 \wedge i \leq k \leq i+1 \wedge i \geq M+2 \wedge k \leq N-1\}$$
$$T_2 = \{(j,k,i) \mid i = j+1 \wedge k \geq i+2 \wedge i \geq M+2 \wedge k \leq N-1\}$$

Upper bound on polynomial over parametric polytope

Variables: “inspection” iterators
Parameters: structural parameters

We would like maximum over integer points
Bernstein expansion gives upper bound over rational points

Maximization

$$\max\left(\max_{(j,k,i)\in T_1} i-Nk-Ni+N^2,\ \max_{(j,k,i)\in T_2} -k+i^2-2Ni+2i+N^2-N+1\right)$$

$$T_1 = \{(j,k,i) \mid i = j+1 \wedge i \leq k \leq i+1 \wedge i \geq M+2 \wedge k \leq N-1\}$$
$$T_2 = \{(j,k,i) \mid i = j+1 \wedge k \geq i+2 \wedge i \geq M+2 \wedge k \leq N-1\}$$

Upper bound on polynomial over parametric polytope

Variables: “inspection” iterators

Parameters: structural parameters

We would like maximum over integer points

Bernstein expansion gives upper bound over rational points

Maximization

$$\max\left(\max_{(j,k,i)\in T_1} i-Nk-Ni+N^2,\ \max_{(j,k,i)\in T_2} -k+i^2-2Ni+2i+N^2-N+1\right)$$

$$T_1 = \{(j,k,i) \mid i = j+1 \wedge i \leq k \leq i+1 \wedge i \geq M+2 \wedge k \leq N-1\}$$
$$T_2 = \{(j,k,i) \mid i = j+1 \wedge k \geq i+2 \wedge i \geq M+2 \wedge k \leq N-1\}$$

Upper bound on polynomial over parametric polytope

Variables: “inspection” iterators
Parameters: structural parameters

We would like maximum over integer points
Bernstein expansion gives upper bound over rational points

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis
  Lexicographic maximum of parametric polytope (`PIP`)
  $\Rightarrow$ piecewise quasi-affine map
- counting
  Number of elements in parametric polytope (`barvinok`)
  $\Rightarrow$ piecewise quasi-polynomial
- maximization (computing an upper bound)

  Given a communication channel, what is the maximal number of tokens in the channel during the execution?

Derivation of Process Networks

Problem: what are the channels and how big are they ?

- dataflow analysis
  Lexicographic maximum of parametric polytope (`PIP`)
  $\Rightarrow$ piecewise quasi-affine map
- counting
  Number of elements in parametric polytope (`barvinok`)
  $\Rightarrow$ piecewise quasi-polynomial
- maximization (computing an upper bound)

  > *Given a communication channel, what is the maximal number of tokens in the channel during the execution?*

  Upper bound of parametric polynomial over parametric polytope (`bernstein`)

Bernstein Expansion over an Interval

Write $p(x) \in \mathbb{Q}[x]$ in terms of Bernstein base polynomials

$$p(x) = \sum_{i=0}^{d} a_i x^i = \sum_{i=0}^{d} a_i x^i (x + (1-x))^{d-i} = \sum_{k=0}^{d} b_k^d B_k^d(x)$$

with

$$B_k^d(x) = \binom{d}{k} x^k (1-x)^{d-k}$$

Bernstein coefficients

$$b_k^d = \sum_{i=0}^{k} \frac{\binom{k}{i}}{\binom{d}{i}} a_i$$

Example:

$$p(x) = x^3 - 5x^2 + 2x + 4 = 4B_0^3(x) + \frac{14}{3} B_1^3(x) + \frac{11}{3} B_2^3(x) + 2B_3^3(x)$$

Bernstein Expansion over an Interval

Write $p(x) \in \mathbb{Q}[x]$ in terms of Bernstein base polynomials

$$p(x) = \sum_{i=0}^{d} a_i x^i = \sum_{i=0}^{d} a_i x^i (x + (1-x))^{d-i} = \sum_{k=0}^{d} b_k^d B_k^d(x)$$

with

$$B_k^d(x) = \binom{d}{k} x^k (1-x)^{d-k}$$

Bernstein coefficients

$$b_k^d = \sum_{i=0}^{k} \frac{\binom{k}{i}}{\binom{d}{i}} a_i$$

Example:

$$p(x) = x^3 - 5x^2 + 2x + 4 = 4B_0^3(x) + \frac{14}{3} B_1^3(x) + \frac{11}{3} B_2^3(x) + 2B_3^3(x)$$

Bernstein Expansion over an Interval

Write $p(x) \in \mathbb{Q}[x]$ in terms of Bernstein base polynomials

$$p(x) = \sum_{i=0}^{d} a_i x^i = \sum_{i=0}^{d} a_i x^i (x + (1-x))^{d-i} = \sum_{k=0}^{d} b_k^d B_k^d(x)$$

with

$$B_k^d(x) = \binom{d}{k} x^k (1-x)^{d-k}$$

Bernstein coefficients

$$b_k^d = \sum_{i=0}^{k} \frac{\binom{k}{i}}{\binom{d}{i}} a_i$$

Example:

$$p(x) = x^3 - 5x^2 + 2x + 4 = 4B_0^3(x) + \frac{14}{3} B_1^3(x) + \frac{11}{3} B_2^3(x) + 2B_3^3(x)$$

Bernstein Expansion over an Interval

$$p(x) = \sum_{k=0}^{d} b_k^d B_k^d(x) \quad \text{with} \quad B_k^d(x) = \binom{d}{k} x^k (1-x)^{d-k}$$

Properties of Bernstein base polynomials:

- $1 = (x + (1-x))^d = \sum_{k=0}^{d} B_k^d(x)$
- on interval $[0,1]$: $B_k^d(x) \geq 0$

$\Rightarrow p(x)$ is convex combination of b_k^d on interval $[0,1]$

$$\min_{0 \leq i \leq d} b_i^d \leq p(x) \leq \max_{0 \leq i \leq d} b_i^d \qquad \text{for } x \in [0,1]$$

bound is sharp if min or max of b_i^d is b_0^d or b_d^d

Bernstein Expansion over an Interval

$$p(x) = \sum_{k=0}^{d} b_k^d B_k^d(x) \quad \text{with} \quad B_k^d(x) = \binom{d}{k} x^k (1-x)^{d-k}$$

Properties of Bernstein base polynomials:

- $1 = (x + (1-x))^d = \sum_{k=0}^{d} B_k^d(x)$
- on interval $[0,1]$: $B_k^d(x) \geq 0$

$\Rightarrow p(x)$ is convex combination of b_k^d on interval $[0,1]$

$$\min_{0 \leq i \leq d} b_i^d \leq p(x) \leq \max_{0 \leq i \leq d} b_i^d \qquad \text{for } x \in [0,1]$$

bound is sharp if min or max of b_i^d is b_0^d or b_d^d

Bernstein Expansion over an Interval

$$p(x) = \sum_{k=0}^{d} b_k^d B_k^d(x) \quad \text{with} \quad B_k^d(x) = \binom{d}{k} x^k (1-x)^{d-k}$$

Properties of Bernstein base polynomials:

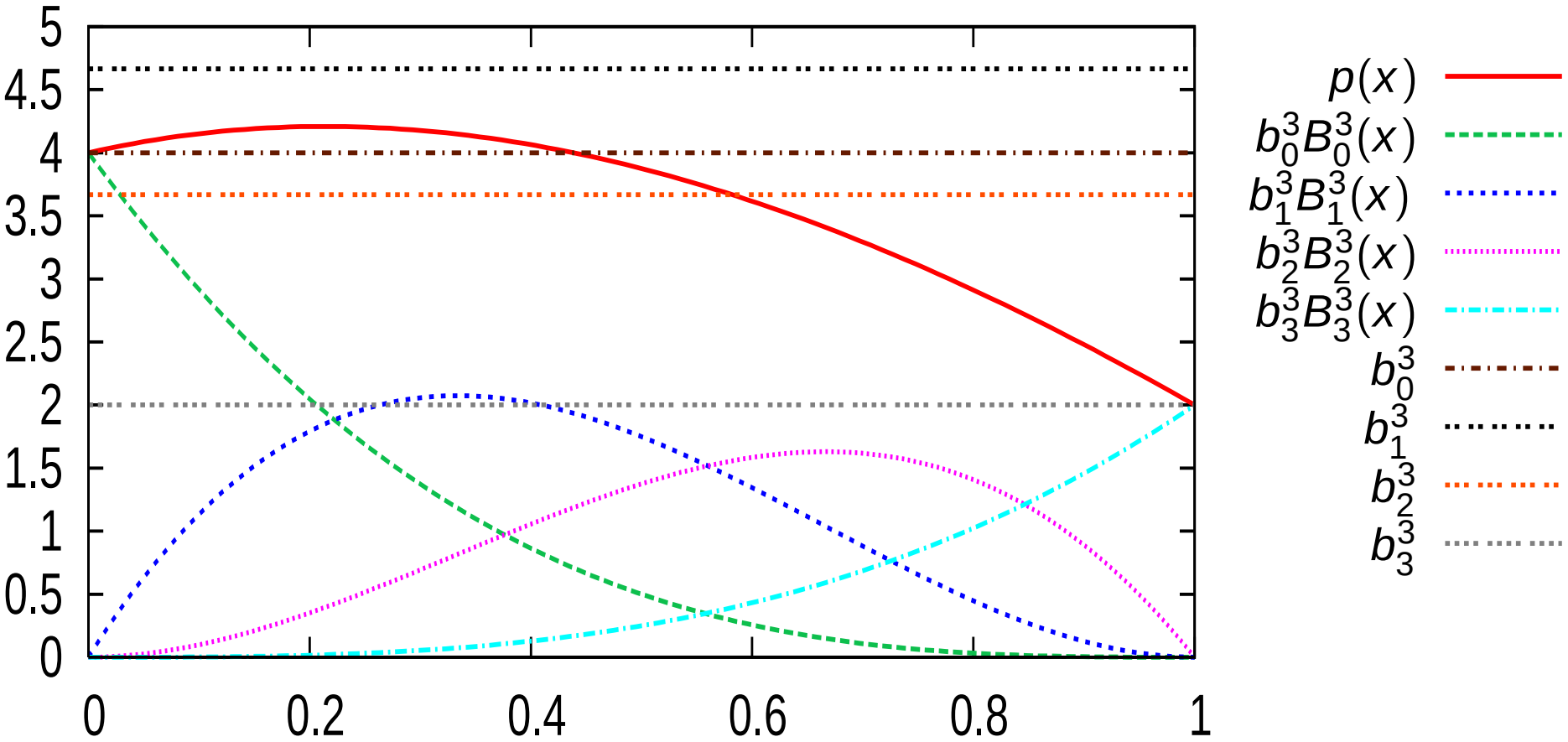
- $1 = (x + (1-x))^d = \sum_{k=0}^{d} B_k^d(x)$
- on interval $[0,1]$: $B_k^d(x) \geq 0$

$\Rightarrow p(x)$ is convex combination of b_k^d on interval $[0,1]$

$$\min_{0 \leq i \leq d} b_i^d \leq p(x) \leq \max_{0 \leq i \leq d} b_i^d \qquad \text{for } x \in [0,1]$$

bound is sharp if min or max of b_i^d is b_0^d or b_d^d

Bernstein Expansion over an Interval



Bernstein Expansion over an Interval: Extensions

- Parametric boxes (Clauss and Tchoupaeva 2004)
  Product Bernstein base polynomials over unit cube $[0,1]^n$
  Linear transformation of parametric boxes
  $[l_1(\mathbf{q}), u_1(\mathbf{q})] \times [l_2(x_1, \mathbf{q}), u_2(x_1, \mathbf{q})] \times \cdots \times$
  $[l_n(x_1, \ldots, x_{n-1}, \mathbf{q}), u_n(x_1, \ldots, x_{n-1}, \mathbf{q})]$

$$y_i = \frac{x_i - a_i(x_1, \ldots, x_{i-1}, \mathbf{q})}{b_i(x_1, \ldots, x_{i-1}, \mathbf{q}) - a_i(x_1, \ldots, x_{i-1}, \mathbf{q})}$$

  - – only works on boxes
  - \+ bounds can be arbitrary polynomials
  - – requires detection and handling of roots of $b_i - a_i$
- General parametric polytopes
  (Clauss, Fernández, Garbervetsky and V. 2009)

Bernstein Expansion over an Interval: Extensions

- Parametric boxes (Clauss and Tchoupaeva 2004)
  Product Bernstein base polynomials over unit cube $[0,1]^n$
  Linear transformation of parametric boxes
  $[l_1(\mathbf{q}), u_1(\mathbf{q})] \times [l_2(x_1, \mathbf{q}), u_2(x_1, \mathbf{q})] \times \cdots \times$
  $[l_n(x_1, \ldots, x_{n-1}, \mathbf{q}), u_n(x_1, \ldots, x_{n-1}, \mathbf{q})]$

$$y_i = \frac{x_i - a_i(x_1, \ldots, x_{i-1}, \mathbf{q})}{b_i(x_1, \ldots, x_{i-1}, \mathbf{q}) - a_i(x_1, \ldots, x_{i-1}, \mathbf{q})}$$

  - $-$ only works on boxes
  - $+$ bounds can be arbitrary polynomials
  - $-$ requires detection and handling of roots of $b_i - a_i$
- General parametric polytopes
  (Clauss, Fernández, Garbervetsky and V. 2009)

Bernstein Expansion over an Interval: Extensions

- Parametric boxes (Clauss and Tchoupaeva 2004)
  Product Bernstein base polynomials over unit cube $[0,1]^n$
  Linear transformation of parametric boxes
  $[l_1(\mathbf{q}), u_1(\mathbf{q})] \times [l_2(x_1, \mathbf{q}), u_2(x_1, \mathbf{q})] \times \cdots \times$
  $[l_n(x_1, \ldots, x_{n-1}, \mathbf{q}), u_n(x_1, \ldots, x_{n-1}, \mathbf{q})]$

$$y_i = \frac{x_i - a_i(x_1, \ldots, x_{i-1}, \mathbf{q})}{b_i(x_1, \ldots, x_{i-1}, \mathbf{q}) - a_i(x_1, \ldots, x_{i-1}, \mathbf{q})}$$

  - $-$ only works on boxes
  - $+$ bounds can be arbitrary polynomials
  - $-$ requires detection and handling of roots of $b_i - a_i$
- General parametric polytopes
  (Clauss, Fernández, Garbervetsky and V. 2009)

Bernstein Expansion over an Interval: Extensions

- Parametric boxes (Clauss and Tchoupaeva 2004)
  Product Bernstein base polynomials over unit cube $[0,1]^n$
  Linear transformation of parametric boxes
  $[l_1(\mathbf{q}), u_1(\mathbf{q})] \times [l_2(x_1, \mathbf{q}), u_2(x_1, \mathbf{q})] \times \cdots \times$
  $[l_n(x_1, \ldots, x_{n-1}, \mathbf{q}), u_n(x_1, \ldots, x_{n-1}, \mathbf{q})]$

$$y_i = \frac{x_i - a_i(x_1, \ldots, x_{i-1}, \mathbf{q})}{b_i(x_1, \ldots, x_{i-1}, \mathbf{q}) - a_i(x_1, \ldots, x_{i-1}, \mathbf{q})}$$

  - $-$ only works on boxes
  - $+$ bounds can be arbitrary polynomials
  - $-$ requires detection and handling of roots of $b_i - a_i$
- General parametric polytopes
  (Clauss, Fernández, Garbervetsky and V. 2009)

Bernstein Expansion over an Interval

$$p(x) = \sum_{k=0}^{d} b_k^d B_k^d(x) \quad \text{with} \quad B_k^d(x) = \binom{d}{k} x^k (1-x)^{d-k}$$

Write

$$x = \alpha_0\, 0 + \alpha_1\, 1$$

with $\alpha_i \geq 0$ and $\alpha_0 + \alpha_1 = 1$
$\Rightarrow (\alpha_0, \alpha_1)$ barycentric coordinates on $[0, 1]$

$$p(\alpha_0, \alpha_1) = p(\alpha_0 v_0 + \alpha_1 v_1) = \sum_{k=0}^{d} b_k^d B_k^d(\alpha_0, \alpha_1)$$

with

$$B_k^d(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k}$$

Bernstein Expansion over an Interval

$$p(x) = \sum_{k=0}^{d} b_k^d B_k^d(x) \quad \text{with} \quad B_k^d(x) = \binom{d}{k} x^k (1-x)^{d-k}$$

Write

$$x = \alpha_0 v_0 + \alpha_1 v_1$$

with $\alpha_i \geq 0$ and $\alpha_0 + \alpha_1 = 1$
$\Rightarrow (\alpha_0, \alpha_1)$ barycentric coordinates on $[v_0, v_1]$

$$p(\alpha_0, \alpha_1) = p(\alpha_0 v_0 + \alpha_1 v_1) = \sum_{k=0}^{d} b_k^d B_k^d(\alpha_0, \alpha_1)$$

with

$$B_k^d(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k}$$

Bernstein Expansion over an Interval

$$p(x) = \sum_{k=0}^{d} b_k^d B_k^d(x) \quad \text{with} \quad B_k^d(x) = \binom{d}{k} x^k (1-x)^{d-k}$$

Write

$$x = \alpha_0 v_0 + \alpha_1 v_1$$

with $\alpha_i \geq 0$ and $\alpha_0 + \alpha_1 = 1$
$\Rightarrow (\alpha_0, \alpha_1)$ barycentric coordinates on $[v_0, v_1]$

$$p(\alpha_0, \alpha_1) = p(\alpha_0 v_0 + \alpha_1 v_1) = \sum_{k=0}^{d} b_k^d B_k^d(\alpha_0, \alpha_1)$$

with

$$B_k^d(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k}$$

Bernstein Expansion

polytope

$$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$$

multivariate polynomial

$$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$$

$$p(\alpha) = p(\sum_i \alpha_i \mathbf{v}_i)$$

$$p(\alpha) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\alpha)$$

Bernstein base polynomials

$$B^d_{\mathbf{k}}(\alpha) = \binom{d}{\mathbf{k}} \alpha^{\mathbf{k}}$$

bounds

$$\min_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

interval

$$\{ \alpha_0\, 0 + \alpha_1\, 1 \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$$

univariate polynomial

$$p(x) \in \mathbb{Q}[x]$$

$$p(\alpha_0, \alpha_1) = p(\alpha_0\, 0 + \alpha_1\, 1)$$

$$p(\alpha_0, \alpha_1) = \sum_{k=0}^{d} b^d_k B^d_k(\alpha_0, \alpha_1)$$

Bernstein base polynomials

$$B^d_k(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k}$$

bounds

$$\min_{0\leq i\leq d} b^d_i \leq p(x) \leq \max_{0\leq i\leq d} b^d_i$$

Bernstein Expansion

polytope

$$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$$

multivariate polynomial

$$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$$

$$p(\alpha) = p(\sum_i \alpha_i \mathbf{v}_i)$$

$$p(\alpha) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\alpha)$$

Bernstein base polynomials

$$B^d_{\mathbf{k}}(\alpha) = \binom{d}{\mathbf{k}} \alpha^{\mathbf{k}}$$

bounds

$$\min_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

interval

$$\{ \alpha_0\, 0 + \alpha_1\, 1 \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$$

univariate polynomial

$$p(x) \in \mathbb{Q}[x]$$

$$p(\alpha_0, \alpha_1) = p(\alpha_0\, 0 + \alpha_1\, 1)$$

$$p(\alpha_0, \alpha_1) = \sum_{k=0}^{d} b^d_k B^d_k(\alpha_0, \alpha_1)$$

Bernstein base polynomials

$$B^d_k(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k}$$

bounds

$$\min_{0\leq i\leq d} b^d_i \leq p(x) \leq \max_{0\leq i\leq d} b^d_i$$

Bernstein Expansion

polytope

$$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$$

multivariate polynomial

$$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$$

$$p(\boldsymbol{\alpha}) = p(\textstyle\sum_i \alpha_i \mathbf{v}_i)$$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein base polynomials

$$B^d_{\mathbf{k}}(\alpha) = \binom{d}{\mathbf{k}} \alpha^{\mathbf{k}}$$

bounds

$$\min_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

interval

$$\{ \alpha_0\, 0 + \alpha_1\, 1 \mid \alpha_i \geq 0, \textstyle\sum_i \alpha_i = 1 \}$$

univariate polynomial

$$p(x) \in \mathbb{Q}[x]$$

$$p(\alpha_0, \alpha_1) = p(\alpha_0\, 0 + \alpha_1\, 1)$$

$$p(\alpha_0, \alpha_1) = \textstyle\sum_{k=0}^{d} b^d_k B^d_k(\alpha_0, \alpha_1)$$

Bernstein base polynomials

$$B^d_k(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k}$$

bounds

$$\min_{0\leq i\leq d} b^d_i \leq p(x) \leq \max_{0\leq i\leq d} b^d_i$$

Generalized Bernstein Base Polynomials

Bernstein base polynomials

$$B_k^d(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k} \quad 1 = (\alpha_1 + \alpha_0)^d = \sum_{k=0}^{d} B_k^d(\alpha_0, \alpha_1)$$

Generalized Bernstein base polynomials

$$1 = (\alpha_1 + \alpha_2 + \cdots + \alpha_n)^d = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} B_{\mathbf{k}}^d(\boldsymbol{\alpha})$$

$$B_{\mathbf{k}}^d(\boldsymbol{\alpha}) = \binom{d}{k_1, k_2, \ldots, k_n} \alpha_1^{k_1} \alpha_2^{k_2} \cdots \alpha_n^{k_n} = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$$

$$\binom{d}{k_1, k_2, \ldots, k_n} = \frac{d!}{k_1! k_2! \ldots k_n!}$$

Generalized Bernstein Base Polynomials

Bernstein base polynomials

$$B_k^d(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k} \quad 1 = (\alpha_1 + \alpha_0)^d = \sum_{k=0}^{d} B_k^d(\alpha_0, \alpha_1)$$

Generalized Bernstein base polynomials

$$1 = (\alpha_1 + \alpha_2 + \cdots + \alpha_n)^d = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} B_{\mathbf{k}}^d(\boldsymbol{\alpha})$$

$$B_{\mathbf{k}}^d(\boldsymbol{\alpha}) = \binom{d}{k_1, k_2, \ldots, k_n} \alpha_1^{k_1} \alpha_2^{k_2} \cdots \alpha_n^{k_n} = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$$

$$\binom{d}{k_1, k_2, \ldots, k_n} = \frac{d!}{k_1! k_2! \ldots k_n!}$$

Bernstein Expansion

polytope

$$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$$

multivariate polynomial

$$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$$

$$p(\boldsymbol{\alpha}) = p(\textstyle\sum_i \alpha_i \mathbf{v}_i)$$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein base polynomials

$$B^d_{\mathbf{k}}(\alpha) = \binom{d}{\mathbf{k}} \alpha^{\mathbf{k}}$$

bounds

$$\min_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

interval

$$\{ \alpha_0\, 0 + \alpha_1\, 1 \mid \alpha_i \geq 0, \textstyle\sum_i \alpha_i = 1 \}$$

univariate polynomial

$$p(x) \in \mathbb{Q}[x]$$

$$p(\alpha_0, \alpha_1) = p(\alpha_0\, 0 + \alpha_1\, 1)$$

$$p(\alpha_0, \alpha_1) = \textstyle\sum_{k=0}^{d} b^d_k B^d_k(\alpha_0, \alpha_1)$$

Bernstein base polynomials

$$B^d_k(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k}$$

bounds

$$\min_{0\leq i\leq d} b^d_i \leq p(x) \leq \max_{0\leq i\leq d} b^d_i$$

Bernstein Expansion

polytope

$$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$$

multivariate polynomial

$$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$$

$$p(\boldsymbol{\alpha}) = p(\textstyle\sum_i \alpha_i \mathbf{v}_i)$$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein base polynomials

$$B^d_{\mathbf{k}}(\boldsymbol{\alpha}) = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$$

bounds

$$\min_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

interval

$$\{ \alpha_0\, 0 + \alpha_1\, 1 \mid \alpha_i \geq 0, \textstyle\sum_i \alpha_i = 1 \}$$

univariate polynomial

$$p(x) \in \mathbb{Q}[x]$$

$$p(\alpha_0, \alpha_1) = p(\alpha_0\, 0 + \alpha_1\, 1)$$

$$p(\alpha_0, \alpha_1) = \textstyle\sum_{k=0}^{d} b^d_k B^d_k(\alpha_0, \alpha_1)$$

$$B^d_k(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k}$$

bounds

$$\min_{0\leq i\leq d} b^d_i \leq p(x) \leq \max_{0\leq i\leq d} b^d_i$$

Bernstein Expansion

polytope

$$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$$

multivariate polynomial

$$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$$

$$p(\boldsymbol{\alpha}) = p(\textstyle\sum_i \alpha_i \mathbf{v}_i)$$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein base polynomials

$$B^d_{\mathbf{k}}(\boldsymbol{\alpha}) = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$$

bounds

$$\min_{\substack{\mathbf{k} \geq \mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k} \geq \mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

interval

$$\{ \alpha_0\, 0 + \alpha_1\, 1 \mid \alpha_i \geq 0, \textstyle\sum_i \alpha_i = 1 \}$$

univariate polynomial

$$p(x) \in \mathbb{Q}[x]$$

$$p(\alpha_0, \alpha_1) = p(\alpha_0\, 0 + \alpha_1\, 1)$$

$$p(\alpha_0, \alpha_1) = \textstyle\sum_{k=0}^{d} b^d_k B^d_k(\alpha_0, \alpha_1)$$

$$B^d_k(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k}$$

$$\min_{0 \leq i \leq d} b^d_i \leq p(x) \leq \max_{0 \leq i \leq d} b^d_i$$

$\mathcal{V}$-parametric polytopes

$P : D \rightarrow \mathbb{Q}^n$:

$$\mathbf{q} \mapsto P(\mathbf{q}) = \left\{ \mathbf{x} \mid \exists \alpha_i \in \mathbb{Q} : \mathbf{x} = \sum_i \alpha_i \mathbf{v}_i(\mathbf{q}), \alpha_i \geq 0, \sum_i \alpha_i = 1 \right\}$$

$D \subset \mathbb{Q}^r$: parameter domain
$\mathbf{v}_i(\mathbf{q}) \in \mathbb{Q}[\mathbf{q}]$ arbitrary polynomials in parameters
$\mathbf{v}_i(\mathbf{q})$ are generators of the polytope

Bernstein Expansion

polytope

$$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$$

multivariate polynomial

$$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$$

$$p(\boldsymbol{\alpha}) = p(\textstyle\sum_i \alpha_i \mathbf{v}_i)$$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein base polynomials

$$B^d_{\mathbf{k}}(\boldsymbol{\alpha}) = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$$

bounds

$$\min_{\substack{\mathbf{k} \geq \mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k} \geq \mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

interval

$$\{ \alpha_0\, 0 + \alpha_1\, 1 \mid \alpha_i \geq 0, \textstyle\sum_i \alpha_i = 1 \}$$

univariate polynomial

$$p(x) \in \mathbb{Q}[x]$$

$$p(\alpha_0, \alpha_1) = p(\alpha_0\, 0 + \alpha_1\, 1)$$

$$p(\alpha_0, \alpha_1) = \textstyle\sum_{k=0}^{d} b^d_k B^d_k(\alpha_0, \alpha_1)$$

$$B^d_k(\alpha_0, \alpha_1) = \binom{d}{k} \alpha_1^k \alpha_0^{d-k}$$

$$\min_{0 \leq i \leq d} b^d_i \leq p(x) \leq \max_{0 \leq i \leq d} b^d_i$$

Bernstein Expansion

polytope

$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$

multivariate polynomial

$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$

$p(\boldsymbol{\alpha}) = p(\sum_i \alpha_i \mathbf{v}_i)$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein base polynomials

$B^d_{\mathbf{k}}(\boldsymbol{\alpha}) = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$

bounds

$$\min_{\substack{\mathbf{k} \geq \mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k} \geq \mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

Bernstein Expansion

polytope

$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$

parametric polytope

$\{ \sum_i \alpha_i \mathbf{v}_i(\mathbf{q}) \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$

multivariate polynomial

$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$

$p(\boldsymbol{\alpha}) = p(\sum_i \alpha_i \mathbf{v}_i)$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein base polynomials

$B^d_{\mathbf{k}}(\boldsymbol{\alpha}) = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$

bounds

$$\min_{\substack{\mathbf{k} \geq \mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k} \geq \mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

Bernstein Expansion

polytope

$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$

multivariate polynomial

$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$

$p(\boldsymbol{\alpha}) = p(\sum_i \alpha_i \mathbf{v}_i)$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein base polynomials

$B^d_{\mathbf{k}}(\boldsymbol{\alpha}) = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$

bounds

$$\min_{\substack{\mathbf{k} \geq \mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k} \geq \mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

parametric polytope

$\{ \sum_i \alpha_i \mathbf{v}_i(\mathbf{q}) \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$

parametric polynomial

$p(\mathbf{x}) \in (\mathbb{Q}[\mathbf{q}])[\mathbf{x}]$

$p(\boldsymbol{\alpha}) = p(\sum_i \alpha_i \mathbf{v}_i(\mathbf{q}))$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}}(\mathbf{q}) B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein Expansion

polytope

$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$

multivariate polynomial

$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$

$p(\boldsymbol{\alpha}) = p(\sum_i \alpha_i \mathbf{v}_i)$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein base polynomials

$$B^d_{\mathbf{k}}(\boldsymbol{\alpha}) = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$$

bounds

$$\min_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

parametric polytope

$\{ \sum_i \alpha_i \mathbf{v}_i(\mathbf{q}) \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$

parametric polynomial

$p(\mathbf{x}) \in (\mathbb{Q}[\mathbf{q}])[\mathbf{x}]$

$p(\boldsymbol{\alpha}) = p(\sum_i \alpha_i \mathbf{v}_i(\mathbf{q}))$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}}(\mathbf{q}) B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

$$B^d_{\mathbf{k}}(\boldsymbol{\alpha}) = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$$

Bernstein Expansion

polytope

$\{ \sum_i \alpha_i \mathbf{v}_i \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$

multivariate polynomial

$p(\mathbf{x}) \in \mathbb{Q}[\mathbf{x}]$

$p(\boldsymbol{\alpha}) = p(\sum_i \alpha_i \mathbf{v}_i)$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}} B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

Bernstein base polynomials

$$B^d_{\mathbf{k}}(\boldsymbol{\alpha}) = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$$

bounds

$$\min_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}} \leq p(\mathbf{x}) \leq \max_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}$$

parametric polytope

$\{ \sum_i \alpha_i \mathbf{v}_i(\mathbf{q}) \mid \alpha_i \geq 0, \sum_i \alpha_i = 1 \}$

parametric polynomial

$p(\mathbf{x}) \in (\mathbb{Q}[\mathbf{q}])[\mathbf{x}]$

$p(\boldsymbol{\alpha}) = p(\sum_i \alpha_i \mathbf{v}_i(\mathbf{q}))$

$$p(\boldsymbol{\alpha}) = \sum_{\substack{k_1,k_2,\ldots,k_n \geq 0 \\ k_1+k_2+\cdots+k_n=d}} b^d_{\mathbf{k}}(\mathbf{q}) B^d_{\mathbf{k}}(\boldsymbol{\alpha})$$

$$B^d_{\mathbf{k}}(\boldsymbol{\alpha}) = \binom{d}{\mathbf{k}} \boldsymbol{\alpha}^{\mathbf{k}}$$

$$\min_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}(\mathbf{q}) \leq p(\mathbf{q})(\mathbf{x}) \leq \max_{\substack{\mathbf{k}\geq\mathbf{0} \\ \sum k_i = d}} b^d_{\mathbf{k}}(\mathbf{q})$$

Parametric Example

$$p(x_1, x_2) = \frac{1}{2}x_1^2 + \frac{1}{2}x_1 + x_2$$

$$P = \text{conv}\{(0,0), (N,0), (N,N)\}$$

To compute:

$$\max_{(x_1,x_2)\in P} p(x_1, x_2)$$

$$(x_1, x_2) = \alpha_1(0,0) + \alpha_2(N,0) + \alpha_3(N,N)$$

$$p(\alpha_1, \alpha_2, \alpha_3) = \frac{1}{2}N^2\alpha_2^2 + N^2\alpha_2\alpha_3 + \frac{1}{2}N^2\alpha_3^2 + \frac{1}{2}N\alpha_2 + \frac{3}{2}N\alpha_3$$

Parametric Example

$$p(x_1, x_2) = \frac{1}{2}x_1^2 + \frac{1}{2}x_1 + x_2$$

$$P = \text{conv}\{(0,0), (N,0), (N,N)\}$$

To compute:

$$\max_{(x_1,x_2)\in P} p(x_1, x_2)$$

$$(x_1, x_2) = \alpha_1(0,0) + \alpha_2(N,0) + \alpha_3(N,N)$$

$$p(\alpha_1, \alpha_2, \alpha_3) = \frac{1}{2}N^2\alpha_2^2 + N^2\alpha_2\alpha_3 + \frac{1}{2}N^2\alpha_3^2 + \frac{1}{2}N\alpha_2 + \frac{3}{2}N\alpha_3$$

Parametric Example

$$p(\alpha_1,\alpha_2,\alpha_3) = \frac{1}{2}N^2\alpha_2^2 + N^2\alpha_2\alpha_3 + \frac{1}{2}N^2\alpha_3^2 + \frac{1}{2}N\alpha_2 + \frac{3}{2}N\alpha_3$$

$$p(\alpha_1,\alpha_2,\alpha_3) = \left(\frac{1}{2}N^2 + \frac{1}{2}N\right)\alpha_2^2 + \left(\frac{1}{2}N^2 + \frac{3}{2}N\right)\alpha_3^2 + \frac{1}{2}N\alpha_1\alpha_2 + \frac{3}{2}N\alpha_1\alpha_3 + (N^2+2N)\alpha_2\alpha_3$$

$$0\,B_{2,0,0} + \left(\frac{1}{2}N^2 + \frac{1}{2}N\right)B_{0,2,0} + \left(\frac{1}{2}N^2 + \frac{3}{2}N\right)B_{0,0,2} + \frac{1}{4}NB_{1,1,0} + \frac{3}{4}NB_{1,0,1} + \left(\frac{1}{2}N^2 + N\right)B_{0,1,1}$$

Parametric Example

$$p(\alpha_1, \alpha_2, \alpha_3) = \frac{1}{2}N^2\alpha_2^2 + N^2\alpha_2\alpha_3 + \frac{1}{2}N^2\alpha_3^2 + \frac{1}{2}N\alpha_2 + \frac{3}{2}N\alpha_3$$

$$p(\alpha_1, \alpha_2, \alpha_3) = \left(\frac{1}{2}N^2 + \frac{1}{2}N\right)\alpha_2^2 + \left(\frac{1}{2}N^2 + \frac{3}{2}N\right)\alpha_3^2 + \frac{1}{2}N\alpha_1\alpha_2 + \frac{3}{2}N\alpha_1\alpha_3 + (N^2 + 2N)\alpha_2\alpha_3$$

$$0\,B_{2,0,0} + \left(\frac{1}{2}N^2 + \frac{1}{2}N\right)B_{0,2,0} + \left(\frac{1}{2}N^2 + \frac{3}{2}N\right)B_{0,0,2} + \frac{1}{4}NB_{1,1,0} + \frac{3}{4}NB_{1,0,1} + \left(\frac{1}{2}N^2 + N\right)B_{0,1,1}$$

Parametric Example

$$p(\alpha_1,\alpha_2,\alpha_3) = \frac{1}{2}N^2\alpha_2^2 + N^2\alpha_2\alpha_3 + \frac{1}{2}N^2\alpha_3^2 + \frac{1}{2}N\alpha_2 + \frac{3}{2}N\alpha_3$$

$$p(\alpha_1,\alpha_2,\alpha_3) = \left(\frac{1}{2}N^2 + \frac{1}{2}N\right)\alpha_2^2 + \left(\frac{1}{2}N^2 + \frac{3}{2}N\right)\alpha_3^2 + \frac{1}{2}N\alpha_1\alpha_2 + \frac{3}{2}N\alpha_1\alpha_3 + (N^2+2N)\alpha_2\alpha_3$$

$$0\,B_{2,0,0} + \left(\frac{1}{2}N^2 + \frac{1}{2}N\right)B_{0,2,0} + \left(\frac{1}{2}N^2 + \frac{3}{2}N\right)B_{0,0,2} + \frac{1}{4}NB_{1,1,0} + \frac{3}{4}NB_{1,0,1} + \left(\frac{1}{2}N^2 + N\right)B_{0,1,1}$$

Parametric Example

$$p(\alpha_1,\alpha_2,\alpha_3) = \frac{1}{2}N^2\alpha_2^2 + N^2\alpha_2\alpha_3 + \frac{1}{2}N^2\alpha_3^2 + \frac{1}{2}N\alpha_2 + \frac{3}{2}N\alpha_3$$

$$p(\alpha_1,\alpha_2,\alpha_3) = \left(\frac{1}{2}N^2 + \frac{1}{2}N\right)\alpha_2^2 + \left(\frac{1}{2}N^2 + \frac{3}{2}N\right)\alpha_3^2 + \frac{1}{2}N\alpha_1\alpha_2 + \frac{3}{2}N\alpha_1\alpha_3 + (N^2+2N)\alpha_2\alpha_3$$

$$0\,B_{2,0,0} + \left(\frac{1}{2}N^2 + \frac{1}{2}N\right)B_{0,2,0} + \left(\frac{1}{2}N^2 + \frac{3}{2}N\right)B_{0,0,2} + \frac{1}{4}NB_{1,1,0} + \frac{3}{4}NB_{1,0,1} + \left(\frac{1}{2}N^2 + N\right)B_{0,1,1}$$

$\mathcal{H}$-parametric polytopes

$$P : \mathbb{Q}^r \rightarrow \mathbb{Q}^n : \mathbf{q} \mapsto P(\mathbf{q}) = \{\mathbf{x} \in \mathbb{Q}^d \mid A\mathbf{x} \geq D\mathbf{q} + \mathbf{d}\}$$

Conversion to $\mathcal{V}$-parametric polytopes:

- Compute parametric vertices
- Compute chamber decomposition

Example:

$$T_1 = \{(j,k,i) \mid i = j+1 \wedge i \leq k \leq i+1 \wedge i \geq M+2 \wedge k \leq N-1\}$$

$$\begin{cases} \text{conv}\left\{ \begin{pmatrix} M+1 \\ N-1 \\ M+2 \end{pmatrix}, \begin{pmatrix} M+1 \\ M+2 \\ M+2 \end{pmatrix}, \begin{pmatrix} N-2 \\ N-2 \\ N-2 \end{pmatrix} \right\} & \text{if } N-4 \leq M \leq N-3 \\ \text{conv}\left\{ \begin{pmatrix} M+1 \\ M+3 \\ M+2 \end{pmatrix} \begin{pmatrix} N-3 \\ N-1 \\ N-2 \end{pmatrix} \begin{pmatrix} M+1 \\ M+2 \\ M+2 \end{pmatrix} \begin{pmatrix} N-2 \\ N-1 \\ N-1 \end{pmatrix} \right\} & \text{if } M \leq N-5 \end{cases}$$

$\mathcal{H}$-parametric polytopes

$$P : \mathbb{Q}^r \rightarrow \mathbb{Q}^n : \mathbf{q} \mapsto P(\mathbf{q}) = \{\mathbf{x} \in \mathbb{Q}^d \mid A\mathbf{x} \geq D\mathbf{q} + \mathbf{d}\}$$

Conversion to $\mathcal{V}$-parametric polytopes:

- Compute parametric vertices
- Compute chamber decomposition

Example:

$$T_1 = \{(j,k,i) \mid i = j+1 \wedge i \leq k \leq i+1 \wedge i \geq M+2 \wedge k \leq N-1\}$$

$$\begin{cases} \operatorname{conv}\left\{ \begin{pmatrix} M+1 \\ N-1 \\ M+2 \end{pmatrix}, \begin{pmatrix} M+1 \\ M+2 \\ M+2 \end{pmatrix}, \begin{pmatrix} N-2 \\ N-2 \\ N-2 \end{pmatrix} \right\} & \text{if } N-4 \leq M \leq N-3 \\ \operatorname{conv}\left\{ \begin{pmatrix} M+1 \\ M+3 \\ M+2 \end{pmatrix} \begin{pmatrix} N-3 \\ N-1 \\ N-2 \end{pmatrix} \begin{pmatrix} M+1 \\ M+2 \\ M+2 \end{pmatrix} \begin{pmatrix} N-2 \\ N-1 \\ N-1 \end{pmatrix} \right\} & \text{if } M \leq N-5 \end{cases}$$

$\mathcal{H}$-parametric polytopes

$$P : \mathbb{Q}^r \rightarrow \mathbb{Q}^n : \mathbf{q} \mapsto P(\mathbf{q}) = \{\mathbf{x} \in \mathbb{Q}^d \mid A\mathbf{x} \geq D\mathbf{q} + \mathbf{d}\}$$

Conversion to $\mathcal{V}$-parametric polytopes:

- Compute parametric vertices
- Compute chamber decomposition

Example:

$$T_1 = \{(j,k,i) \mid i = j+1 \wedge i \leq k \leq i+1 \wedge i \geq M+2 \wedge k \leq N-1\}$$

$$\begin{cases} \text{conv}\left\{ \begin{pmatrix} M+1 \\ N-1 \\ M+2 \end{pmatrix}, \begin{pmatrix} M+1 \\ M+2 \\ M+2 \end{pmatrix}, \begin{pmatrix} N-2 \\ N-2 \\ N-2 \end{pmatrix} \right\} & \text{if } N-4 \leq M \leq N-3 \\ \text{conv}\left\{ \begin{pmatrix} M+1 \\ M+3 \\ M+2 \end{pmatrix} \begin{pmatrix} N-3 \\ N-1 \\ N-2 \end{pmatrix} \begin{pmatrix} M+1 \\ M+2 \\ M+2 \end{pmatrix} \begin{pmatrix} N-2 \\ N-1 \\ N-1 \end{pmatrix} \right\} & \text{if } M \leq N-5 \end{cases}$$

FIFO Size Example

$$\max\left(\max_{(j,k,i)\in T_1} i-Nk-Ni+N^2,\ \max_{(j,k,i)\in T_2} -k+i^2-2Ni+2i+N^2-N+1\right)$$

$$T_1=\{(j,k,i)\mid i=j+1\wedge i\le k\le i+1\wedge i\ge M+2\wedge k\le N-1\}$$
$$T_2=\{(j,k,i)\mid i=j+1\wedge k\ge i+2\wedge i\ge M+2\wedge k\le N-1\}$$

$\max_{(j,k,i)\in T_1} i-Nk-Ni+N^2$:

$$\begin{cases}\max(M+N^2-4N-2MN+2,\\ \quad -N^2+3N-1,\\ \quad M-N-MN+2) & \text{if } N-4\le M\le N-3\\ \max(M+N^2-5N-2MN+2,\\ \quad M+N^2-4N-2MN+2,\\ \quad -N^2+4N+2,\\ \quad -N^2+3N-1) & \text{if } M\le N-5\end{cases}$$

FIFO Size Example

$$\max\left(\max_{(j,k,i)\in T_1} i-Nk-Ni+N^2,\ \max_{(j,k,i)\in T_2} -k+i^2-2Ni+2i+N^2-N+1\right)$$

$$T_1=\{(j,k,i)\mid i=j+1\wedge i\le k\le i+1\wedge i\ge M+2\wedge k\le N-1\}$$
$$T_2=\{(j,k,i)\mid i=j+1\wedge k\ge i+2\wedge i\ge M+2\wedge k\le N-1\}$$

$\max_{(j,k,i)\in T_1} i-Nk-Ni+N^2$:

$$\begin{cases}\max(M+N^2-4N-2MN+2,\\ \quad -N^2+3N-1,\\ \quad M-N-MN+2) & \text{if } N-4\le M\le N-3\\ \max(M+N^2-5N-2MN+2,\\ \quad M+N^2-4N-2MN+2,\\ \quad -N^2+4N+2,\\ \quad -N^2+3N-1) & \text{if } M\le N-5\end{cases}$$

Simplification

Some polynomials $p_i(\mathbf{q})$ in max-expression may be redundant
$\Rightarrow$ drop $p_i(\mathbf{q})$ if $p_i(\mathbf{q}) - p_j(\mathbf{q}) \leq 0$ $(i \neq j)$ on their domain

Easy ways of determining sign of $d(\mathbf{q}) = p_i(\mathbf{q}) - p_j(\mathbf{q})$

- $d(\mathbf{q}) = D$, a constant
- $d(\mathbf{q})$ is degree-1 polynomial (add constraint to domain)
- domain is bounded $\Rightarrow$ apply Bernstein expansion
  $(M+1)N^2 - 4N - 2(M+1) = (N^2 + 2N - 1) =$
  $3+M = 2N^2 - 7N - 2MN$ on $10 \leq M \leq N = 5 \leq 95$
  Bernstein coefficients: 116, 8756, 17813, 199, 27, 1274
  univariate $\Rightarrow$ Cauchy's bounds
- lower bound on parameter
  $\Rightarrow$ check coefficients of Taylor expansion about lower bound

Simplification

Some polynomials $p_i(\mathbf{q})$ in max-expression may be redundant
$\Rightarrow$ drop $p_i(\mathbf{q})$ if $p_i(\mathbf{q}) - p_j(\mathbf{q}) \leq 0$ $(i \neq j)$ on their domain

Easy ways of determining sign of $d(\mathbf{q}) = p_i(\mathbf{q}) - p_j(\mathbf{q})$

- $d(\mathbf{q}) = D$, a constant
- $d(\mathbf{q})$ is degree-1 polynomial (add constraint to domain)
- domain is bounded $\Rightarrow$ apply Bernstein expansion
  $(M + N^2 - 4N - 2MN + 2) - (-N^2 + 3N - 1) =$
  $3 + M + 2N^2 - 7N - 2MN$ on $10 \leq M \leq N - 5 \leq 95$
  Bernstein coefficients: $113, 8756, 17313, 199, 27, 1274$
  univariate $\Rightarrow$ Cauchy's bounds
- lower bound on parameter
  $\Rightarrow$ check coefficients of Taylor expansion about lower bound

Simplification

Some polynomials $p_i(\mathbf{q})$ in max-expression may be redundant
$\Rightarrow$ drop $p_i(\mathbf{q})$ if $p_i(\mathbf{q}) - p_j(\mathbf{q}) \leq 0$ $(i \neq j)$ on their domain

Easy ways of determining sign of $d(\mathbf{q}) = p_i(\mathbf{q}) - p_j(\mathbf{q})$

- $d(\mathbf{q}) = D$, a constant
- $d(\mathbf{q})$ is degree-1 polynomial (add constraint to domain)
- domain is bounded $\Rightarrow$ apply Bernstein expansion
  $(M + N^2 - 4N - 2MN + 2) - (-N^2 + 3N - 1) =$
  $3 + M + 2N^2 - 7N - 2MN$ on $10 \leq M \leq N - 5 \leq 95$
  Bernstein coefficients: $113, 8756, 17313, 199, 27, 1274$
  univariate $\Rightarrow$ Cauchy's bounds
- lower bound on parameter
  $\Rightarrow$ check coefficients of Taylor expansion about lower bound

Simplification

Some polynomials $p_i(\mathbf{q})$ in max-expression may be redundant
$\Rightarrow$ drop $p_i(\mathbf{q})$ if $p_i(\mathbf{q}) - p_j(\mathbf{q}) \leq 0$ $(i \neq j)$ on their domain

Easy ways of determining sign of $d(\mathbf{q}) = p_i(\mathbf{q}) - p_j(\mathbf{q})$

- $d(\mathbf{q}) = D$, a constant
- $d(\mathbf{q})$ is degree-1 polynomial (add constraint to domain)
- domain is bounded $\Rightarrow$ apply Bernstein expansion
  $(M + N^2 - 4N - 2MN + 2) - (-N^2 + 3N - 1) =$
  $3 + M + 2N^2 - 7N - 2MN$ on $10 \leq M \leq N - 5 \leq 95$
  Bernstein coefficients: $113, 8756, 17313, 199, 27, 1274$
  univariate $\Rightarrow$ Cauchy's bounds
- lower bound on parameter
  $\Rightarrow$ check coefficients of Taylor expansion about lower bound

Simplification

Some polynomials $p_i(\mathbf{q})$ in max-expression may be redundant
$\Rightarrow$ drop $p_i(\mathbf{q})$ if $p_i(\mathbf{q}) - p_j(\mathbf{q}) \leq 0$ $(i \neq j)$ on their domain

Easy ways of determining sign of $d(\mathbf{q}) = p_i(\mathbf{q}) - p_j(\mathbf{q})$

- $d(\mathbf{q}) = D$, a constant
- $d(\mathbf{q})$ is degree-1 polynomial (add constraint to domain)
- domain is bounded $\Rightarrow$ apply Bernstein expansion
  $(M + N^2 - 4N - 2MN + 2) - (-N^2 + 3N - 1) =$
  $3 + M + 2N^2 - 7N - 2MN$ on $10 \leq M \leq N - 5 \leq 95$
  Bernstein coefficients: $113, 8756, 17313, 199, 27, 1274$
  univariate $\Rightarrow$ Cauchy's bounds
- lower bound on parameter
  $\Rightarrow$ check coefficients of Taylor expansion about lower bound

Simplification

Some polynomials $p_i(\mathbf{q})$ in max-expression may be redundant
$\Rightarrow$ drop $p_i(\mathbf{q})$ if $p_i(\mathbf{q}) - p_j(\mathbf{q}) \leq 0$ $(i \neq j)$ on their domain

Easy ways of determining sign of $d(\mathbf{q}) = p_i(\mathbf{q}) - p_j(\mathbf{q})$

- $d(\mathbf{q}) = D$, a constant
- $d(\mathbf{q})$ is degree-1 polynomial (add constraint to domain)
- domain is bounded $\Rightarrow$ apply Bernstein expansion
  $(M + N^2 - 4N - 2MN + 2) - (-N^2 + 3N - 1) =$
  $3 + M + 2N^2 - 7N - 2MN$ on $10 \leq M \leq N - 5 \leq 95$
  Bernstein coefficients: $113, 8756, 17313, 199, 27, 1274$
  univariate $\Rightarrow$ Cauchy's bounds
- lower bound on parameter
  $\Rightarrow$ check coefficients of Taylor expansion about lower bound

Simplification

$\max_{(j,k,i)\in T_1} i - Nk - Ni + N^2$:

$$\begin{cases} \max(M + N^2 - 4N - 2MN + 2, \\ \qquad -N^2 + 3N - 1, \\ \qquad M - N - MN + 2) & \text{if } N - 4 \leq M \leq N - 3 \\ \max(M + N^2 - 5N - 2MN + 2, \\ \qquad M + N^2 - 4N - 2MN + 2, \\ \qquad -N^2 + 4N + 2, \\ \qquad -N^2 + 3N - 1) & \text{if } M \leq N - 5 \end{cases}$$

Context: $10 \leq M \leq 100 \wedge 10 \leq N \leq 100$

$$\begin{cases} M + N^2 - 4N - 2MN + 2, & \text{if } N - 4 \leq M \leq N - 3 \\ M + N^2 - 4N - 2MN + 2, & \text{if } M \leq N - 5 \end{cases}$$

Simplification

$\max_{(j,k,i)\in T_1} i - Nk - Ni + N^2$:

$$\begin{cases} \max(M + N^2 - 4N - 2MN + 2, \\ \qquad -N^2 + 3N - 1, & \text{if } N - 4 \leq M \leq N - 3 \\ \qquad M - N - MN + 2) \\ \max(M + N^2 - 5N - 2MN + 2, \\ \qquad M + N^2 - 4N - 2MN + 2, \\ \qquad -N^2 + 4N + 2, & \text{if } M \leq N - 5 \\ \qquad -N^2 + 3N - 1) \end{cases}$$

Context: $10 \leq M \leq 100 \wedge 10 \leq N \leq 100$

$$\begin{cases} M + N^2 - 4N - 2MN + 2, & \text{if } N - 4 \leq M \leq N - 3 \\ M + N^2 - 4N - 2MN + 2, & \text{if } M \leq N - 5 \end{cases}$$

Bernstein Expansion over an Interval: Extensions

- Parametric boxes (Clauss and Tchoupaeva 2004)
- General parametric polytopes
  (Clauss, Fernández, Garbervetsky and V. 2009)

Bernstein Expansion over an Interval: Extensions

- Parametric boxes (Clauss and Tchoupaeva 2004)
- General parametric polytopes
  (Clauss, Fernández, Garbervetsky and V. 2009)
- Incremental Computation
  - Parametric boxes
  - $\mathcal{H}$-parametric polytopes

  (Clauss, Fernández, Garbervetsky and V. 2009)

Bernstein Expansion over an Interval: Extensions

- Parametric boxes (Clauss and Tchoupaeva 2004)
- General parametric polytopes
  (Clauss, Fernández, Garbervetsky and V. 2009)
- Incremental Computation
  - Parametric boxes
  - $\mathcal{H}$-parametric polytopes

  (Clauss, Fernández, Garbervetsky and V. 2009)

Incremental Computation

- Parametric boxes
  $[l_1(\mathbf{q}), u_1(\mathbf{q})] \times [l_2(x_1, \mathbf{q}), u_2(x_1, \mathbf{q})] \times \cdots \times$
  $[l_n(x_1, \ldots, x_{n-1}, \mathbf{q}), u_n(x_1, \ldots, x_{n-1}, \mathbf{q})]$

  $\Rightarrow$ symbolic Bernstein expansion on last coordinate

  $$\min_{0 \leq i \leq d} b_i^d(\hat{\mathbf{x}}, \mathbf{q}) \leq p(\hat{\mathbf{x}}, \mathbf{q})(x_n) \leq \max_{0 \leq i \leq d} b_i^d(\hat{\mathbf{x}}, \mathbf{q})$$

  $\hat{\mathbf{x}} = (x_1, \ldots, x_{n-1})$

  $\Rightarrow$ apply recursively on each coefficient $b_i^d(\hat{\mathbf{x}}, \mathbf{q})$ over box
  $[l_1(\mathbf{q}), u_1(\mathbf{q})] \times [l_2(x_1, \mathbf{q}), u_2(x_1, \mathbf{q})] \times \cdots \times$
  $[l_{n-1}(x_1, \ldots, x_{n-2}, \mathbf{q}), u_{n-1}(x_1, \ldots, x_{n-2}, \mathbf{q})]$

  Note: no problems with roots of polynomials

Incremental Computation

- Parametric boxes
  $[l_1(\mathbf{q}), u_1(\mathbf{q})] \times [l_2(x_1, \mathbf{q}), u_2(x_1, \mathbf{q})] \times \cdots \times$
  $[l_n(x_1, \ldots, x_{n-1}, \mathbf{q}), u_n(x_1, \ldots, x_{n-1}, \mathbf{q})]$

  $\Rightarrow$ symbolic Bernstein expansion on last coordinate

$$\min_{0 \leq i \leq d} b_i^d(\hat{\mathbf{x}}, \mathbf{q}) \leq p(\hat{\mathbf{x}}, \mathbf{q})(x_n) \leq \max_{0 \leq i \leq d} b_i^d(\hat{\mathbf{x}}, \mathbf{q})$$

  $\hat{\mathbf{x}} = (x_1, \ldots, x_{n-1})$

  $\Rightarrow$ apply recursively on each coefficient $b_i^d(\hat{\mathbf{x}}, \mathbf{q})$ over box
  $[l_1(\mathbf{q}), u_1(\mathbf{q})] \times [l_2(x_1, \mathbf{q}), u_2(x_1, \mathbf{q})] \times \cdots \times$
  $[l_{n-1}(x_1, \ldots, x_{n-2}, \mathbf{q}), u_{n-1}(x_1, \ldots, x_{n-2}, \mathbf{q})]$

  Note: no problems with roots of polynomials

Incremental Computation

- Parametric boxes
  $[l_1(\mathbf{q}), u_1(\mathbf{q})] \times [l_2(x_1, \mathbf{q}), u_2(x_1, \mathbf{q})] \times \cdots \times$
  $[l_n(x_1, \ldots, x_{n-1}, \mathbf{q}), u_n(x_1, \ldots, x_{n-1}, \mathbf{q})]$

  $\Rightarrow$ symbolic Bernstein expansion on last coordinate

$$\min_{0 \le i \le d} b_i^d(\hat{\mathbf{x}}, \mathbf{q}) \le p(\hat{\mathbf{x}}, \mathbf{q})(x_n) \le \max_{0 \le i \le d} b_i^d(\hat{\mathbf{x}}, \mathbf{q})$$

  $\hat{\mathbf{x}} = (x_1, \ldots, x_{n-1})$

  $\Rightarrow$ apply recursively on each coefficient $b_i^d(\hat{\mathbf{x}}, \mathbf{q})$ over box
  $[l_1(\mathbf{q}), u_1(\mathbf{q})] \times [l_2(x_1, \mathbf{q}), u_2(x_1, \mathbf{q})] \times \cdots \times$
  $[l_{n-1}(x_1, \ldots, x_{n-2}, \mathbf{q}), u_{n-1}(x_1, \ldots, x_{n-2}, \mathbf{q})]$

  Note: no problems with roots of polynomials

Incremental Computation

- $\mathcal{H}$-parametric polytopes
  - Consider all but one variable to be parameters
  - Compute parametric vertices (of interval)
  - Compute chamber decomposition
  - Compute Bernstein coefficients
  - Apply recursively on each coefficient over each chamber

Incremental Computation: Comparison

Number of Bernstein coefficients
= number of multinomial coefficients $\binom{d}{k_1,k_2,\ldots,k_n}$
= $\binom{d+n-1}{d} > \frac{n^d}{d!}$
d: degree of polynomial; n: number of vertices

Incremental computation
$\Rightarrow$ 2 vertices (for each level, polynomial and chamber)
$\Rightarrow d + 1$ Bernstein coefficients

But: multiple levels and polynomials + maybe more chambers
But: may be more accurate

Also:
redundant coefficients may be removed at intermediate levels

Incremental Computation: Comparison

Number of Bernstein coefficients
= number of multinomial coefficients $\binom{d}{k_1,k_2,\ldots,k_n}$
= $\binom{d+n-1}{d} > \frac{n^d}{d!}$
d: degree of polynomial; n: number of vertices

Incremental computation
$\Rightarrow$ 2 vertices (for each level, polynomial and chamber)
$\Rightarrow d + 1$ Bernstein coefficients

But: multiple levels and polynomials + maybe more chambers
But: may be more accurate

Also:
redundant coefficients may be removed at intermediate levels

Incremental Computation: Comparison

Number of Bernstein coefficients
= number of multinomial coefficients $\binom{d}{k_1,k_2,\ldots,k_n}$
= $\binom{d+n-1}{d} > \frac{n^d}{d!}$
d: degree of polynomial; n: number of vertices

Incremental computation
$\Rightarrow$ 2 vertices (for each level, polynomial and chamber)
$\Rightarrow d + 1$ Bernstein coefficients

But: multiple levels and polynomials + maybe more chambers
But: may be more accurate

Also:
redundant coefficients may be removed at intermediate levels

Conclusion and Future Work

Conclusion

- Derivation of Process Networks requires
  - Parametric Integer Programming (lexmax)
  - Enumeration of parametric polytopes
  - Bounds on polynomial over parametric polytopes
- Bernstein expansion yields the required bounds

Future work

- Bounds on *quasi*-polynomials
- Further simplifications

`bernstein` library included in `barvinok`, available from `http://freshmeat.net/projects/barvinok/`

Conclusion and Future Work

Conclusion

- Derivation of Process Networks requires
  - Parametric Integer Programming (lexmax)
  - Enumeration of parametric polytopes
  - Bounds on polynomial over parametric polytopes
- Bernstein expansion yields the required bounds

Future work

- Bounds on *quasi*-polynomials
- Further simplifications

`bernstein` library included in `barvinok`, available from `http://freshmeat.net/projects/barvinok/`

Conclusion and Future Work

Conclusion

- Derivation of Process Networks requires
  - Parametric Integer Programming (lexmax)
  - Enumeration of parametric polytopes
  - Bounds on polynomial over parametric polytopes
- Bernstein expansion yields the required bounds

Future work

- Bounds on *quasi*-polynomials
- Further simplifications

`bernstein` library included in `barvinok`, available from `http://freshmeat.net/projects/barvinok/`