Semi-linearity of projections of VAS Reachability sets

Summer 2020

1 Preliminaries

The sets of integers and non-negative integers are denoted as $\mathbb{Z}$ and $\mathbb{N}$. Define $[n]$ as the set $\{x \in \mathbb{N} \mid 1 \leq x \leq n\}$ for $n \in \mathbb{N}$. Vectors and sets of vectors are denoted in bold face. The i-th coordinate of a vector $\mathbf{v}$ is denoted as $\mathbf{v}[i]$. This is extended to sets of vectors as follows. Let $V \subseteq \mathbb{Z}^d$ be a (possibly infinite) set of vectors, then $V[i] := \{\mathbf{v}[i] \mid \mathbf{v} \in V\}$, which represents projection of the set V onto the coordinate i. The $\leq$ relation of $\mathbb{N}$ and $\mathbb{Z}$ is extended to $\mathbb{N}^d$ and $\mathbb{Z}^d$ as $\mathbf{u} \leq \mathbf{v}$ iff $\mathbf{u}[i] \leq \mathbf{v}[i]$ for all $i \in [d]$. Also, $\mathbf{u} < \mathbf{v}$ iff $\mathbf{u} \leq \mathbf{v}$ and $\mathbf{u} \neq \mathbf{v}$. The zero vector in $\mathbb{N}^d$ is denoted by $\mathbf{0}$.

$\mathbb{N}$ is extended to $\mathbb{N}_\omega$ using a special symbol ω which follows $n < \omega$ and $\omega + n = n + \omega = \omega$ for all $n \in \mathbb{N}$. The intuitive meaning of ω is that it signifies an arbitrarily large quantity. We similarly extend $\mathbb{N}^d$ to $\mathbb{N}^d_\omega$. For a vector $\mathbf{x} \in \mathbb{N}^d_\omega$ we define $\Omega(\mathbf{x})$ as the set of coordinates where $\mathbf{x}$ has ω. Formally, $\Omega(\mathbf{x}) = \{i \in [d] \mid \mathbf{x}[i] = \omega\}$.

1.1 Vector Addition Systems

A Vector Addition System (VAS) of dimension d is defined as $\mathcal{V} = (\mathbf{x}_0, T)$, where $\mathbf{x}_0 \in \mathbb{N}^d$ is called the *initial marking* and $T = \{\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_n\} \subseteq \mathbb{Z}^d$ is a finite set of *transitions*, or transition vectors.

A transitions sequence $\sigma = \mathbf{u}_1\mathbf{u}_2 \ldots \mathbf{u}_k$ with $\mathbf{u}_i \in T$ is *fireable from* a marking $\mathbf{y}$ if $\mathbf{y} + \sum_{j=1}^{i} \mathbf{u}_j \geq \mathbf{0}$ for all $i \in [k]$. When σ is fireable from $\mathbf{y}$, we write $\mathbf{y} \xrightarrow{\sigma} \mathbf{z}$ with $\mathbf{z} = \mathbf{y} + \sum_{j=1}^{k} \mathbf{u}_j$ and $\mathbf{z}$ is said to be *reachable* from $\mathbf{y}$ by σ. We may also write

$$\mathbf{y} = \mathbf{y}_0 \xrightarrow{\mathbf{u}_1} \mathbf{y}_1 \xrightarrow{\mathbf{u}_2} \ldots \xrightarrow{\mathbf{u}_k} \mathbf{y}_k = \mathbf{z}$$

with $\mathbf{y}_i = \mathbf{y} + \sum_{j=1}^{i} \mathbf{u}_j$. The reachability set of a VAS $\mathcal{V} = (\mathbf{x}_0, T)$ is thus defined as $Reach(\mathcal{V}) = \{\mathbf{z} \in \mathbb{N}^d \mid \exists \sigma \in \mathbf{T}^* \; \mathbf{x}_0 \xrightarrow{\sigma} \mathbf{z}\}$. The projection of the reachability set onto a coordinate i is $Reach(\mathcal{V})[i] = \{\mathbf{z}[i] \mid \mathbf{z} \in Reach(\mathcal{V})\}$.

Alain: I prefer that sets are not in bold: $Reach(\mathcal{V})$

Sai: Of course. Can we call it $\mathcal{R}$ instead? That is what I use for graphs.

Transition sequences are nothing but words over the alphabet T. Let α and β be two transition sequences. Concatenation of α and β is denoted by $\alpha\beta$, which is extended as $\prod \alpha_i$ for multiple sequences, and α^k denotes k-fold iteration of α. We use $\mathbf{\Delta}(\sigma)$ to denote sum of the vectors in the sequence $\sigma = \mathbf{u}_1\mathbf{u}_2\ldots\mathbf{u}_k$, i.e. $\mathbf{\Delta}(\sigma) = \sum_{i=1}^{k} \mathbf{u}_i$. The empty transition sequence is denoted by ϵ.

1.2 Linear and semi-linear sets

A set $L \subseteq \mathbb{N}^n$ is called *linear* if it can be represented as

$$L = L(\mathbf{x}_0, \mathbf{X}) = \{\, \mathbf{x}_0 + \sum_{i=1}^{k} \mu_i \mathbf{x}_i \mid i \in [k],\, \mu_i \in \mathbb{N}^d,\, \mathbf{x}_i \in \mathbf{X})\}$$

where $\mathbf{x}_0 \in \mathbb{N}^d$ and $\mathbf{P} \subseteq \mathbb{N}^n$ is a finite set. The vector $\mathbf{x}_0$ is called the base vector (or simply the base) and the vectors $\mathbf{x}_i \in \mathbf{X}$ are called period vectors (or simply the periods). A set S is called *semi-linear* if it is a finite union of linear sets. Semi-linear sets are closed under union, intersection and complementation [1].

1.3 Dickson's lemma

We state without proof a well-known lemma from combinatorics.

Lemma 1 (Dickson)**.** *Let $\mathbf{x}_1, \mathbf{x}_2, \ldots$ be an infinite sequence of vectors from $\mathbb{N}_\omega^d$. Then there exist two indices i and j with $i < j$ such that $\mathbf{x}_i \leq \mathbf{x}_j$.*

2 A bit of Graph Theory

Sai: Rewriting the Konig's lemma part. The previous attempt is commented out as of now and I will erase it once you say ok.

Recall that a (directed) *graph* G is a pair (P, E) where P is a set of nodes (or places) and $E \subseteq P \times P$ is a set of edges. We alternatively write (p, q) or $p \rightarrow_G q$ and we denote $\xrightarrow{*}_G$ for the reflexive and transitive closure of $\rightarrow_G$. When G is implicit we write $\rightarrow$ and $\xrightarrow{*}$. A graph is said to be infinite iff P is infinite. A graph may also permit a labeling over its nodes and/or edges. A graph $G' = (P', E')$ is said to be a *subgraph* of a graph $G = (P, E)$ if $P' \subseteq P$ and $E' \subseteq (P' \times P' \cap E)$.

Sai: Now all this can even be in paragraphs instead of bullet points. This felt more organized while writing.

Alain: let like that, it is clear

Sai: Ok

Let G be a graph and p, q two nodes in G.

Alain: I think the best is to define graph with labeled edges because this is the object that we will use in the future as the graph defined by Mayr or the Karp-Miller graph...An unlabeled graph, if we need this notion, is easy to define from labeled graphs and it will not be necessary to redefine notions (if we need)

Sai: Hmm, I will modify things bit then.

- A *path* from a node p to a node q of length n is a sequence of nodes $p_0, p_1, \ldots, p_n$ such that $p = p_0$ and $q = p_n$ and $(p_{i-1}, p_i) \in E$ for every $i \in [n]$. The length of a path ... is n.

  Alain: a path is not totally defined by its sequence of nodes: suppose that between two nodes there exist three edges, which edge do you take for going from p to q ?. Usually and I think it is necessary, we say that a path is $p --> q --> r ... --> z$

  Sai: No but our graph only has an edge set ... so either there is an edge (p,q) or there is not. It is, to be precise, a simple graph.

- A *elementary path* in G is a path $p_0, p_1, \ldots, p_n$ such that all nodes are distinct.

- A *cycle* in G is a path $p_0, p_1, \ldots, p_n$ such that $p_0 = p_n$.

- A *loop* is a cycle of length 1, i.e a path $p \to p$.

  Sai: So a cycle may have intermediate nodes repeating? I was taught a chain like system isn't called a cycle.

  Alain: the terminology is not totally universal: we can use, circuit, loop or cycle, what is important is to define them

  Sai: Now here we need to differentiate between edge and path when we use the $\to$.Below I used $\to_G$ for reachable, but maybe I should use that for the edge relation and use something like $\to_G^*$ or $\overset{*}{\to}_G$ for reachable?

  Alain: yes $\overset{*}{\to}_G$ is the transitive and reflexive closure of $\to_G$

- We say q is *reachable* from p in G if there exists a walk from p to q, and denote it as $p \overset{*}{\to}_G q$. We then naturally define $\mathcal{R}^*(p) = \{q \in P \mid p \overset{*}{\to}_G q\}$ as the set of nodes reachable from a node p.

- We say a graph is *rooted* if there is a node $r \in P$ such that every node is reachable from it, i.e. $\mathcal{R}^*(r) = P$. The node r is called a root of the graph. Observe that if G is a rooted graph with no cycles (a rooted tree) then $\mathcal{R}^*(p)$ represents the nodes of the subtree of G rooted at p.

- We define $\mathcal{S}(p) = \{q \in P \mid (p, q) \in E\}$ as the *successor* set of the node p.

- A graph is said to be *finitely branching* if $\mathcal{S}(p)$ is finite for every $p \in P$.

We are now in a position to state and prove a lemma which is a variant of the well-known König's lemma.

Alain: it is important to quote this result if it is known

Sai: But nowhere can I find it worded as I need it.

Alain: ok, I will look if it exists for rooted graph

Lemma 2. *Every finitely branching infinite rooted graph contains an infinite path $p_0, p_1, ..., p_n, ...$ where $p_i \neq p_j$ for all $i \neq j$.*

Proof. Consider a infinite rooted digraph $G = (P, E)$ along with a root $r \in P$. We will construct, step by step, an infinite path $p_0, p_1, ..., p_n, ...$ where $p_i \neq p_j$ for all $i \neq j$. Let us construct p_1 as follows. Since G is finitely branching, $\mathcal{S}(r)$ is finite. Since the set of nodes P is infinite, there must exist a node $p_1 \in \mathcal{S}(r)$ such that $p_1 \neq r$ and $\mathcal{R}^*(p_1)$ is infinite (because $P = \bigcup_{p \in \mathcal{S}(r)} \mathcal{R}^*(p)$).

Alain: if you don't precise $p_1 \neq r$ then it is possible that $p_1 = r$ and then you obtain the infinite path $r, r, r...r$

We now suppose (induction hypothesis) that for $k \geq 1$, we have construct a path $r, p_1, \ldots, p_k$ such that $\mathcal{R}^*(p_k)$ is infinite and all nodes $r, p_1, \ldots, p_k$ are pairwise different. Let us show how to construct the following node p_{k+1}. Let us remark that the set $\{p \in \mathcal{S}(p_k) \mid \mathcal{R}^*(p) \text{ is infinite and } p \notin \{r, p_1, ..., p_k\}\}$ is finite and not empty. So pick an element in this set and call it p_{k+1}. Now by using the axiom of choice, we are able to construct the infinite path $r, p_1, \ldots, p_k, \ldots$ that satisfies $p_i \neq p_j$ for all $i \neq j$.

Sai: Yes, I was sure my proof was missing something. I don't understand what you mean, nor do I know what the Axiom of Choice is exactly. Although in each iteration we increase the length of our path by 1 ... so how the limiting length of this sequence of paths be finite?

■

Sai: How is this whole section?

Alain: this section is nice and much better than before, you can stop it because now I would like to finish other sections and also to make new original results

3 The Reachability Graph

Alain: the expression "Reachability Graph" choosen by the authors is bad because the reachability graph is always defined as the graph where the set of nodes is the set of reachable configurations ! Mayr defined a variation of the Karp-Miller graph (is it unique ?) so we will define it as a Karp-Miller graph. it is not unique because we don't accelerate all possible sequences and the choice is not deterministic...If no big problem it would be better to use the (unique) KMG with all possible accelerations that are made in the original KM algorithm, look the original paper page 166 (i)

Sai: Ok I agree. Yes this graph is not unique. I saw the original paper and thought of using all accelerations but if we do that then we lose the crucial property of lemma 3. We won't be able to create an ω-cycle.

We now give an algorithm to construct a rooted graph $\mathbb{G}$ for a VAS $\mathcal{V} = (\mathbf{x}_0, T)$ which is similar to the reachability tree constructed in [2]. The graph $\mathbb{G}$ is described by a tuple $(P, E, \boldsymbol{\lambda})$ where P is a set of nodes or places, $E \subseteq P \times P$ is a set of edges, and $\boldsymbol{\lambda} : P \cup E \to \mathbb{N}_\omega^n$ is a labeling on the nodes and edges by markings and transitions respectively. By concatenation of edge labels, we may also label walks of G with transition sequences.

Alain: it could be good to label a cycle with a word

Sai: Ok, I label walks with transition sequences

3.1 Description of the algorithm

The algorithm initialises $\mathbb{G}$ with a node r, which is a root, and no edges. The label of r is $\mathbf{x}_0$. Then it tries firing transitions from the labels of nodes picked from a set $P^* \subseteq P$ which represents the "nodes to be explored". Let the algorithm pick a node $p \in P^*$ such that $\boldsymbol{\lambda}(p) = \mathbf{x}$ and let $\mathbf{u} \in T$ be fireable from $\mathbf{x}$. If there is no edge $p \xrightarrow{\mathbf{u}} p'$

Alain: I prefer "if there is no edge $p \xrightarrow{\mathbf{u}} p'$"

Sai: I like it too, but see what I say under the point about a cycle in the previous section.

then we add a node q and an edge $p \xrightarrow{\mathbf{u}} q$ with $\boldsymbol{\lambda}(q) = \mathbf{x} + \mathbf{u}$

Alain: I prefer: "then we add a node q and an edge $p \xrightarrow{\mathbf{u}} q$ with $\boldsymbol{\lambda}(q) = \mathbf{x} + \mathbf{u}$"

This procedure would build an infinite rooted tree if not for the following two actions.

1. If we discover a node q by firing $\mathbf{u}$ from p as above, and q' has an ancestor q such that $\boldsymbol{\lambda}(q) = \boldsymbol{\lambda}(q')$, we erase the node q from $\mathbb{G}$ and simply loop back from p, adding an edge $p \xrightarrow{\mathbf{u}} q'$ This forms a new cycle in the graph $\mathbb{G}$.

2. I retry (old one is commented out): If on the path from r to p we find a node q', i.e if $r \to_G q' \to_G p$, then we delete q from P and (p, q) from E, and add a an edge (p, q') to $\mathbb{G}$ with the label $\mathbf{u}$. This creates a cycle, and the newly added edge is called a *back edge*.

   Alain: new point 1 is unclear than before and contains typos. Keep old point 1

3. I retry : If on the path from r to p we find a node q', i.e if $r \to_G q' \to_G p$, such that $\boldsymbol{\lambda}(q') < \boldsymbol{\lambda}(q)$ and $\Omega(\boldsymbol{\lambda}(q')) = \Omega(\boldsymbol{\lambda}(q))$ then we re-label q with $\mathbf{x}$ such that $\mathbf{x}[i] = \omega$ when $\boldsymbol{\lambda}(q')[i] < \boldsymbol{\lambda}(q)[i]$ and $\mathbf{x}[i] = \boldsymbol{\lambda}(q)[i]$ otherwise. If there are

multiple nodes which qualify as q', the algorithm chooses one at random. Such a node q is called an *ω-node*. The set of new coordinates of the marking $\lambda(q)$) is denoted by $\Omega'(\lambda(q))$. We also add a cycle as described by algorithm 1 using the label of the path from q' to q. This cycle is called an *ω-cycle*.

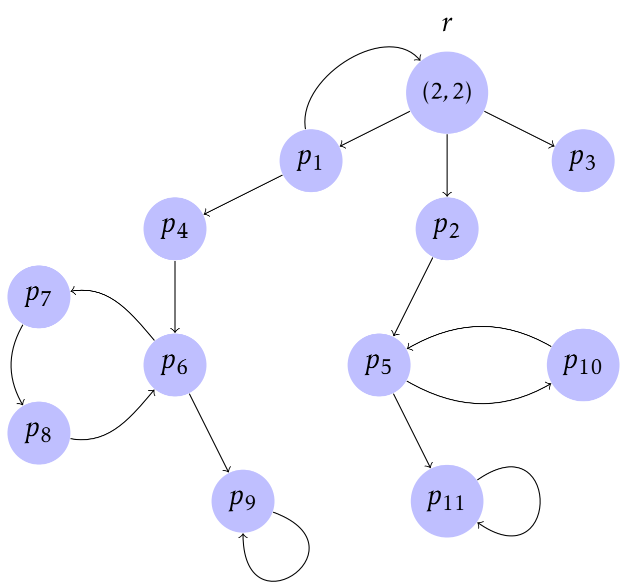
Alain: good

Sai: I will add an example graph here. One showing all the cases

Alain: nice but the reading of the graph is complex since we must go from p_i to its label. I prefer to put in the circle the label and to write outside the circle the number of the node.

Sai: Yes ok.

Alain: begin with an example of VAS with power of 2 at the very beginning and illustrate notions on the example



$\lambda(r) = (2,2)$
$\lambda(p_1) = (3,1)$
$\lambda(p_2) = (1,4)$
$\lambda(p_3) = (1,3)$
$\lambda(p_4) = (4,0)$
$\lambda(p_5) = (1,\omega)$
$\lambda(p_6) = (\omega,2)$
$\lambda(p_7) = (\omega,1)$
$\lambda(p_8) = (\omega,0)$
$\lambda(p_9) = (\omega,\omega)$
$\lambda(p_{10}) = (1,\omega)$
$\lambda(p_{11}) = (\omega,\omega)$

Figure 1: This figure shows a subgraph of the Reachability graph $\mathbb{G}$ for a VAS $\mathcal{V} = (\mathbf{x}_0, \{\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3\})$ with $\mathbf{x}_0 = (2,2)$ and $\mathbf{t}_1 = (1,-2)$, $\mathbf{t}_2 = (-1,1)$, $\mathbf{t}_3 = (1,-1)$.

Algorithm 1: Adding an ω-cycle to $\mathbf{G}$

Input: A labeled rooted graph $\mathbb{G} = (P,E)$, a node $q \in P$, an transition sequence γ

Output: The same labeled rooted graph $\mathbb{G} = (P,E,\lambda)$, modified

Algorithm 2: A Reachability Graph

Input: A Vector Addition System $\mathcal{V} = (\mathbf{x}_0, \mathbf{T})$
Output: A labeled graph $\mathbb{G} = (P, E, \lambda)$

$P \leftarrow \{r\}$; $E \leftarrow \phi$; $P^* \leftarrow \{r\}$; $\lambda(r) \leftarrow \mathbf{x}$;
while $P^* \neq \emptyset$ **do**
  pick out some $p \in P^*$;
  foreach $\mathbf{v} \in \mathbf{V}$ **do**
    if $\mathbf{v} + \lambda(p) \in \mathbb{N}^n_\omega$ **and** $\nexists q \in P$ $s.t$ $\lambda(p,q) = \mathbf{v}$ **then**
      if $\exists q' \in \mathtt{Path}(r,p)$ $s.t$ $\lambda(q') = \mathbf{v} + \lambda(p)$ **then**
        $E = E \cup (p, q')$; $\lambda(p, q') = \mathbf{v}$; // We add a back edge
      else
        $q \leftarrow \mathtt{NewNode}()$; $P \leftarrow P \cup \{q\}$;
        $E \leftarrow E \cup \{(p,q)\}$; $\lambda(q) \leftarrow \mathbf{v} + \lambda(p)$;
        $\lambda(p,q) \leftarrow \mathbf{v}$; $P^* \leftarrow P^* \cup q$;
        if $\exists q' \in \mathtt{Path}(r,p)$ $s.t$ $\lambda(q') < \mathbf{y}$ **and** $\Omega(\lambda(q')) = \Omega(\mathbf{y})$ **then**
          // q is called an ω-node
          foreach $i \in [d]$ **do**
            if $\lambda(q)[i] > \lambda(q')[i]$ **then** $\lambda(q)[i] \leftarrow \omega$;
          end
          // We add an ω-cycle at q using the label of $\mathtt{Path}(q', q)$
          $\gamma \leftarrow \mathtt{EdgeSeq}(q', q)$; $\mathtt{AddCycle}(q, \gamma)$;
        end
      end
    end
  end
end
return $\mathbb{G}$

3.2 Proof of termination

Theorem 1. *Algorithm 2 terminates.*

Proof. To show termination of the algorithm is to show that it generates a finite graph. The graph $\mathbb{G}$ is by definition rooted, and it is finite branching because every node can have at most T successors. Now due to Lemma 2 we just need to show there is no infinite walk in $\mathbb{G}$.

Now, observe that ω's are never lost, i.e. if $p \to_{\mathbb{G}} q$ then $\Omega(\lambda(p)) \subseteq \Omega(\lambda(q))$. So it suffices to show there is no infinite path

Alain: incorrect: say no infinite path $\sigma'.\sigma$ st. σ .....

with all nodes having the same Ω of their labels, because the dimension d of $\mathcal{V}$

is finite.

Let us suppose, for the sake of contradiction, that $p_0, p_1, p_2 \dots$ is an infinite path.

Alain: $\sigma'.\sigma$ with σ infinite path and ...

Sai: What is σ and what is σ'?

Alain: st $\Omega(p_i) = \Omega(p_j)$ for all i, j and $|\Omega(p_i)| = k$

Let $\alpha = \mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \dots$ be a sequence of vectors from $\mathbb{N}^{d-|\Omega(p_i)|}$,

Alain: $\mathbb{N}^{d-k}$

where for each i, $\mathbf{x}_i$ is the projection of $\lambda(p_i)$ onto its finite coordinates.

Alain: a definition is missing: sup(v)

Sai: ??

Now we apply Lemma 1 to infer the existence of i and j such that $i < j$ and $\mathbf{x}_i \leq \mathbf{x}_j$. This is disallowed by the description of the algorithm ($\mathbf{x}_i = \mathbf{x}_j$ by the first action and $\mathbf{x}_i < \mathbf{x}_j$ by the second),

Alain: be more precise

so no infinite path exists in $\mathbb{G}$.

■

4 Some useful results

Sai: What say we define a run here? I don't know how we will define it, but an example of what I mean by a run is $\mathbf{x} = \mathbf{y}_0 \xrightarrow{\mathbf{u}_1} \mathbf{y}_1 \xrightarrow{\mathbf{u}_2} \dots \xrightarrow{\mathbf{u}_n} \mathbf{y}_n = \mathbf{y}$. This is drawing from what I know about a run in an automata.

Alain: perfect

Lemma 3. *If $\mathbf{x} \xrightarrow{\sigma} \mathbf{y}$ for some $\sigma = \mathbf{u}_1 \mathbf{u}_2 \dots \mathbf{u}_n$ and $\mathbf{x} = \lambda(p)$ for some node $p \in P$, then there is a unique walk in $\mathbb{G}$ from p to a node $q \in P$ labeled with σ. Also, $\lambda(q) \geq \mathbf{y}$.*

Sai: This proof is now complete. Please check.

Proof. Let $\mathbf{x} = \mathbf{y}_0 \xrightarrow{\mathbf{u}_1} \mathbf{y}_1 \xrightarrow{\mathbf{u}_2} \dots \xrightarrow{\mathbf{u}_n} \mathbf{y}_n = \mathbf{y}$ in $\mathcal{V}$ and $q_0, q_1, \dots, q_n$ be the σ-labeled walk in $\mathbb{G}$ with $p = q_0$, $q = q_n$, whose existence we shall prove.

We use strong induction on n. Let the base case be $n = 0$, so the lemma is vacuously true. For the sake of induction we assume the lemma is true for all $n \leq k$. The induction hypothesis gives us $\lambda(q_k) \geq \mathbf{y}_k$, and $\mathbf{u}_{k+1}$ is fireable from $\mathbf{y}_k$, so $\mathbf{u}_{k+1}$ is fireable from $\lambda(q_k)$.

If $\lambda(q_k) + \mathbf{u}_{k+1} = \lambda(q_j)$ for some $j \leq k$, then then we extend our walk with $q_{k+1} = q_j$. $\lambda(q_{k+1}) \geq \mathbf{y}_{k+1}$ because $q_{k+1} = q_j$, $\lambda(q_j) \geq \mathbf{y}_j$, and $\mathbf{y}_{k+1}$ differs from $\mathbf{y}_j$ only at the non-ω coordinates of $\lambda(q_j)$. Else, the algorithm adds an edge (q_k, q_{k+1}) labeled with

$\mathbf{u}_{k+1}$. If there is no $j \leq k$ such that $\boldsymbol{\lambda}(q_k) + \mathbf{u}_{k+1} > \boldsymbol{\lambda}(q_j)$, then $\boldsymbol{\lambda}(q_{k+1}) = \boldsymbol{\lambda}(q_k) + \mathbf{u}_{k+1} \geq \mathbf{y}_k + \mathbf{u}_{k+1} = \mathbf{y}_{k+1}$. If there is such a j then by construction $\boldsymbol{\lambda}(q_{k+1}) \geq \boldsymbol{\lambda}(q_k) + \mathbf{u}_{k+1} \geq \mathbf{y}_{k+1}$. In any case, we can extend our walk with q_{k+1} and have $\boldsymbol{\lambda}(q_{k+1}) \geq \mathbf{y}_{k+1}$.

So there exists a walk $q_0, q_1, \ldots, q_n$ with label σ, $p = q_0$, $q = q_n$, and $\boldsymbol{\lambda}(q) \geq \mathbf{y}$ ∎

Let p be an ω-node of G with γ as the label of its ω-cycle. Let $\Omega'(\boldsymbol{\lambda}(p)$ be the set of new ω-coordinates of the marking , and $\Omega(\boldsymbol{\lambda}(p)$ the set of all ω-coordinates, as before. We now describe a property of the transition sequence γ.

Lemma 4.

$$\begin{array}{ll} \boldsymbol{\Delta}(\gamma)[i] > 0 & \textit{for all } i \in \Omega'(\boldsymbol{\lambda}(p) \\ \boldsymbol{\Delta}(\gamma)[i] = 0 & \textit{for all } i \in [d] \setminus \Omega(\boldsymbol{\lambda}(p) \\ \boldsymbol{\Delta}(\gamma)[i] \textit{ is arbitrary} & \textit{otherwise} \end{array}$$

Proof. Because of the definition of an ω-cycle, there must be a node q on the path from r to p in $\mathbb{G}$ such that $\boldsymbol{\lambda}(p) > \boldsymbol{\lambda}(q)$ and γ is the label of the path from q to p. Let $\mathbf{x} = \boldsymbol{\lambda}(q) + \boldsymbol{\Delta}(\gamma)$. When $\mathbf{x}[i] > \boldsymbol{\lambda}(q)[i]$, by definition, ω is introduced. So $\boldsymbol{\Delta}(\gamma)[i] > 0$ for $i \in \Omega'(\boldsymbol{\lambda}(p)$. When $\mathbf{x}[i] = \boldsymbol{\lambda}(q)[i] \neq \omega$, $\boldsymbol{\Delta}(\gamma) = 0$, and when $\mathbf{x}[i] = \boldsymbol{\lambda}(q)[i] = \omega$, $\boldsymbol{\Delta}(\gamma)$ can be any integer, as $\omega + n = \omega$ for any $n \in \mathbb{Z}$. $\mathbf{x}[i] < \boldsymbol{\lambda}(q)[i]$ is not possible because $\mathbf{x} \geq \boldsymbol{\lambda}(q)$. ∎

5 A periodic propery

Sai: Work left here. But I think I will read some stuff first and then work on this lemma. No point thinking on the same lines. So I leave this section mostly as it is.

Let p be an ω-node in $\mathbb{G}$. Suppose there are exactly $k \geq 0$ other ω-nodes on the simple path from r to p so that we have $r = p_0 \xrightarrow{\alpha_1} p_1 \xrightarrow{\alpha_2} \ldots \xrightarrow{\alpha_k} p_k \xrightarrow{\alpha_{k+1}} p_{k+1} = p$.

Let γ_i be the sequence of edge labels on the ω-cycle of the ω-node p_i and γ the same for p. Define $\eta := \sum_{i=1}^{k} n_i \boldsymbol{\sigma}(\gamma_i) + \boldsymbol{\sigma}(\gamma)$ and $\beta(t) := (\Pi_{i=1}^{k} \alpha_i \gamma_i^{tn_i+b_i})\alpha\gamma^t$.

Sai: Work left here

Alain: more simple: Fix now some $n_1, n_2, \ldots, n_k$ and let $\eta := \sum_{i=1}^{k+1} n_i \times \boldsymbol{\Delta}(\gamma_i)$ with $n_{k+1} = 1$ or may be better $\eta(n_1, n_2, \ldots, n_k) := \sum_{i=1}^{k} n_i \times \boldsymbol{\Delta}(\gamma_i) + \boldsymbol{\Delta}(\gamma_{k+1})$ and explaining why the coefficient of γ_{k+1} is 1. and write shortly when no confusion η. But I dislike η for a vector of integers, it is too close from β.

Sai: Why is coefficient of γ_{k+1}1?

Alain: because $n_{k+1} = 1$ because we don't iterate the last cycle because....

Sai: Because?

Lemma. *We break the lemma into two parts. With the above definition in context, there exist natural numbers $n_1, \ldots, n_k$ and $b_1, \ldots, b_k$ such that*

(a) $\eta[i] > 0 \quad \forall i \in \Omega(\boldsymbol{\lambda}(p))$ *and*
(b) $\beta(t)$ *is applicable to* $\mathbf{x} \quad \forall t \geq 0$.

Proof. We shall prove this by induction on k. Base case is $k = 0$. Part (a) follows from lemma 3.4 (will be renamed appropriately later) as $\eta = \boldsymbol{\sigma}(\gamma)$, and $\beta(t) = \alpha\gamma^t$ is applicable to $\mathbf{x}$ because there are no ω-nodes above p in $\mathbb{T}(\mathbf{x}, \mathbf{V})$, hence part (b) also stands true.

The induction hypothesis gives us $n'_1, \ldots, n'_{k-1}$ and $b'_1, \ldots, b'_{k-1}$ such that $\eta'[i] > 0 \ \ \forall\ i \in \Omega(\boldsymbol{\lambda}(p_k))$ and $\beta'(t)$ is applicable on $\mathbf{x}$, where $\eta' = \sum_{i=1}^{k-1} n'_i \boldsymbol{\sigma}(\gamma_i) + \boldsymbol{\sigma}(\gamma_k)$, and $\beta'(t) = (\Pi_{i=1}^{k-1} \alpha_i \gamma_i^{tn'_i + b'_i})\alpha_k \gamma_k^t$.

Now observe $\Omega(\boldsymbol{\lambda}(p_k)) \subseteq \Omega(\boldsymbol{\lambda}(p))$, so there exists a minimal $n' \in \mathbb{N}$ such that $\eta := n'\eta' + \boldsymbol{\sigma}(\gamma)$ satisfies $\eta[i] > 0 \ \ \forall\ i \in \Omega(\boldsymbol{\lambda}(p_k))$. Further $\boldsymbol{\sigma}(\gamma) > 0 \ \forall\ i \in \Omega(\boldsymbol{\lambda}(p_k)) \setminus \Omega(\boldsymbol{\lambda}(p))$ because of lemma 3.4. But $\eta'[i] = 0 \ \ \forall\ i \notin \Omega(\boldsymbol{\lambda}(p_k))$ because lemma 3.4 tells us that vectors in a cycle add up to 0 at non-ω coordinates for all ω-nodes and η' is just a (weighted) sum of cycles. Therefore $\eta[i] > 0 \ \forall\ i \in \Omega(\boldsymbol{\lambda}(p))$. This completes the proof for part (a) using $n_i := n'n'_i \ \forall\ i \in [k-]$ and $n_k := n'$.

Onto part (b). There is a minimal $t' \in \mathbb{N}$ such that $\alpha\gamma$ is applicable to $\mathbf{x} + \boldsymbol{\sigma}(\beta'(t'))$. We construct $\beta(t) := \beta'(n't + t')\alpha\gamma^t$. For $t \leq 1$ it is evident by construction (HOW?) so we need to prove for $t > 0$. To this end, observe that

$$\mathbf{x} + \boldsymbol{\sigma}(\beta'(n't + t')\alpha\gamma^{t-1} \geq \mathbf{x} + \boldsymbol{\sigma}(\beta'(n'(t-1) + t')\alpha\gamma^{t-2}$$

and since γ is applicable to the former so it is to the latter. Therefore $\forall\ t \geq 0\ \beta(t)$ is applicable to $\mathbf{x}$, and $b_i := n'_i t' + b'_i \ \forall\ i \in [k-1]$ and $b_k := t'$ satisfy the lemma.

This finishes the proof. ■

6 A result about semi-linear sets

We now prove a result which will finally bring semi-linear sets into the picture, and be the last lemma we need to prove semi-linearity of $Reach(\mathcal{V})[i]$ (projection of the Reachability set).

Lemma 5. *Let* $S \subseteq \mathbb{N}$ *such that there exist* $a, b \in \mathbb{N}$, $b > 0$ *and* S *follows for all* $s \geq a$ $s \in S \implies s + b \in S$. *Then* S *is a semi-linear set.*

Proof. Consider the elements of S which lie in $[a, a+b)$ and put them in a set C, i.e. $C = \{s \in S \mid a \leq s < a + b\}$. Now for each element $c \in C$ we construct a linear set $L(c, \{b\}) = c + \mathbb{N}b = \{c + nb \mid n \in \mathbb{N}\}$.

These sets are all disjoint. We prove this by contradiction. Suppose there exist $c_1, c_2 \in C$ such that $c_1 \neq c_2$ and $L(c_1, \{b\}) \cap L(c_2, \{b\}) \neq \phi$. Then there exist $n_1, n_2 \in \mathbb{N}$ such that $c_1 + n_1 b = c_2 + n_2 b$. This can be rewritten as $c_1 - c_2 = (n_2 - n_1)b$. But

$|c_1 - c_2| < b$. So it follows that $n_1 = n_2$ and $c_1 = c_2$, which is a contradiction, so the sets are disjoint.

Now observe that every $s \in S_{\geq a} = \{s \in S \mid s \geq a\}$ can be written as $c + nb$ for some $c \in C$ so

$$\bigcup_{c \in C} L(c, \{b\}) = S_{\geq a}$$

Also, $S_{<a} = \{s \in S \mid s < a\}$ is a finite set, so it can be represented as

$$S_{<a} = \bigcup_{s \in S_{<a}} L(s, \{0\})$$

This shows $S_{\geq a}$ and $S_{<a}$ are semi-linear sets. Therefore S is also a semi linear set, because semi-linear sets are closed under union.

■

Sai: Is it too long or complicated? I defined $S_{\geq a}$ on the fly :p Maybe S^+ and S^- would be better?

References

[1] Seymour Ginsburg. *The Mathematical Theory of Context-Free Languages*. McGraw-Hill, USA, 1966.

[2] Richard M. Karp and Raymond E. Miller. Parallel program schemata. *J. Comput. Syst. Sci.*, 3(2):147–195, May 1969.