A **Forward Construction** of **Inductive Invariants** for **Vector Addition Systems**

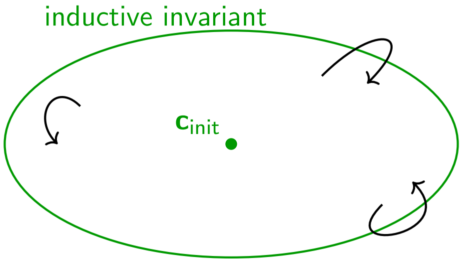
Clotilde Bizière Jérôme Leroux Grégoire Sutre

LaBRI, Université de Bordeaux (France)

SAMSA Workshop, Warsaw, 04/06/2025

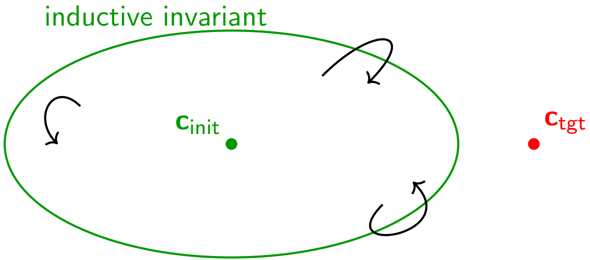
Introduction: inductive invariants

Introduction: inductive invariants



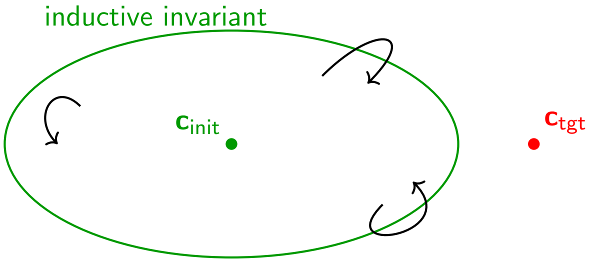
- An inductive invariant is a set which
  - contains the initial configuration
  - is stable under transitions

Introduction: inductive invariants



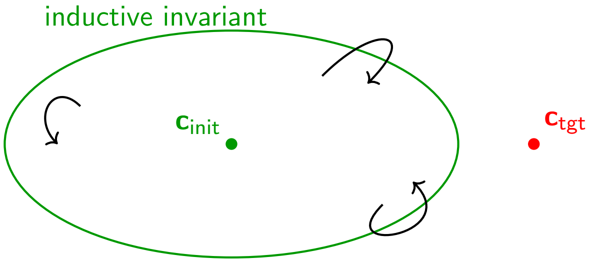
- An inductive invariant is a set which
  - contains the initial configuration
  - is stable under transitions
- If you find an inductive invariant which doesn't contain $\mathbf{c}_{\text{tgt}}$, then $\mathbf{c}_{\text{tgt}}$ is not reachable.

Introduction: inductive invariants



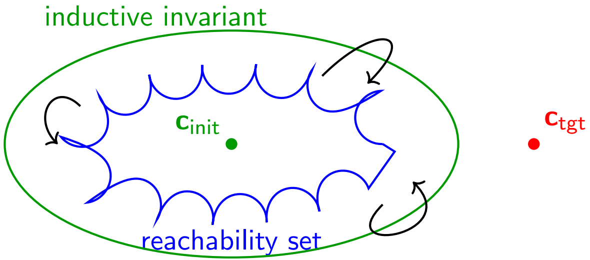
- An inductive invariant is a set which
  - contains the initial configuration
  - is stable under transitions
- If you find an inductive invariant which doesn't contain $\mathbf{c}_{\text{tgt}}$, then $\mathbf{c}_{\text{tgt}}$ is not reachable.
- Conversely, if $\mathbf{c}_{\text{tgt}}$ is not reachable, then there are inductive invariants which don't contain $\mathbf{c}_{\text{tgt}}$

Introduction: inductive invariants



- An inductive invariant is a set which
  - contains the initial configuration
  - is stable under transitions
- If you find an inductive invariant which doesn't contain $\mathbf{c}_{\mathrm{tgt}}$, then $\mathbf{c}_{\mathrm{tgt}}$ is not reachable.
- Conversely, if $\mathbf{c}_{\mathrm{tgt}}$ is not reachable, then there are inductive invariants which don't contain $\mathbf{c}_{\mathrm{tgt}}$... take the reachability set itself !

Introduction: inductive invariants



- An inductive invariant is a set which
  - contains the initial configuration
  - is stable under transitions
- If you find an inductive invariant which doesn't contain c_{tgt}, then c_{tgt} is not reachable.
- Conversely, if c_{tgt} is not reachable, then there are inductive invariants which don't contain c_{tgt}... take the reachability set itself !

Introduction: inductive invariants

Introduction: inductive invariants

- Typical situation:

Introduction: inductive invariants

- Typical situation:
  - The configurations of the system are easily enumerable

Introduction: inductive invariants

- Typical situation:
  - The configurations of the system are easily enumerable $\Rightarrow$ simple semi-algorithm for reachability

Introduction: inductive invariants

- Typical situation:
  - The configurations of the system are easily enumerable $\Rightarrow$ simple semi-algorithm for reachability
  - But the potential inductive invariants are uncountable.

Introduction: inductive invariants

- Typical situation:
  - The configurations of the system are easily enumerable $\Rightarrow$ simple semi-algorithm for reachability
  - But the potential inductive invariants are uncountable.
- If we can find a family $\mathcal{F}$ of sets such that

Introduction: inductive invariants

- Typical situation:
  - The configurations of the system are easily enumerable $\Rightarrow$ simple semi-algorithm for reachability
  - But the potential inductive invariants are uncountable.
- If we can find a family $\mathcal{F}$ of sets such that
  - $\mathcal{F}$ is recursively enumerable

Introduction: inductive invariants

- Typical situation:
  - The configurations of the system are easily enumerable $\Rightarrow$ simple semi-algorithm for reachability
  - But the potential inductive invariants are uncountable.
- If we can find a family $\mathcal{F}$ of sets such that
  - $\mathcal{F}$ is recursively enumerable
  - non-reachability is certified by inductive invariants in $\mathcal{F}$

Introduction: inductive invariants

- Typical situation:
  - The configurations of the system are easily enumerable $\Rightarrow$ simple semi-algorithm for reachability
  - But the potential inductive invariants are uncountable.
- If we can find a family $\mathcal{F}$ of sets such that
  - $\mathcal{F}$ is recursively enumerable
  - non-reachability is certified by inductive invariants in $\mathcal{F}$
  - (it is decidable whether a set in $\mathcal{F}$ is an inductive invariant)

Introduction: inductive invariants

- Typical situation:
  - The configurations of the system are easily enumerable $\Rightarrow$ simple semi-algorithm for reachability
  - But the potential inductive invariants are uncountable.
- If we can find a family $\mathcal{F}$ of sets such that
  - $\mathcal{F}$ is recursively enumerable
  - non-reachability is certified by inductive invariants in $\mathcal{F}$
  - (it is decidable whether a set in $\mathcal{F}$ is an inductive invariant)

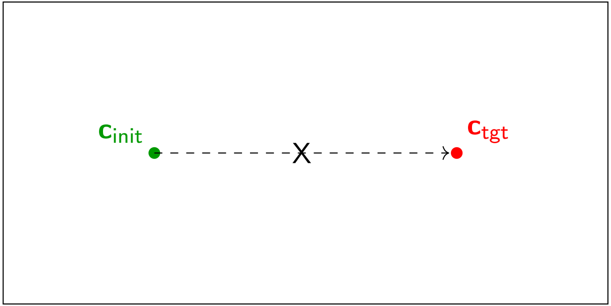
  then there is an semi-algorithm for non-reachability.

Introduction: inductive invariants

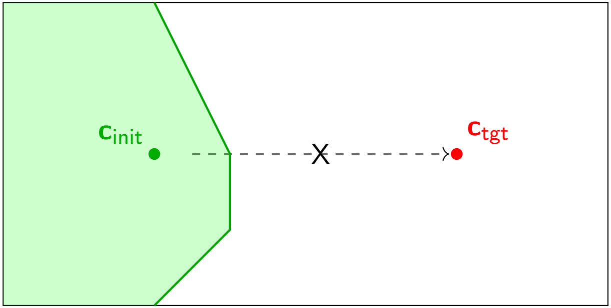
- Typical situation:
  - The configurations of the system are easily enumerable $\Rightarrow$ simple semi-algorithm for reachability
  - But the potential inductive invariants are uncountable.
- If we can find a family $\mathcal{F}$ of sets such that
  - $\mathcal{F}$ is recursively enumerable
  - non-reachability is certified by inductive invariants in $\mathcal{F}$
  - (it is decidable whether a set in $\mathcal{F}$ is an inductive invariant)

  then there is an semi-algorithm for non-reachability.
- In 2011, Leroux proved that VAS non-reachability is certified by semilinear sets.

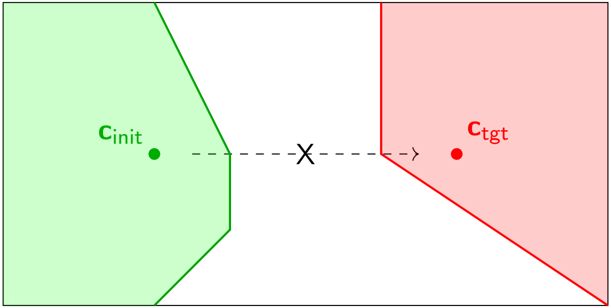
Back-and-Forth Construction of Inductive Invariants



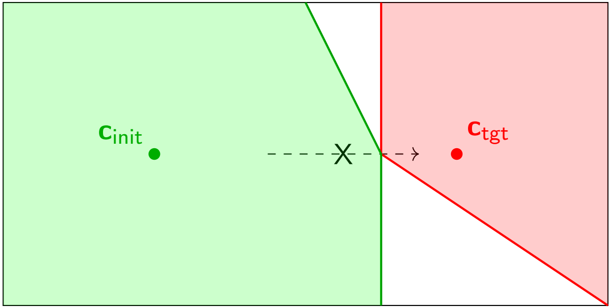
Back-and-Forth Construction of Inductive Invariants



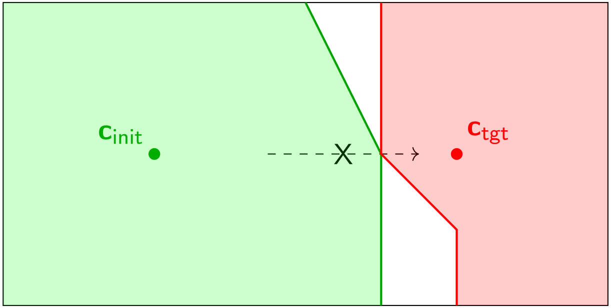
Back-and-Forth Construction of Inductive Invariants



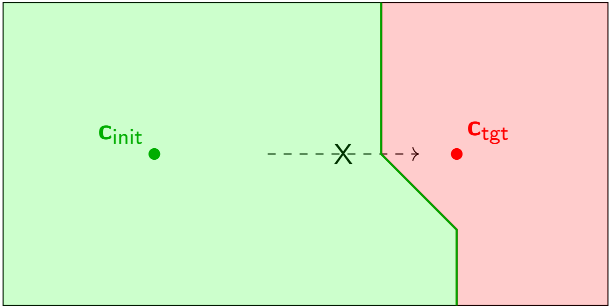
Back-and-Forth Construction of Inductive Invariants



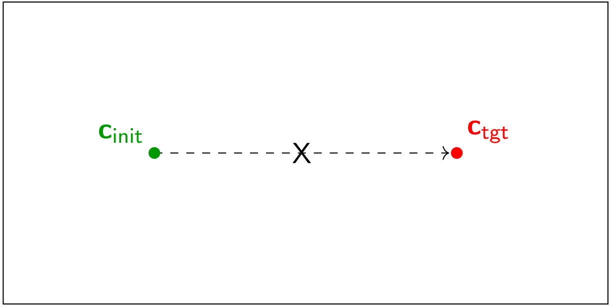
Back-and-Forth Construction of Inductive Invariants



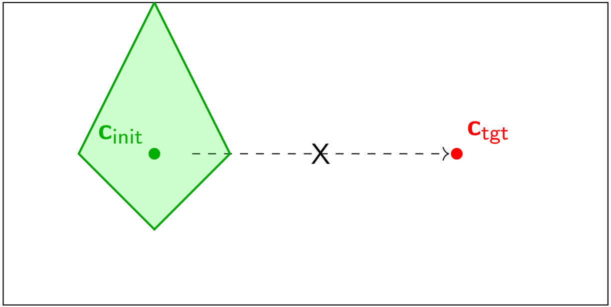
Back-and-Forth Construction of Inductive Invariants



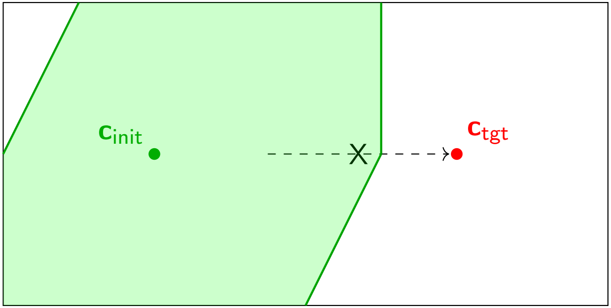
Forward Construction



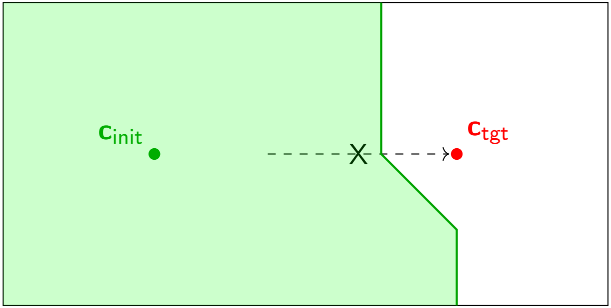
Forward Construction



Forward Construction



Forward Construction



Definitions (VAS + semilinear sets)

- A Vector Addition System (VAS) is a pair $(\mathbf{c}_{\text{init}}, \mathbf{A})$ where $\mathbf{c}_{\text{init}} \in \mathbb{N}^d$ and $\mathbf{A} \subseteq \mathbb{Z}^d$ is finite.

Definitions (VAS + semilinear sets)

- A Vector Addition System (VAS) is a pair $(\mathbf{c}_{\text{init}}, \mathbf{A})$ where $\mathbf{c}_{\text{init}} \in \mathbb{N}^d$ and $\mathbf{A} \subseteq \mathbb{Z}^d$ is finite.
- It generates a transition system whose configurations are vectors in $\mathbb{N}^d$ and whose transitions are of the form $\mathbf{x} \to \mathbf{x} + \mathbf{a}$ for $\mathbf{a} \in \mathbf{A}$.

Definitions (VAS + semilinear sets)

- A Vector Addition System (VAS) is a pair $(\mathbf{c}_{\text{init}}, \mathbf{A})$ where $\mathbf{c}_{\text{init}} \in \mathbb{N}^d$ and $\mathbf{A} \subseteq \mathbb{Z}^d$ is finite.
- It generates a transition system whose configurations are vectors in $\mathbb{N}^d$ and whose transitions are of the form $\mathbf{x} \to \mathbf{x} + \mathbf{a}$ for $\mathbf{a} \in \mathbf{A}$.
- A semilinear set is a finite union of sets of the form

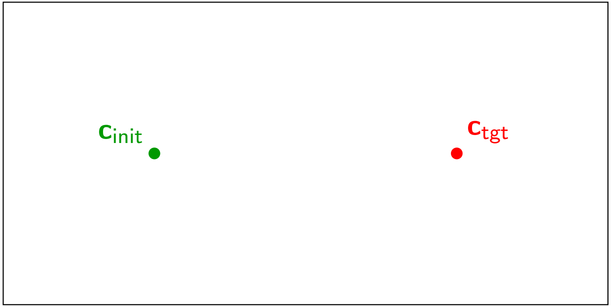
$$\mathbf{b} + \mathbf{P}^* \quad \text{(called linear sets)}$$

for some $\mathbf{b} \in \mathbb{N}^d$ (the basis) and finite $\mathbf{P} \subseteq \mathbb{N}^d$ (the periods), where

$$\mathbf{P}^* := \{\mathbf{p_1} + \ldots + \mathbf{p_n} \mid n \in \mathbb{N}, \mathbf{p_1}, \ldots, \mathbf{p_n} \in \mathbf{P}\}$$

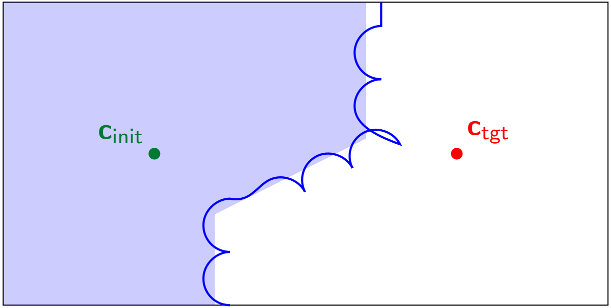
Leroux's Back-and-Forth Construction (in more detail)

Linearization: a tight over-approximation of a VAS reachability set by a semilinear set



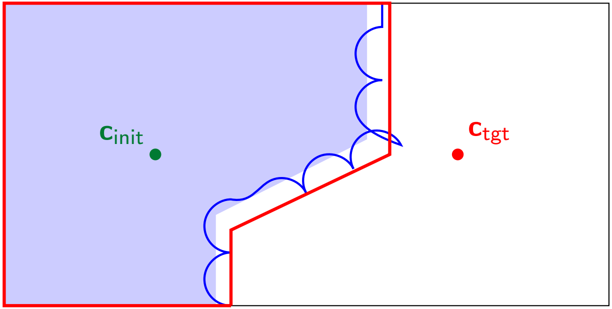
Leroux's Back-and-Forth Construction (in more detail)

Linearization: a tight over-approximation of a VAS reachability set by a semilinear set



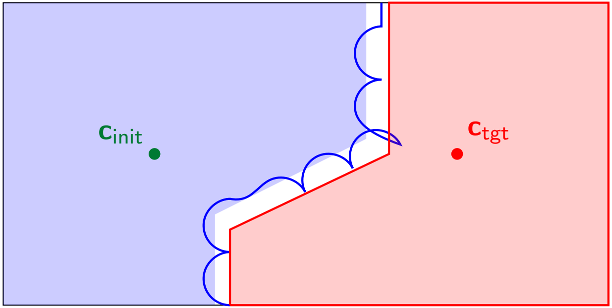
Leroux's Back-and-Forth Construction (in more detail)

Linearization: a tight over-approximation of a VAS reachability set by a semilinear set



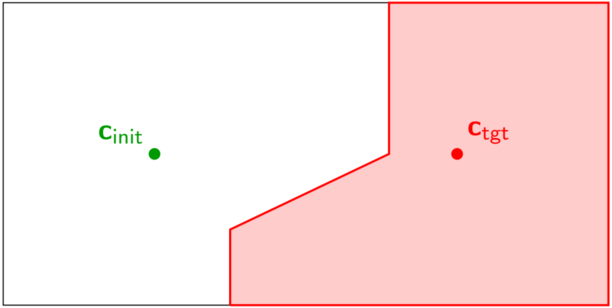
Leroux's Back-and-Forth Construction (in more detail)

Linearization: a tight over-approximation of a VAS reachability set by a semilinear set



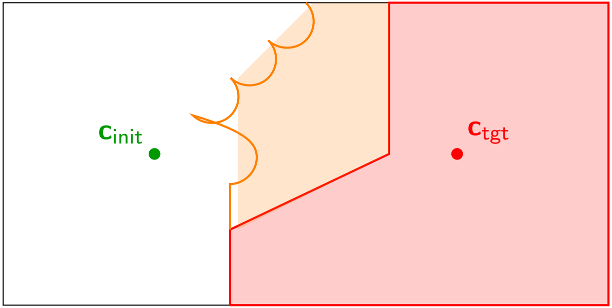
Leroux's Back-and-Forth Construction (in more detail)

Linearization: a tight over-approximation of a VAS reachability set by a semilinear set



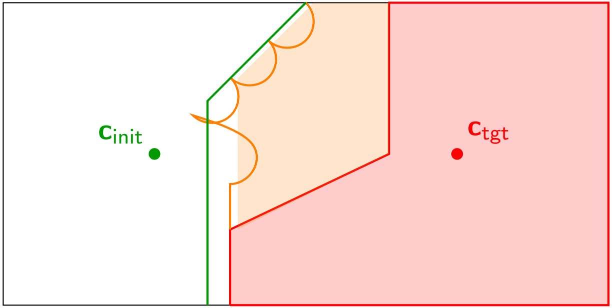
Leroux's Back-and-Forth Construction (in more detail)

Linearization: a tight over-approximation of a VAS reachability set by a semilinear set



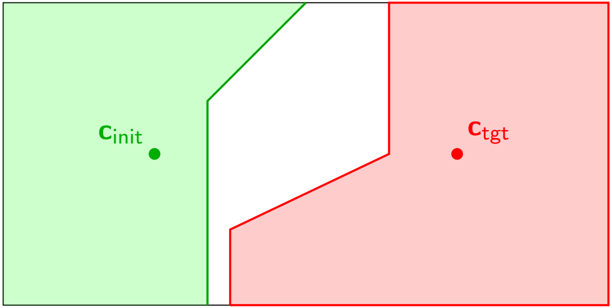
Leroux's Back-and-Forth Construction (in more detail)

Linearization: a tight over-approximation of a VAS reachability set by a semilinear set



Leroux's Back-and-Forth Construction (in more detail)

Linearization: a tight over-approximation of a VAS reachability set by a semilinear set



A Well Quasi-Order on Runs

Goal: Reachability set = finite union of $\mathbf{b} + \mathbf{P}$ where $\mathbf{b} \in \mathbb{N}^d$ and $\mathbf{P} \subseteq \mathbb{N}^d$ is a periodic set (stable by addition + contains 0).

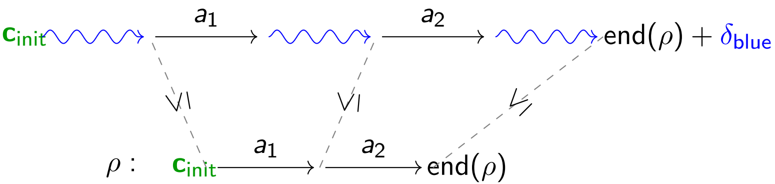
A Well Quasi-Order on Runs

Goal: Reachability set = finite union of $\mathbf{b} + \mathbf{P}$ where $\mathbf{b} \in \mathbb{N}^d$ and $\mathbf{P} \subseteq \mathbb{N}^d$ is a periodic set (stable by addition + contains 0).

$$\rho : \quad \mathbf{c}_{\text{init}} \xrightarrow{a_1} \xrightarrow{a_2} \text{end}(\rho)$$

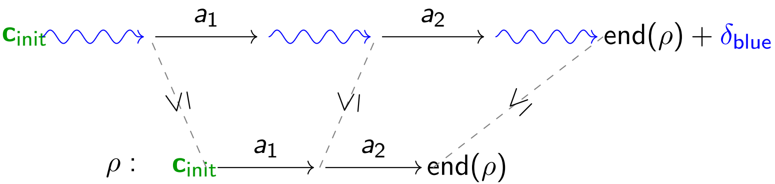
A Well Quasi-Order on Runs

Goal: Reachability set $=$ finite union of $\mathbf{b} + \mathbf{P}$ where $\mathbf{b} \in \mathbb{N}^d$ and $\mathbf{P} \subseteq \mathbb{N}^d$ is a periodic set (stable by addition $+$ contains 0).



A Well Quasi-Order on Runs

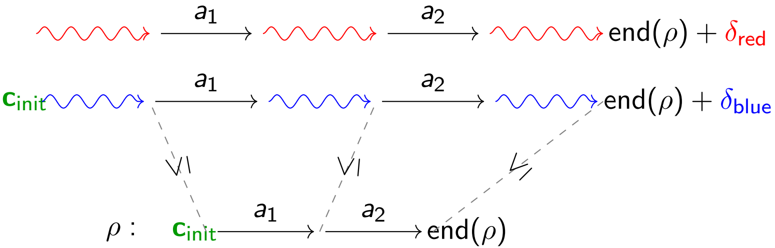
Goal: Reachability set $=$ finite union of $\mathbf{b} + \mathbf{P}$ where $\mathbf{b} \in \mathbb{N}^d$ and $\mathbf{P} \subseteq \mathbb{N}^d$ is a periodic set (stable by addition + contains 0).



Reachability set $= \bigcup_{\text{minimal } \rho} \text{end}(\rho) + \mathbf{P}_\rho$ where $\mathbf{P}_\rho := \{\text{end}(\rho') - \text{end}(\rho) \mid \rho' \geq \rho\}$

A Well Quasi-Order on Runs

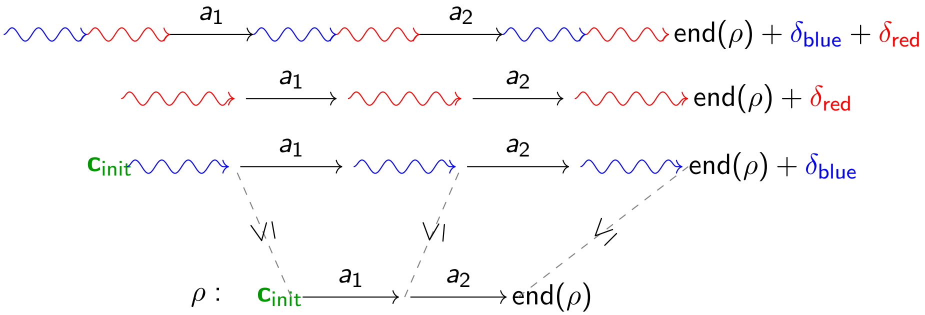
Goal: Reachability set $=$ finite union of $\mathbf{b} + \mathbf{P}$ where $\mathbf{b} \in \mathbb{N}^d$ and $\mathbf{P} \subseteq \mathbb{N}^d$ is a periodic set (stable by addition $+$ contains 0).



Reachability set $= \bigcup_{\text{minimal } \rho} \text{end}(\rho) + \mathbf{P}_\rho$ where $\mathbf{P}_\rho := \{\text{end}(\rho') - \text{end}(\rho) \mid \rho' \geq \rho\}$

A Well Quasi-Order on Runs

Goal: Reachability set $=$ finite union of $\mathbf{b} + \mathbf{P}$ where $\mathbf{b} \in \mathbb{N}^d$ and $\mathbf{P} \subseteq \mathbb{N}^d$ is a periodic set (stable by addition $+$ contains 0).



Reachability set $= \bigcup_{\text{minimal } \rho} \mathrm{end}(\rho) + \mathbf{P}_\rho$ where $\mathbf{P}_\rho := \{\mathrm{end}(\rho') - \mathrm{end}(\rho) \mid \rho' \geq \rho\}$

Linearizations

- $\mathrm{Lin}(P) := \mathrm{Gr}(P) \cap \mathrm{Cone}(P)$

Linearizations

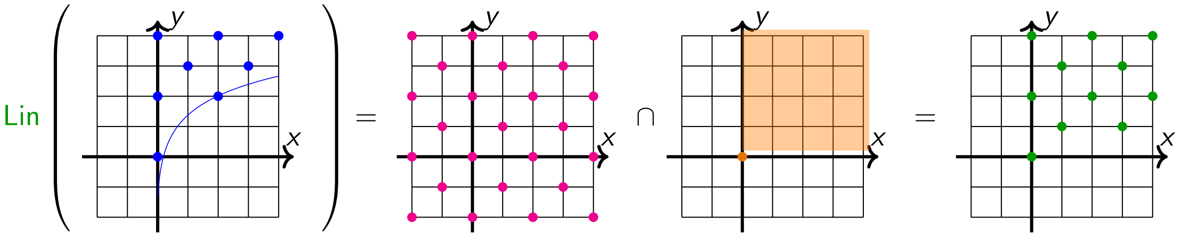
- $\mathrm{Lin}(P) := \mathrm{Gr}(P) \cap \mathrm{Cone}(P)$

Linearizations

- $\mathrm{Lin}(P) \coloneqq \mathrm{Gr}(P) \cap \mathrm{Cone}(P) = (P - P) \cap \mathbb{Q}_{\geq 0} P$

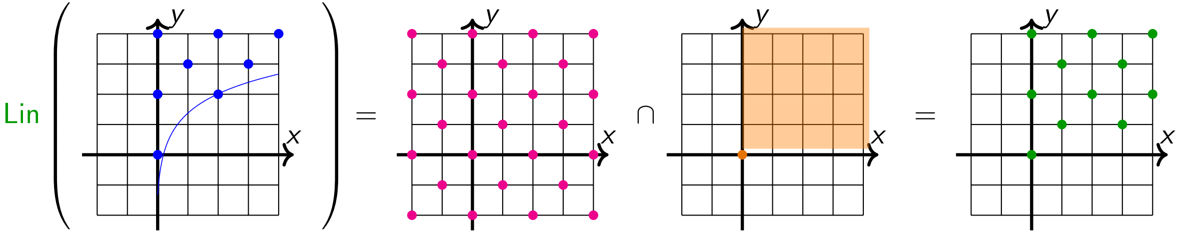
Linearizations

- $\mathrm{Lin}(P) \coloneqq \mathrm{Gr}(P) \cap \mathrm{Cone}(P) = (P - P) \cap \mathbb{Q}_{\geq 0} P$



Linearizations

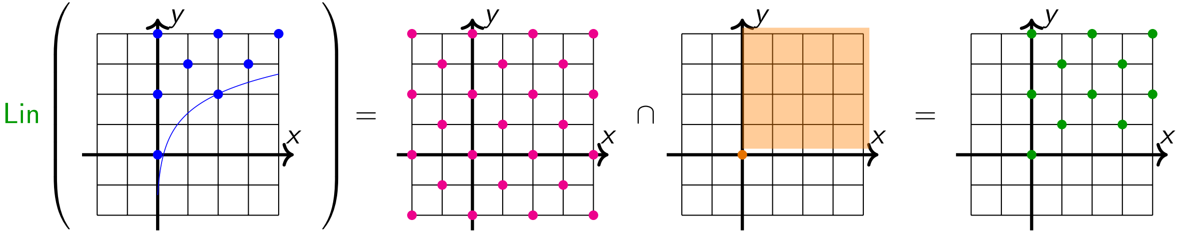
- $\mathrm{Lin}(P) \coloneqq \mathrm{Gr}(P) \cap \mathrm{Cone}(P) = (P - P) \cap \mathbb{Q}_{\geq 0} P$



- $\mathrm{Lin}(P)$ is semilinear.

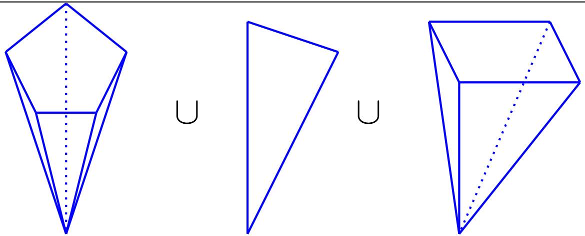
Linearizations

- $\textsf{Lin}(P) \coloneqq \textsf{Gr}(P) \cap \textsf{Cone}(P) = (P - P) \cap \mathbb{Q}_{\geq 0} P$

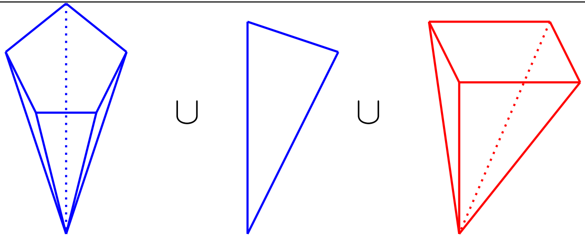


- $\textsf{Lin}(P)$ is semilinear.
- $\overline{\textsf{Lin}}(P) \coloneqq \textsf{Gr}(P) \cap \overline{\textsf{Cone}}(P) = (P - P) \cap \overline{\mathbb{Q}_{\geq 0} P}$

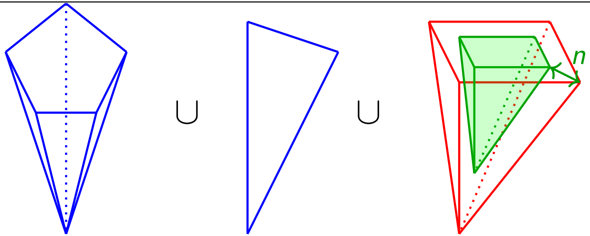
Our Forward Construction

Source	Semilinear overapproximation of Reach(Source)\Source
$\{\mathbf{c}_{\text{init}}\}$	

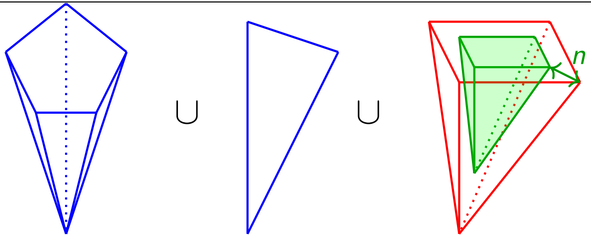
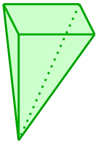
Our Forward Construction

Source	Semilinear overapproximation of Reach(Source)$\setminus$Source
$\{\mathbf{c}_{\text{init}}\}$	

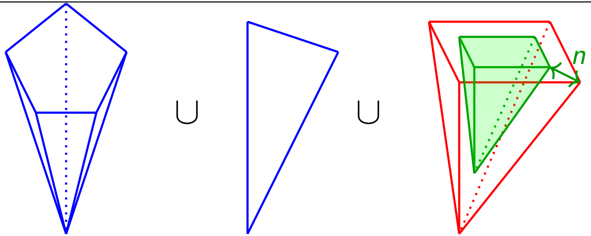
Our Forward Construction

Source	Semilinear overapproximation of Reach(Source)$\setminus$Source
$\{\mathbf{c}_{\text{init}}\}$	

Our Forward Construction

Source	Semilinear overapproximation of Reach(Source)\Source
$\{\mathbf{c}_{\text{init}}\}$	
$\{c_{\text{init}}\} \cup$ 	

Our Forward Construction

Source	Semilinear overapproximation of Reach(Source)$\setminus$Source
$\{\mathbf{c}_{\text{init}}\}$	
$\{c_{\text{init}}\} \cup$