

Zone-Based Reachability Analysis

Lecture #19 of Advanced Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

E-mail: katoen@cs.rwth-aachen.de

January 25, 2011

TCTL model checking

- Model checking timed automata against TCTL is **decidable**
 - example TCTL-formula: $\forall \diamond^{\leq 10} \text{goal}$
- Key ingredient for decidability: finite quotient wrt. a bisimulation
 - bisimulation = equivalence on clock valuations
 - equivalence classes are called *regions*
- Region automaton is highly impractical for tool implementation
 - the number of regions lies in $\Theta(|C|! \cdot \prod_{x \in C} c_x)$
- In practice, coarser abstractions than regions are used
 - this lecture considers time-bounded reachability using **zones**

Reachability analysis

- **Forward** analysis:
 - starting from some initial configuration
 - determine configurations that are reachable within 1, 2, 3, . . . steps
 - until either the **goal** configuration is reached, or the computation **terminates**
- **Backward** analysis:
 - starting from the goal configuration
 - determine configurations that can reach the goal within 1, 2, 3, . . . steps
 - until either the **initial** configuration is reached, or the computation **terminates**

how can these approaches be realized for timed automata?

Symbolic reachability analysis

- Use a **symbolic** representation of timed automata configurations
 - needed as there are infinitely many configurations
 - example: state regions $\langle \ell, [\eta] \rangle$
- For set z of clock valuations and edge $e = \ell \xleftarrow{g:\alpha, D} \ell'$ let:

$$Post_e(z) = \{ \eta' \in \mathbb{R}_{\geq 0}^n \mid \exists \eta \in z, d \in \mathbb{R}_{\geq 0}. \eta + d \models g \wedge \eta' = \text{reset } D \text{ in } (\eta + d) \}$$

$$Pre_e(z) = \{ \eta \in \mathbb{R}_{\geq 0}^n \mid \exists \eta' \in z, d \in \mathbb{R}_{\geq 0}. \eta + d \models g \wedge \eta' = \text{reset } D \text{ in } (\eta + d) \}$$
- Intuition:
 - $\eta' \in Post_e(z)$ if for some $\eta \in z$ and delay d , $(\ell, \eta) \xrightarrow{d} \dots \xrightarrow{e} (\ell', \eta')$
 - $\eta \in Pre_e(z)$ if for some $\eta' \in z$ and delay d , $(\ell, \eta) \xrightarrow{d} \dots \xrightarrow{e} (\ell', \eta')$

Zones

- Clock constraints are *conjunctions* of constraints of the form:
 - $x \prec c$ and $x - y \prec c$ for $\prec \in \{ <, \leq, =, \geq, > \}$, and $c \in \mathbb{Z}$
- A *zone* is a set of clock valuations satisfying a clock constraint
 - a clock zone for g is the set of clock valuations satisfying g
- Clock zone of g : $\llbracket g \rrbracket = \{ \eta \in \text{Eval}(C) \mid \eta \models g \}$
- The *state zone* of $s = \langle \ell, \eta \rangle$ is $\langle \ell, z \rangle$ with $\eta \in z$
- For *zone* z and edge e , $\text{Post}_e(z)$ and $\text{Pre}_e(z)$ are *zones*

state zones will be used as symbolic representations for configurations

Example zones

on the black board

zones are convex polyhedra

Operations on zones

- Future of z :
 - $\vec{z} = \{ \eta + d \mid \eta \in z \wedge d \in \mathbb{R}_{\geq 0} \}$
- Past of z :
 - $\overleftarrow{z} = \{ \eta - d \mid \eta \in z \wedge d \in \mathbb{R}_{\geq 0} \}$
- Intersection of two zones:
 - $z \cap z' = \{ \eta \mid \eta \in z \wedge \eta \in z' \}$
- Clock reset in a zone:
 - reset D in $z = \{ \text{reset } D \text{ in } \eta \mid \eta \in z \}$
- Inverse clock reset of a zone:
 - $\text{reset}^{-1} D \text{ in } z = \{ \eta \mid \text{reset } D \text{ in } \eta \in z \}$

Operations on zones: examples

on the black board

zones are closed under all aforementioned operations

Symbolic successors and predecessors

Recall that for edge $e = \ell \xleftarrow{g:\alpha, D} \ell'$ we have:

$$Post_e(z) = \{ \eta' \in \mathbb{R}_{\geq 0}^n \mid \exists \eta \in z, d \in \mathbb{R}_{\geq 0}. \eta + d \models g \wedge \eta' = \text{reset } D \text{ in } (\eta + d) \}$$

$$Pre_e(z) = \{ \eta \in \mathbb{R}_{\geq 0}^n \mid \exists \eta' \in z, d \in \mathbb{R}_{\geq 0}. \eta + d \models g \wedge \eta' = \text{reset } D \text{ in } (\eta + d) \}$$

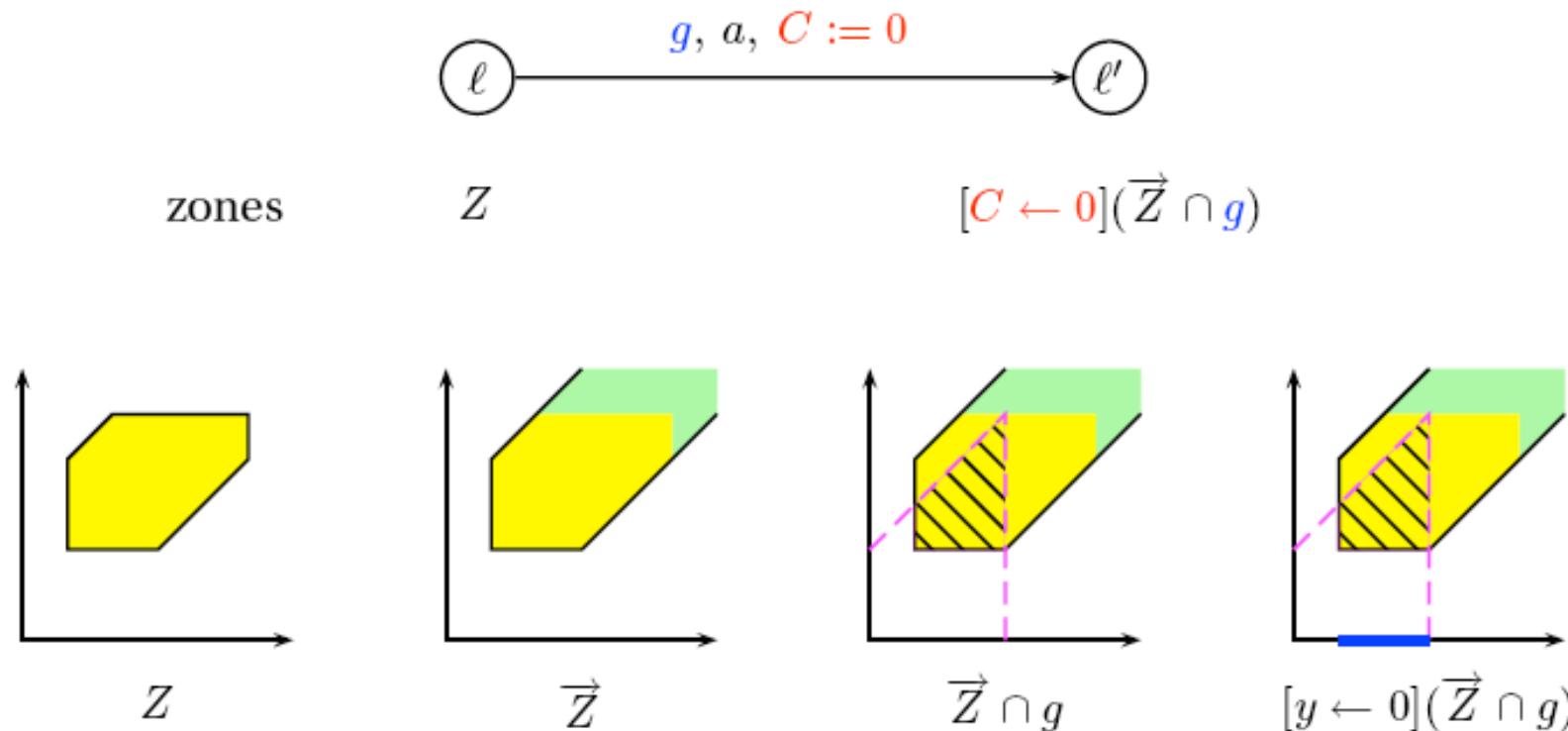
This can also be expressed symbolically using operations on zones:

$$Post_e(z) = \text{reset } D \text{ in } (\vec{z} \cap \llbracket g \rrbracket)$$

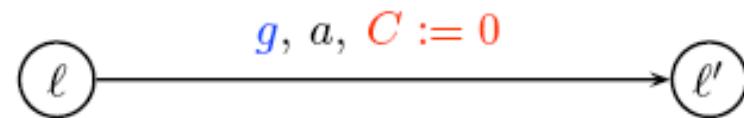
and

$$Pre_e(z) = \overleftarrow{\text{reset}^{-1} D \text{ in } (z \cap \llbracket D = 0 \rrbracket)} \cap \llbracket g \rrbracket$$

Zone successor: example

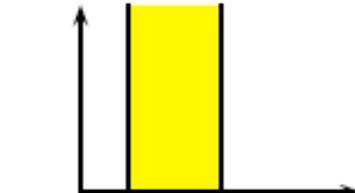
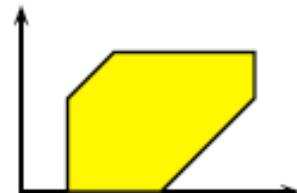


Zone predecessor: example



$$\overleftarrow{[C \leftarrow 0]^{-1}(Z \cap (C = 0)) \cap g}$$

Z



$$\overleftarrow{[C \leftarrow 0]^{-1}(Z \cap (C = 0)) \cap g}$$

Backward symbolic transition system (1)

Backward symbolic transition system of TA with $|C| = n$ is inductively defined by:

$$\frac{e = \ell \xleftarrow{g:\alpha, D} \ell' \quad z = \text{Pre}_e(z')}{(\ell', z') \Leftarrow (\ell, z)}$$

Iterative backward reachability analysis computation schemata:

$$\begin{aligned} T_0 &= \{ (\ell, \mathbb{R}_{\geq 0}^n) \mid \ell \text{ is a goal location} \} \\ T_1 &= T_0 \cup \{ (\ell, z) \mid \exists (\ell', z') \in T_0 \text{ such that } (\ell', z') \Leftarrow (\ell, z) \} \\ &\dots &\dots \\ T_{k+1} &= T_k \cup \{ (\ell, z) \mid \exists (\ell', z') \in T_k \text{ such that } (\ell', z') \Leftarrow (\ell, z) \} \\ &\dots &\dots \end{aligned}$$

until either the computation stabilizes or reaches an initial configuration (ℓ_0, z_0)

Backward symbolic transition system (2)

Backward symbolic transition system of TA is inductively defined by:

$$\frac{e = \ell \xleftarrow{g:\alpha, D} \ell' \quad z = \text{Pre}_e(z')}{(\ell', z') \Leftarrow (\ell, z)}$$

Iterative backward reachability analysis computation schemata:

$$T_0 = \{ (\ell, \mathbb{R}_{\geq 0}^n) \mid \ell \text{ is a goal location} \}$$

$$T_1 = T_0 \cup \{ (\ell, z) \mid \exists (\ell', z') \in T_0. (\ell', z') \Leftarrow (\ell, z) \text{ and } \ell' = \ell \text{ implies } z \not\subseteq z' \}$$

... ...

$$T_{k+1} = T_k \cup \{ (\ell, z) \mid \exists (\ell', z') \in T_k. (\ell', z') \Leftarrow (\ell, z) \text{ and } \ell' = \ell \text{ implies } z \not\subseteq z' \}$$

... ...

until either the computation stabilizes or reaches an initial configuration (ℓ_0, z_0)

Termination and correctness [\[Henzinger et al., 1994\]](#)

The backward computation terminates and is correct wrt. reachability properties

Because of the bisimulation property, it holds:

Every set of valuations which is computed along the backward computation is a finite union of regions

Forward reachability analysis (1)

Forward symbolic transition system of TA is inductively defined by:

$$\frac{e = \ell \xleftarrow{g:\alpha, D} \ell' \quad z' = \text{Post}_e(z)}{(\ell, z) \Rightarrow (\ell', z')}$$

Iterative forward reachability analysis computation schemata:

$$\begin{aligned} T_0 &= \{ (\ell_0, z_0) \mid \forall x \in C. z_0(x) = 0 \} \\ T_1 &= T_0 \cup \{ (\ell', z') \mid \exists (\ell, z) \in T_0 \text{ such that } (\ell, z) \Rightarrow (\ell', z') \} \\ &\dots &\dots \\ T_{k+1} &= T_k \cup \{ (\ell', z') \mid \exists (\ell, z) \in T_k \text{ such that } (\ell, z) \Rightarrow (\ell', z') \} \\ &\dots &\dots \end{aligned}$$

until either the computation stabilizes or reaches a symbolic state containing a goal configuration

Forward reachability analysis (2)

Forward symbolic transition system of TA is inductively defined by:

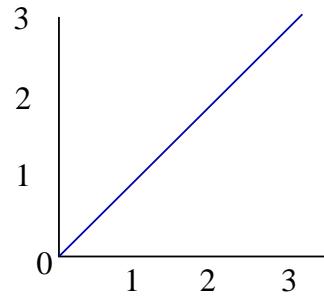
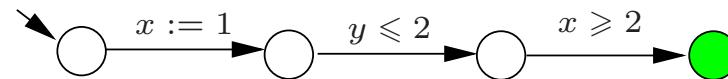
$$\frac{e = \ell \xleftarrow{g:\alpha, D} \ell' \quad z' = \text{Post}_e(z)}{(\ell, z) \Rightarrow (\ell', z')}$$

Iterative forward reachability analysis computation schemata:

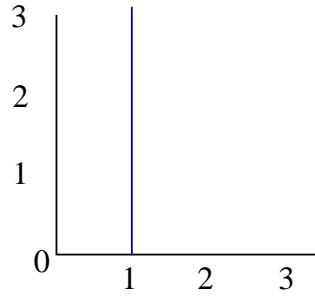
$$\begin{aligned} T_0 &= \{ (\ell_0, z_0) \mid \forall x \in C. z_0(x) = 0 \} \\ T_1 &= T_0 \cup \{ (\ell', z') \mid \exists (\ell, z) \in T_0. (\ell, z) \Rightarrow (\ell', z') \text{ and } \ell = \ell' \text{ implies } z \not\subseteq z' \} \\ \dots &\quad \dots \\ T_{k+1} &= T_k \cup \{ (\ell', z') \mid \exists (\ell, z) \in T_k. (\ell, z) \Rightarrow (\ell', z') \text{ and } \ell = \ell' \text{ implies } z \not\subseteq z' \} \\ \dots &\quad \dots \end{aligned}$$

until either the computation stabilizes or reaches a symbolic state containing a goal configuration

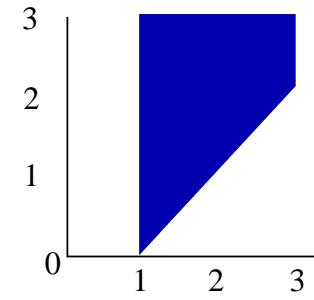
Forward reachability analysis: intuition



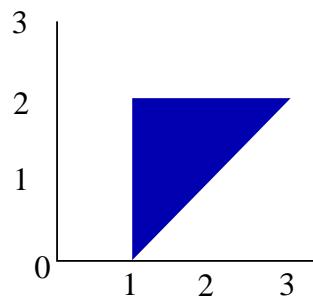
leaving initial



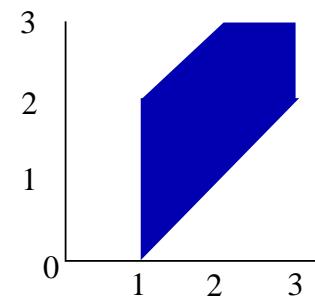
entering first



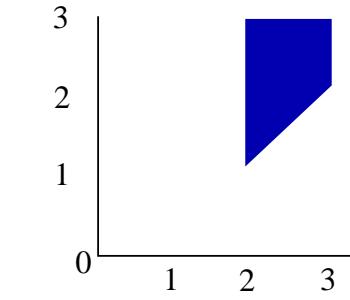
leaving first



entering second



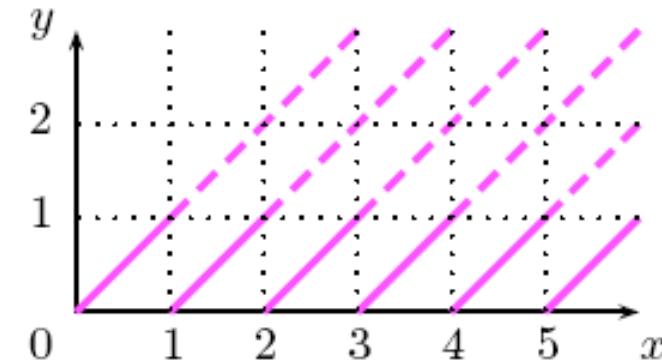
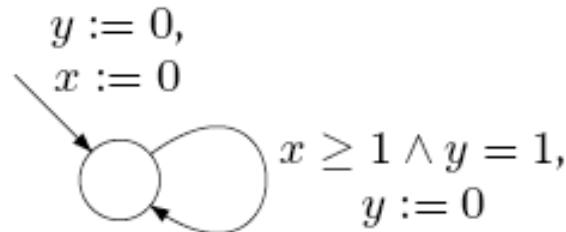
leaving second



entering third

Possible non-termination

The forward analysis is correct but may **not** terminate:



→ an infinite number of steps...

Solution: abstract forward reachability

Let γ associate sets of valuations to sets of valuations

Abstract forward symbolic transition system of TA is defined by:

$$\frac{(\ell, z) \Rightarrow (\ell', z') \quad z = \gamma(z)}{(\ell, z) \Rightarrow_{\gamma} (\ell', \gamma(z'))}$$

Iterative forward reachability analysis computation schemata:

$$T_0 = \{ (\ell_0, \gamma(z_0)) \mid \forall x \in C. z_0(x) = 0 \}$$

$$T_1 = T_0 \cup \{ (\ell', z') \mid \exists (\ell, z) \in T_0 \text{ such that } (\ell, z) \Rightarrow_{\gamma} (\ell', z') \}$$

$$\dots \quad \dots$$

$$T_{k+1} = T_k \cup \{ (\ell', z') \mid \exists (\ell, z) \in T_k \text{ such that } (\ell, z) \Rightarrow_{\gamma} (\ell', z') \}$$

$$\dots \quad \dots$$

with inclusion check and termination criteria as before

Soundness and correctness

- **Soundness:**

$$\underbrace{\langle \ell_0, \gamma(z_0) \rangle \Rightarrow^*_{\gamma} \langle \ell, z \rangle}_{\text{abstract symbolic reachability}} \quad \text{implies} \quad \exists \underbrace{\langle \ell_0, \eta_0 \rangle \rightarrow^* \langle \ell, \eta \rangle}_{\text{reachability in } TS(TA)} \text{ with } \eta \in z$$

- **Completeness:**

$$\underbrace{\langle \ell_0, \eta_0 \rangle \rightarrow^* \langle \ell, \eta \rangle}_{\text{reachability in } TS(TA)} \quad \text{implies} \quad \exists \underbrace{\langle \ell_0, \gamma(\{ \eta_0 \}) \rangle \Rightarrow^*_{\gamma} \langle \ell, z \rangle}_{\text{abstract symbolic reachability}} \text{ for some } z \text{ with } \eta \in z$$

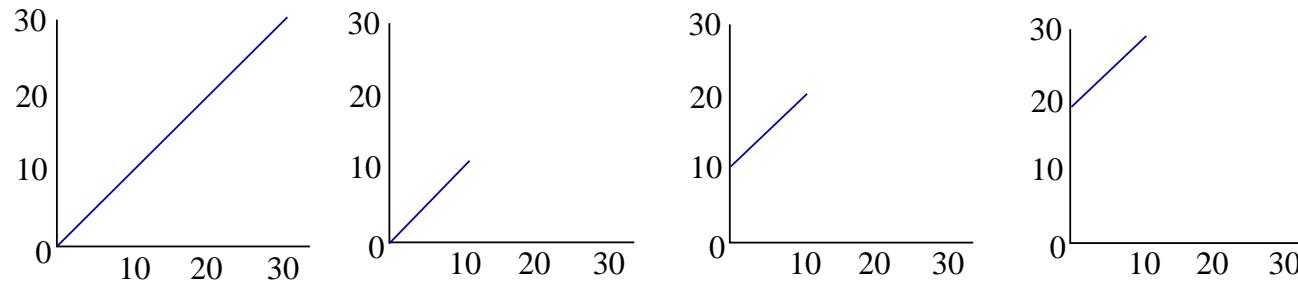
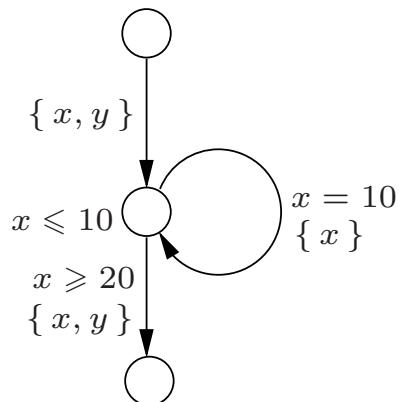
for any choice of γ , soundness and completeness are desirable

Criteria on the abstraction operator

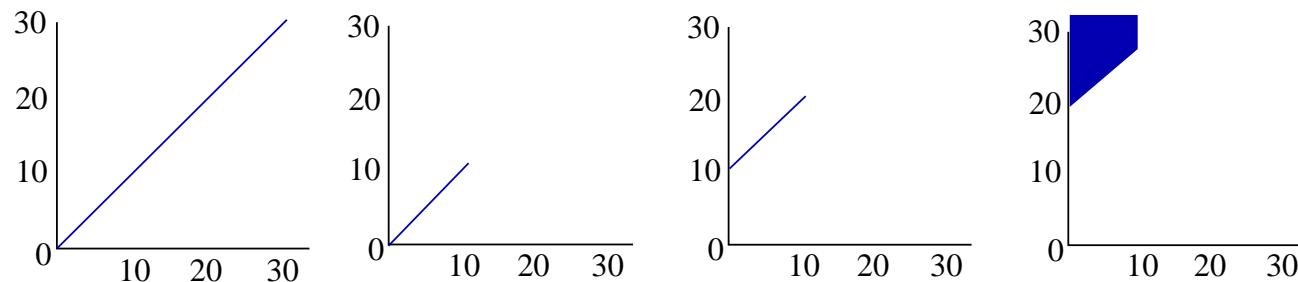
- **Finiteness:** $\{ \gamma(z) \mid \gamma \text{ defined on } z \}$ is finite
- **Correctness:** γ is sound wrt. reachability
- **Completeness:** γ is complete wrt. reachability
- **Effectiveness:** γ is defined on zones, and $\gamma(z)$ is a zone

Normalization: intuition

symbolic semantics has infinitely many zones:



normalization yields a finite zone graph:

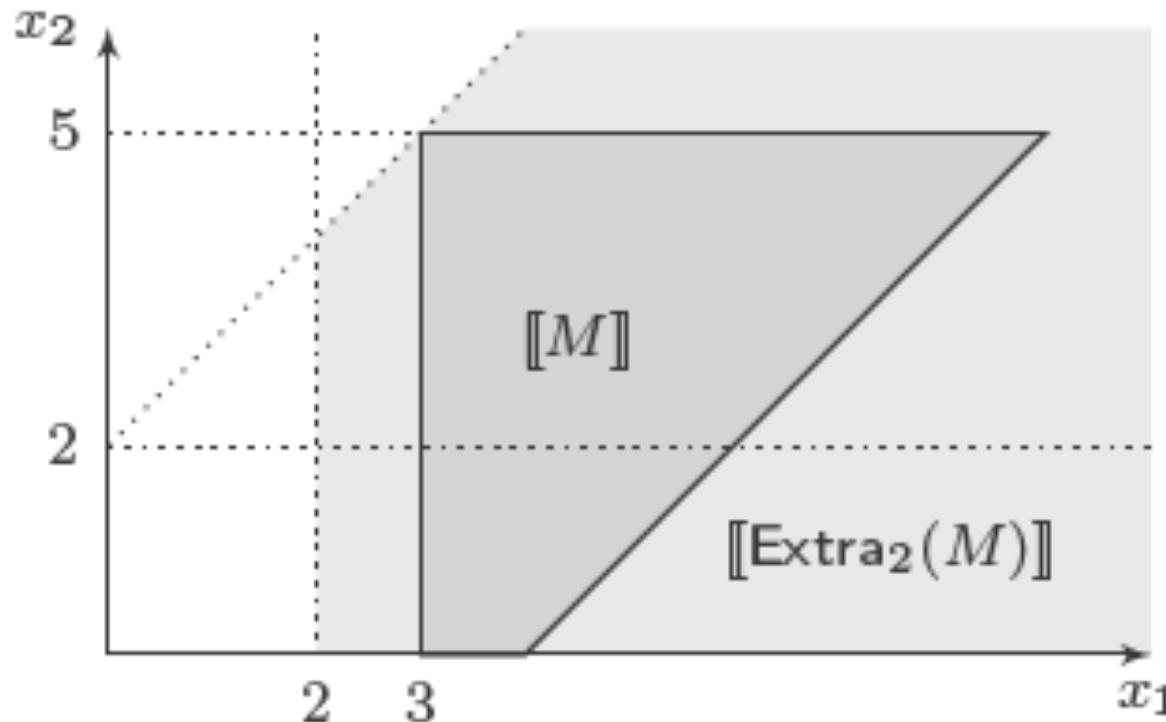


k -Normalization [Daws & Yovine, 1998]

Let $k \in \mathbb{N}$.

- A k -bounded zone is described by a k -bounded clock constraint
 - e.g., zone $z = (x \geq 3) \wedge (y \leq 5) \wedge (x - y \leq 4)$ is not 2-bounded
 - but zone $z' = (x \geq 2) \wedge (y - x \leq 2)$ is 2-bounded
 - note that: $z \subseteq z'$
- Let $norm_k(z)$ be the smallest k -bounded zone containing zone z

Example of k -normalization



Facts about k -normalization [Bouyer, 2003]

- **Finiteness:** $norm_k(\cdot)$ is a finite abstraction operator
- **Correctness:** $norm_k(\cdot)$ is sound wrt. reachability
 - provided k is the maximal constant appearing in the constraints of TA
- **Completeness:** $norm_k(\cdot)$ is complete wrt. reachability
 - since $z \subseteq norm_k(z)$, so $norm_k(\cdot)$ is an over-approximation
- **Effectiveness:** $norm_k(z)$ is a zone
 - this will be made clear in the sequel when considering zone representations