Symbolic Data Structure Based on Intervals for Parametric Verification

Petr Matoušek

matousp@fit.vutbr.cz

Brno University of Technology, Czech republic

1. Introduction

❖ Talk Outline

1. Parametric Reasoning — Overview
2. Symbolic Data Structures — DBMs, PDBMs, Polyhedra
3. Parametric Hypercubes
4. Operation on Parametric Hypercubes
5. Conclusion

Parametric Analysis — Overview

❖ Parametric Analysis

- analysis of timed systems with parameters
- parameters—special variables that are not changed during the execution, e.g. $x < MAXDELAY$
- may range over infinite domains, e.g. integers
- related by a set of parameters, e.g. $MAXDELAY > 0$

❖ Parametric Reasoning

- verifies that the system satisfies a property for all possible values of the parameters (parametric verification)
- finds constraints on the parameters that satisfies the property (parameter synthesis)

Parametric Analysis — Overview

❖ Glossary

guard – a constraint (condition) on the variables;

it guards transition t so that the transition cannot occur unless the condition on the variables is satisfied; for example, $x < 5 \ \wedge \ y = t + 2$

invariant – a constraint over a state

it has to be satisfied while activity is in the state; for example $x \leq 12$

parametric verification – verification on a model where clocks/counters can be compared with parameters;

during analysis and verification parameters are considered to be special symbols

symbolic data structure – data structure that represents data symbolically (by formulas) instead of explicitly (by enumeration)

it is used to represent infinite sets of configurations; for example,

$(x_1 - x_2 < 2) \wedge (0 < x_2 \leq 2) \wedge (1 \leq x_1)$

Parametric Analysis — Overview

R.Alur, T.Henzinger, M.Y.Vardi (1993):

- **Verification of parameterized systems is generally undecidable.**

A.Annichini, E.Asarin, A.Bouajjani (2000):

- **A semi-algorithmical approach for parametric verification.**
- **A new data structure - Parametric Difference Bound Matrices (PDBMs)**

- **Efficiency of the verification depends on symbolic data structures:**
  - **Difference Bound Matrix (DBM) - D.Dill, 1989**
  - **Polyhedra (for hybrid automata) - T.Henzinger et al., 1997**
  - **Parameteric DBM (PDBM) - A.Annichine et al., 2000**
  - **Parametric Hypercube (pHCube) - P.Matoušek, 2004**

Symbolic Data Structures

❖ Difference Bound Matrix (DBM)

- **represents clock zones for timed automata**
- **contraints:** $\boxed{x_i - x_j \prec c}$ **where** $x_i, x_j \in X, \prec \in \{<, \leq\}, c \in \mathbb{Z}$
- **each entry** $\mathcal{D}_{i,j}$ **has the form** $(d_{i,j}, \prec_{i,j})$
- **if no bound is defined then** $\mathcal{D}_{i,j} = (\infty, <)$
- **a special variable** x_0 **is always** 0

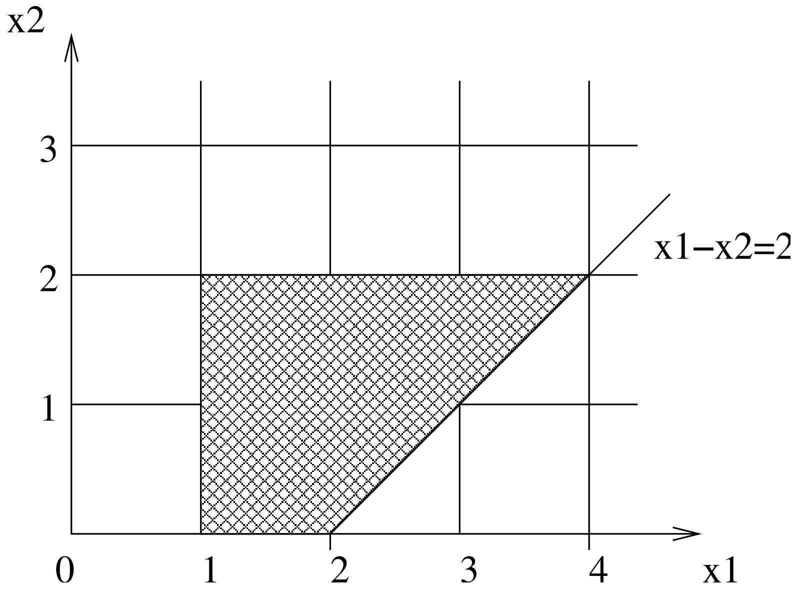
❖ Example: $\varphi = (x_1 - x_2 < 2) \wedge (0 < x_2 \leq 2) \wedge (1 \leq x_1)$

$$\begin{vmatrix} (x_0 - x_0 \leq 0) & (\mathbf{x_0} - \mathbf{x_1} \leq -\mathbf{1}) & (\mathbf{x_0} - \mathbf{x_2} < \mathbf{0}) \\ (x_1 - x_0 < \infty) & (x_1 - x_1 \leq 0) & (\mathbf{x_1} - \mathbf{x_2} < \mathbf{2}) \\ (\mathbf{x_2} - \mathbf{x_0} \leq \mathbf{2}) & (x_2 - x_1 < \infty) & (x_2 - x_2 \leq 0) \end{vmatrix}$$

Symbolic Data Structures

- **Difference Bound Matrix (DBM)**
- **Example:** $\varphi = (x_1 - x_2 < 2) \wedge (0 < x_2 \leq 2) \wedge (1 \leq x_1)$

That clock zone can be represented by the matrix $\mathcal{D}$:



$$\mathcal{D} = \left| \begin{array}{cccc} & x_0 & x_1 & x_2 \\ x_0 & (0, \leq) & (-\mathbf{1}, \leq) & (\mathbf{0}, <) \\ x_1 & (\infty, <) & (0, \leq) & (\mathbf{2}, <) \\ x_2 & (\mathbf{2}, \leq) & (\infty, <) & (0, \leq) \end{array} \right|$$

Symbolic Data Structures

❖ Difference Bound Matrix (DBM) — operations

- **intersection $\mathcal{D} = \mathcal{D}^1 \wedge \mathcal{D}^2$, adding a constraint $\mathcal{D}_{i,j} = (min(c_1, c_2), \prec)$ where $min(c_1, c_2)$ is defined as follows:**
  - **If $c_1 < c_2$ then $' \prec' =' \prec'_1$.**
  - **If $c_2 < c_1$ then $' \prec' =' \prec'_2$.**
  - **If $c_1 = c_2$ and $' \prec'_1 =' \prec'_2$, then $' \prec' =' \prec'_1$.**
  - **If $c_1 = c_2$ and $' \prec'_2 \neq' \prec'_2$, then $' \prec' =' <'$.**
- **elapsing of time $\mathcal{D}' = \mathcal{D} \Uparrow$, removing the upper bounds on clocks**
  - **$\mathcal{D}'_{i,0} = (\infty, <)$ for any $i \neq 0$.**
  - **$\mathcal{D}'_{i,j} = \mathcal{D}_{i,j}$ if $i = 0$ or $j \neq 0$.**

Symbolic Data Structures

❖ Difference Bound Matrix (DBM) — operations

- **clock reset $\mathcal{D}' = \mathcal{D}[\lambda := 0]$, set clocks to zero**
  - **If $x_i, x_j \in \lambda$ then $D'_{i,j} = (0, \leq)$.**
  - **If $x_i \in \lambda, x_j \notin \lambda$ then $\mathcal{D}'_{i,j} = \mathcal{D}_{0,j}$.**
  - **If $x_j \in \lambda, x_i \notin \lambda$ then $\mathcal{D}'_{i,j} = \mathcal{D}_{i,0}$.**
  - **If $x_i, x_j \notin \lambda$ then $\mathcal{D}'_{i,j} = \mathcal{D}_{i,j}$.**
- **inclusion test**
- **normalization**

Symbolic Data Structures

❖ Parametric DBM (PDBM)

- **represents clocks and counters for parameterized timed automata**
- **contraints:** $\boxed{x_i - x_j \prec t}$ **where** $x_i, x_j \in X, \prec \in \{<, \leq\}$
- t **is a term given by** $t ::= c \mid p \mid t - t \mid t + t \mid c * t$**,** $p \in \mathcal{P}$**,** $c \in \mathbb{Z}$
- **each entry** $\mathcal{M}_{i,j}$ **has the form** $(t_{i,j}, \prec_{i,j})$
- **if no bound is defined then** $\mathcal{M}_{i,j} = (\infty, <)$
- **a special variable** x_0 **is always** 0

❖ **Example:** $\mathcal{M} = (y >= 0) \wedge (x >= 0) \wedge (y \leq Max) \wedge (y - x \leq 0) \wedge (x \leq 10 + Max) \wedge (x - y) \leq 10$

Symbolic Data Structures

❖ Constraint Parametric DBM (PDBM)

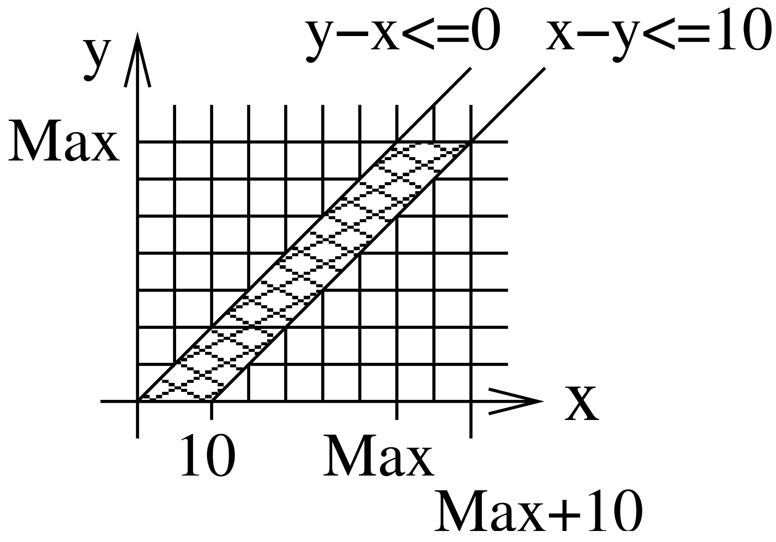
- **a constrained PDBM $\tilde{\mathcal{M}} = (\mathcal{M}, \Phi)$ where $\mathcal{M}$ is a PDBM and $\Phi \in F(\mathcal{P})$ is a parameter constraint**
- **$F(\mathcal{P})$ is a formula given by $\varphi ::= true \mid t \leq t \mid \neg\varphi \mid \varphi \wedge \varphi$**
- **contraints: $\boxed{x_i - x_j \prec t}$ where $x_i, x_j \in X, \prec \in \{<, \leq\}$**

❖ Example: $\tilde{\mathcal{M}} = ((y >= 0) \wedge (x >= 0) \wedge (y \leq Max) \wedge (y - x \leq 0) \wedge (x \leq 10 + Max) \wedge (x - y) \leq 10), Max > 0)$

Symbolic Data Structures

- **Example:** $\tilde{\mathcal{M}} = ((y >= 0) \wedge (x >= 0) \wedge (y \leq Max) \wedge (y - x \leq 0) \wedge (x \leq 10 + Max) \wedge (x - y) \leq 10), Max > 0)$

$$(M_5, \varphi_5) = \begin{pmatrix} & x_0 & x & y \\ x_0 & (0, \leq) & (0, \leq) & (0, \leq) \\ x & (10 + Max, \leq) & (0, \leq) & (10, \leq) \\ y & (100, \leq) & (0, \leq) & (0, \leq) \end{pmatrix}, -Max < 0$$



Symbolic Data Structures

- **Constrained parameterized bound $\tilde{\mathcal{PB}}$ has form $((t, \prec), \varphi)$**
  - **Example:** $b(\tilde{x}) = ((<, 2 * p - q), p \leq 3 + q)$
    semantics: $x < 2 * p - q \,\wedge\, p \leq 3 + q$
- **Total order over parameterized bounds $\subseteq^{\mathcal{PB}}$**
  - **Let** $\tilde{b_1} = ((t_1, \prec_1), \varphi_1),\ \tilde{b_2} = ((t_2, \prec_2), \varphi_2)$ **are two constrained parameterized bounds**
  - $\tilde{b_1} \subseteq^{\mathcal{PB}} \tilde{b_2}$ **iff** φ_{incl} **is satisfiable:**

$$\varphi_{incl} = \forall p_i \in \mathcal{P} \,.\, \varphi_1 \Rightarrow \varphi_2 \,\wedge\, ((t_1 < t_2) \,\vee\, (t_1 = t_2 \wedge \prec_1 \leq \prec_2))$$

  - **it means non-formally** $(\varphi_1 \subseteq \varphi_2)$ **and** $(t_1 \subseteq t_2)$

Symbolic Data Structures

❖ **Operator** $\otimes : \tilde{\mathcal{PB}} \times \tilde{\mathcal{PB}} \rightarrow 2^{\tilde{\mathcal{PB}}}$**—minimum on parametrized bounds**

- **formulas on constraints:**

$$\begin{aligned}
\Phi_< &\equiv \exists p \in \mathcal{P}.\varphi_1 \wedge \varphi_2 \wedge t_1 < t_2 \\
\Phi_= &\equiv \exists p \in \mathcal{P}.\varphi_1 \wedge \varphi_2 \wedge t_1 = t_2 \\
\Phi_> &\equiv \exists p \in \mathcal{P}.\varphi_1 \wedge \varphi_2 \wedge t_1 > t_2
\end{aligned}$$

- **minimum:**

$$\begin{aligned}
\tilde{b_1} \otimes \tilde{b_2} = min(\tilde{b_1}, \tilde{b_2}) &= min_{\leq}(\tilde{b_1}, \tilde{b_2}, \Phi_<) \\
&\cup \; min_=(\tilde{b_1}, \tilde{b_2}, \Phi_=) \\
&\cup \; min_>(\tilde{b_1}, \tilde{b_2}, \Phi_>)
\end{aligned}$$

Symbolic Data Structures

- **Operator $\otimes : \tilde{\mathcal{PB}} \times \tilde{\mathcal{PB}} \rightarrow 2^{\tilde{\mathcal{PB}}}$—minimum on parameterized bounds**

where

$$\textit{min}_<(\tilde{b}_1, \tilde{b}_2, \Phi_<) = \begin{cases} \{((t_1, \prec_1), \varphi_1 \wedge \varphi_2 \wedge (t_1 < t_2))\} & \textbf{if } \Phi_< \\ \emptyset & \textbf{otherwise} \end{cases}$$

$$\textit{min}_=(\tilde{b}_1, \tilde{b}_2, \Phi_=) = \begin{cases} \{((t_1, \prec_1), \varphi_1 \wedge \varphi_2 \wedge (t_1 = t_2))\} & \textbf{if } \Phi_= \wedge \prec_1 \leq \prec_2 \\ \{((t_2, \prec_2), \varphi_1 \wedge \varphi_2 \wedge (t_1 = t_2))\} & \textbf{if } \Phi_= \wedge \prec_2 < \prec_1 \\ \emptyset & \textbf{otherwise} \end{cases}$$

$$\textit{min}_>(\tilde{b}_1, \tilde{b}_2, \Phi_>) = \begin{cases} \{((t_2, \prec_2), \varphi_1 \wedge \varphi_2 \wedge (t_1 > t_2))\} & \textbf{if } \Phi_> \\ \emptyset & \textbf{otherwise} \end{cases}$$

The result of min operation — a set of one, two or three constrained parameterized bounds.

Symbolic Data Structures

- **Operator min $\tilde{\mathcal{PB}} \times \tilde{\mathcal{PB}} \rightarrow 2^{\tilde{\mathcal{PB}}}$**
- **Example**
  - **let let $\tilde{b}_1 = ((p+3, <), p > 8)$ and $\tilde{b}_2 = ((q, \leq), q < p+3)$.**

$$
\begin{aligned}
min(\tilde{b}_1, \tilde{b}_2) \quad &= \quad ((p+3, <), p > 8 \ \wedge \ q < p+3 \ \wedge \ p+3 < q) \\
&\cup \quad ((p+3, <), p > 8 \ \wedge \ q < p+3 \ \wedge \ p+3 = q) \\
&\cup \quad ((q, \leq), p > 8 \ \wedge \ q < p+3 \ \wedge \ p+3 > q)
\end{aligned}
$$

  - **$\Phi_<$ and $\Phi_=$ are trivially false**
  - $min(\tilde{b}_1, \tilde{b}_2) = ((q, \leq), p > 8 \ \wedge \ q < p+3)$**.**

Symbolic Data Structures

❖ Parametric DBM — operations

- **intersection** $\tilde{M}_1 \otimes \tilde{M}_2$**, where** $\tilde{M}_1 = (M_1, \Phi_1)$ **and** $\tilde{M}_2 = (M_2, \Phi_2)$**.**
- **inclusion:** $\forall x_i, x_j \in X \,.\, (M_1(i,j), \varphi_1) \subseteq^{\mathcal{PB}} (M_2(i,j), \varphi_2)$
- **adding constraint:** $\tilde{M}' = \tilde{M}_g \otimes \tilde{M}$
- **elapsing of time:** $\tilde{M}' = (M', \varphi) = \tilde{M} \Uparrow$**:**
  - $M'_{i,0} = (\infty, <)$ **if** $i \neq 0 \;\wedge\; x_i$ **is a clock.**
  - $M'_{i,j} = M_{i,j}$ **otherwise.**
- **linear assignment:** $\tilde{M}' = A * \tilde{M} * A^T + B$
- **a set of configurations:**

$$\boxed{post(q, \tilde{M}) = \{(q', \tilde{M}') \mid (A * (\tilde{M} \otimes \tilde{M}_g) * A^T + B) \Uparrow\}}$$

Symbolic Data Structures

❖ Polyhedra

- represents clocks and other variables for linear hybrid automata
- contraints: $\boxed{Ax \leq b}$
- dual representation

❖ Example:

Set of linear constraints

$$\begin{aligned} x + y &\geq 7 \\ y &\geq 2 \\ -x + y &\leq 1 \end{aligned}$$

Set of vertices and rays

$$v_1 = \begin{pmatrix} 5 \\ 2 \end{pmatrix} \qquad v_2 = \begin{pmatrix} 3 \\ 4 \end{pmatrix}$$

$$r_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad r_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Symbolic Data Structures

❖ Polyhedra

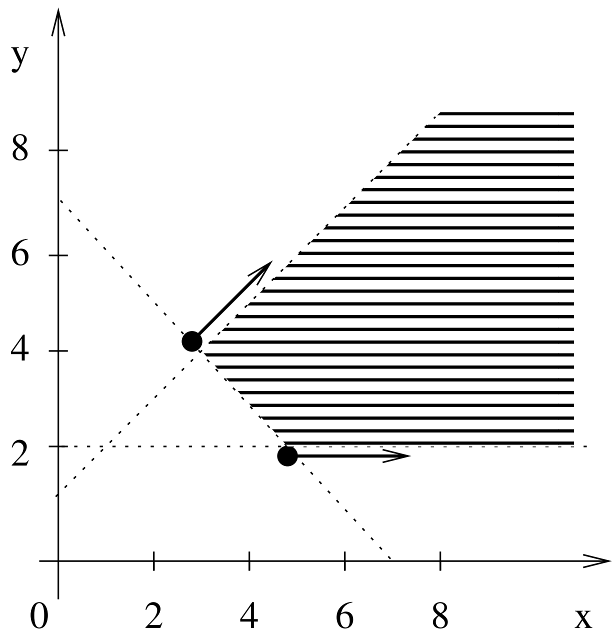
Set of linear constraints

$$\begin{aligned} x + y &\geq 7 \\ y &\geq 2 \\ -x + y &\leq 1 \end{aligned}$$

Set of vertices and rays

$$v_1 = \begin{pmatrix} 5 \\ 2 \end{pmatrix} \qquad v_2 = \begin{pmatrix} 3 \\ 4 \end{pmatrix}$$

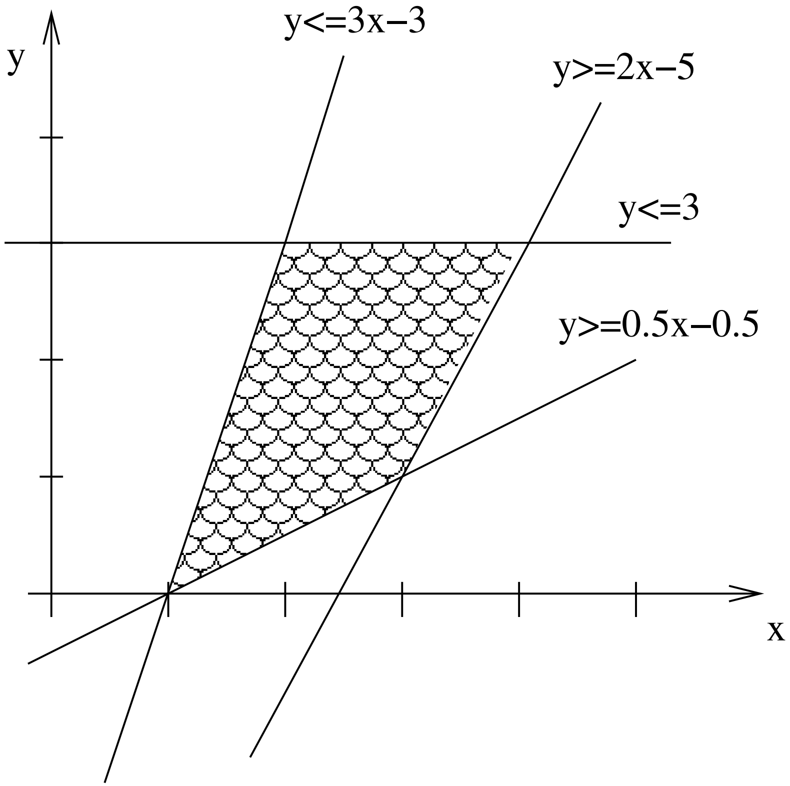
$$r_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad r_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

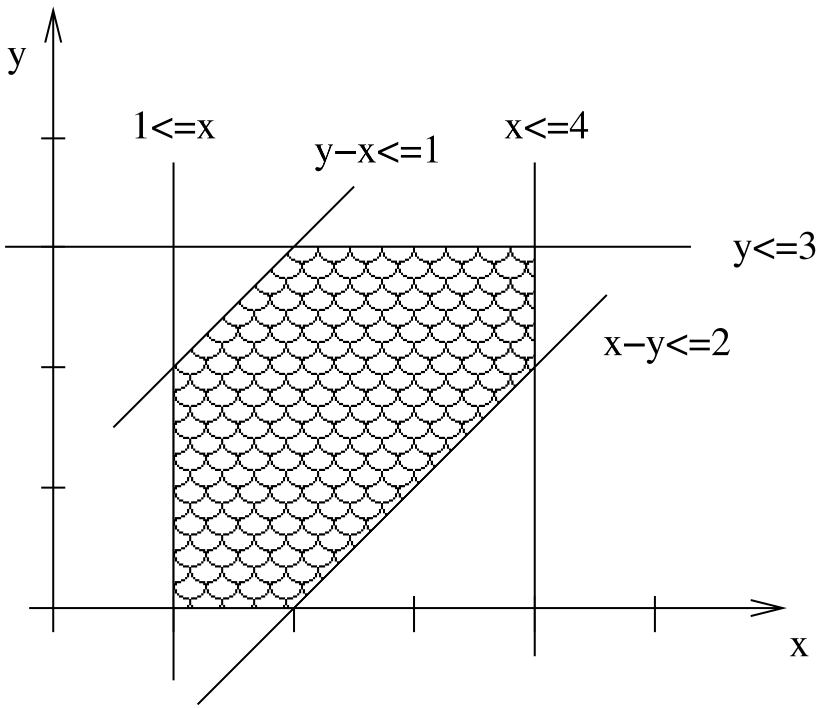


Symbolic Data Structures

❖ Polyhedra vs. PDBM

- **polyhedra** $y \geq 0.5x - 0.5, y \leq 3, y \geq 2x - 5, y \geq 3x - 3$
- **DBM** $0 < y \leq 3, 1 \leq x \leq 4, x - y \leq 2, y - x \leq 1$





Parametric Hypercubes

❖ Parametric Hypercubes (pHCubes)

- **represents counters for parameterized counter automata**
- **contraints:** $\boxed{x_1 + \ldots + x_n \prec t}$ **where** $x_i \in C, \prec \in \{<, \leq\}$
- t **is a term given by** $t ::= c \mid p \mid t - t \mid t + t \mid c * t$**,** $p \in \mathcal{P}$**,** $c \in \mathbb{Z}$
- **bounds on every variable in the form** $t_i^j \prec_i x_j \prec_s t_s^j$
- $ph(X) = (\vec{I}, \varphi) = (I_1, \ldots, I_n, \varphi) = (\langle a_1, b_1\rangle, \ldots, \langle a_n, b_n\rangle, \varphi)$**, where** $a_i = (t_i, \prec_i)$**, and** φ **is a constraint over parameters** $\mathcal{P}$
- **if no bound defined then** $t_i = (\infty, <)$

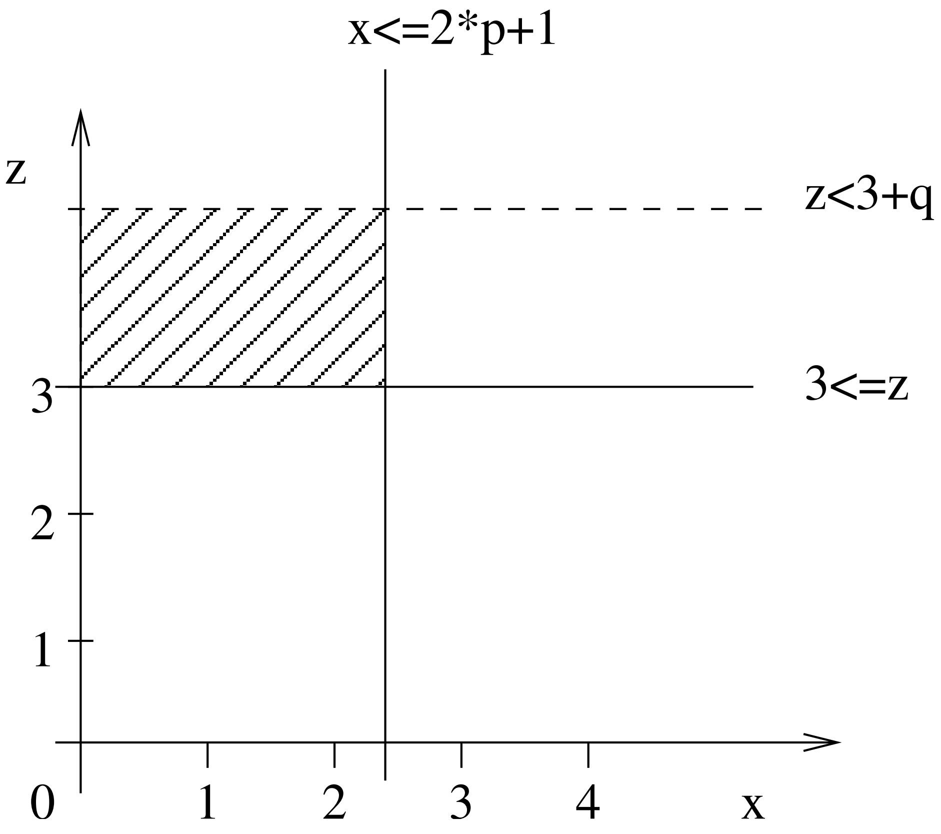
❖ Example

$$ph(X) = (\langle(<, 0), (\leq, 2 * p + 1)\rangle, \langle(<, \infty), (<, \infty)\rangle, \langle(\leq, -3), (<, 3 + q)\rangle, -p \leq 1 \;\wedge -q \leq -1)$$

Parametric Hypercubes

❖ **Example**

$ph(X) = (\langle(<, 0), (\leq, 2 * p + 1)\rangle, \langle(<, \infty), (<, \infty)\rangle, \langle(\leq, -3), (<, 3 + q)\rangle, -p \leq 1 \;\wedge -q \leq -1)$



$$\begin{aligned} 0 < \quad & x & \leq 2 * p + 1 \\ -\infty < \quad & y & < \infty \\ 3 \leq \quad & z & < 3 + q \\ \varphi \quad & = & p \geq -1 \;\wedge\; q \geq 1 \end{aligned}$$

Parametric Hypercubes

❖ Total order on parameterized bounds

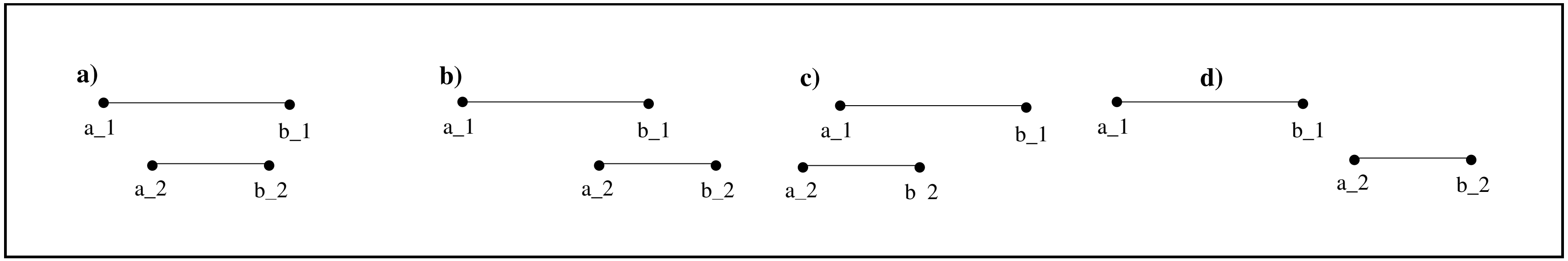
- **lower bounds:** $a_1 = (t_i^1, \prec_i^1)$, $a_2 = (t_i^2, \prec_i^2)$
- **upper bounds:** $b_1 = (t_s^1, \prec_s^1)$, $b_2 = (t_s^2, \prec_s^2)$

$$[\![a_1 \subseteq^{\mathcal{PB}} a_2]\!] = (t_i^1 < t_i^2) \vee ((t_t^1 = t_i^2) \wedge (\prec_i^1 \leq \prec_i^2)) \quad \textbf{(a)}$$

$$[\![b_1 \subseteq^{\mathcal{PB}} b_2]\!] = (t_s^1 < t_s^2) \vee ((t_s^1 = t_s^2) \wedge (\prec_s^1 \leq \prec_s^2)) \quad \textbf{(b)}$$

$$[\![a_1 \subseteq^{\mathcal{PB}} b_2]\!] = (-t_i^1 < t_s^2) \vee ((-t_i^1 = t_s^2) \wedge (\prec_i^1 \leq \prec_s^2)) \quad \textbf{(c)}$$

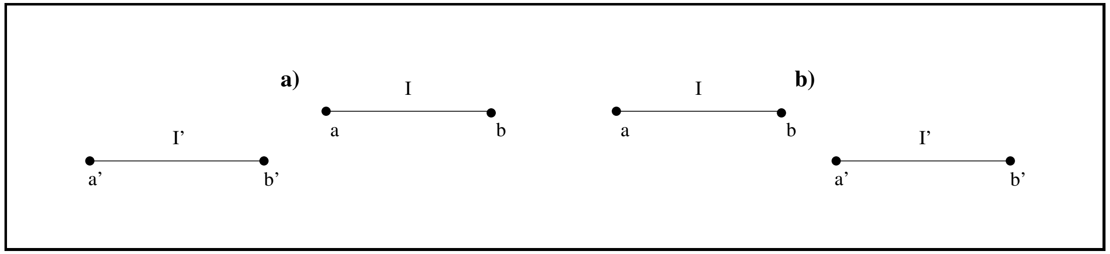
$$[\![b_1 \subseteq^{\mathcal{PB}} a_2]\!] = (t_s^1 < -t_i^2) \vee ((t_s^1 = -t_i^2) \wedge (\prec_i^1 \leq \prec_2^2)) \quad \textbf{(d)}$$



Parametric Hypercubes

❖ Operations

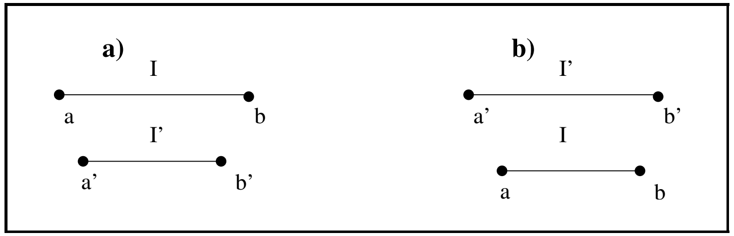
- **inclusion:**
  $$ph \subseteq ph' \iff \forall p \in \mathcal{P} \,.\, \varphi \Rightarrow (\varphi' \wedge \bigwedge_i (a_i \subseteq^{\mathcal{PB}} a'_i \wedge b_i \subseteq^{\mathcal{PB}} b'_i))$$
  For implementation:
  $$ph \not\subseteq ph' \iff \exists p \in \mathcal{P} \,.\, \neg(\varphi \Rightarrow \varphi') \vee (\varphi \wedge \bigwedge_i (\neg(a_i \subseteq^{\mathcal{PB}} a'_i \wedge b_i \subseteq^{\mathcal{PB}} b'_i))$$
- **testing emptiness:**
  ph **is not empty iff** $\exists p \in \mathcal{P} \,.\, \varphi \wedge \bigwedge_i (a_i \subseteq^{\mathcal{PB}} b_i)$
- **intersection:**
  1. **If** $I' < I$ **(a) or** $I < I'$ **(b), then** $I \cap I' = \emptyset, \psi = b' \subset^{\mathcal{PB}} a \vee b \subset^{\mathcal{PB}} a'$



Parametric Hypercubes

- **intersection**

  2. **If** $I' \subseteq I$ **(a) or** $I \subseteq I'$ **(b),** $\psi = a \subseteq^{\mathcal{PB}} a' \ \wedge \ b' \subseteq^{\mathcal{PB}} b$



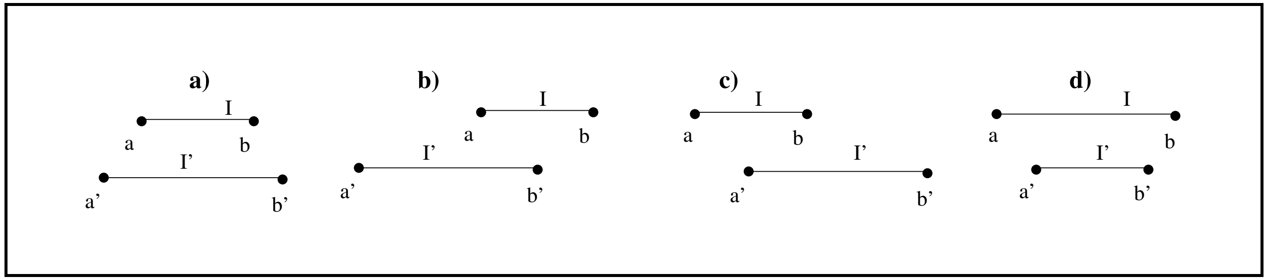
  3. **If** $I \cap I' \neq \emptyset$**, the following four cases are possible:**

     (a) $\psi = a' \subset^{\mathcal{PB}} a \ \wedge \ b \subset^{\mathcal{PB}} b'$**, i.e.,** $\psi = t_i < t'_i \ \wedge \ t_s < t'_s$, $I'' = \langle a, b \rangle$

     (b) $\psi = a' \subset^{\mathcal{PB}} a \ \wedge \ b' \subseteq^{\mathcal{PB}} b$**, i.e.,** $\psi = t_i < t'_i \ \wedge \ t'_s \leq t_s$, $I'' = \langle a, b' \rangle$

     (c) $\psi = a \subseteq^{\mathcal{PB}} a' \ \wedge \ b \subset^{\mathcal{PB}} b'$**, i.e.,** $\psi = t'_i \leq t_i \ \wedge \ t_s < t'_s$, $I'' = \langle a', b \rangle$

     (d) $\psi = a \subseteq^{\mathcal{PB}} a' \ \wedge \ b' \subseteq^{\mathcal{PB}} b$**, i.e.,** $\psi = t'_i \leq t_i \ \wedge \ t'_s \leq t_s$, $I'' = \langle a', b' \rangle$



Parametric Hypercubes

1. **Adding simple constraint** $x \prec_g t_g$
   - **interval** $I(x) = (\langle a, b \rangle, \varphi)$
   - **guard** $I_g(x) = (\langle a_g, b_g \rangle, \varphi_g)$**, i.e.** $x \prec_{gs} t_{gs} \ \wedge \ -x \prec_{gi} t_{gi}$

$$\boxed{I'(x) = min_{\mathcal{I}}(I(x), I_g(x))}$$

- $\boldsymbol{min}(a, a_g) = \boldsymbol{min}_<(a, a_g, \Phi_<) \ \cup \ \boldsymbol{min}_=(a, a_g, \Phi_=) \ \cup \ \boldsymbol{min}_>(a, a_g, \Phi_>)$
- $\boldsymbol{min}(b, b_g) = \boldsymbol{min}_<(b, b_g, \Phi_<) \ \cup \ \boldsymbol{min}_=(b, b_g, \Phi_=) \ \cup \ \boldsymbol{min}_>(b, b_g, \Phi_>)$

Parametric Hypercubes

2. **Adding general constraint** $x_0 t_0 + \ldots + x_n t_n \prec t_{n+1}$
   - **not exact, only approximate value of bounds of a guard**
   - **computation for the upper bound:**

$$\begin{aligned} x_0 t_0 + \ldots + x_n t_n &\prec_{gs} t_{n+1} \\ x_0 &\prec_{gs} \frac{t_{n+1}}{t_0} - \frac{t_1}{t_0} x_1 - \ldots - \frac{t_n}{t_0} x_n \end{aligned}$$

   - **substitution of** x_i **by their upper bounds**

$$x_0 \prec_{gs} \underbrace{\frac{t_{n+1}}{t_0} - \frac{t_1}{t_0} t_{1s} - \ldots - \frac{t_n}{t_0} t_{ns}}_{t_{gs}}$$

$$\boxed{I'(x) = min_{\mathcal{I}}(I(x), I_g(x))}$$

Parametric Hypercubes

❖ Linear assignment

1. **simple linear assignment $x := t'$:**

$$\boxed{I'(x) = I(x)|_{x:=t'} = (\langle(-t', \leq), (t', \leq)\rangle, \varphi)}$$

2. **general linear assignment $x := x_0 t_0 + \ldots + x_n t_n + t_{n+1}$**
   - **lower bound:** $t'_i = \tilde{t_0} + \ldots + \tilde{t_n} + t_{n+1}$
     $\tilde{t_i} = t_{il}$ **if** $t_i \geq 0, \tilde{t_i} = t_{iu}$ **if** $t_i < 0$
   - **upper bound:** $t'_s = \tilde{t_0} + \ldots + \tilde{t_n} + t_{n+1}$

$$\boxed{I'(x) = I(x)|_{x:=\vec{t}} = (\langle(t'_i, \leq), (t'_s, \leq)\rangle, \varphi)}$$

❖ Example:

- **suppose intervals** $1 < x < 3$**,** $2 < y < 5$ **and** $0 < z < 10$
- **general linear assignment** $x := 2 - y + z$
- **a new interval of** x **is** $-3 < x < 10$

Parametric Hypercubes

❖ Operation $post()$

$$\boxed{post(q, ph(X)) = \{(q', ph'(X)) \mid (ph(X) \cap ph_g(X))|_{X:=T(X)}\}}$$

- $ph_g(X)$ **is a valuation of guards**
- $T(X)$ **is a set of linear assignments**

Conclusion

❖ Contribution of Parameterized Hypecubes

- **linear constraints over transitions of the form** $x_0t_0 + \ldots + x_nt_n \prec t_{n+1}$
- **general actions (assignment) over transitions of the form** $x := x_0t_0 + \ldots + x_nt_n + t_{n+1}$
- **space reduction in comparison with PDBMs**
- **normal form is not needed for intervals manipulation**