PIEO Trees for Fun and Profit

We assume familiarity with [MLF$^+$23], adopt its notational conventions, and borrow many of its definitions!

1 Structure & Semantics

Definition 1.1. For sets S, D, and predicates F over D, let $\textbf{PIEO}(S, D, F)$ denote the set of *PIEOs* that

(1) hold entries in S, decorated with meta-data in D
(2) are ordered by **Rk**
(3) support predicates in F
(4) admit partial functions

$$\text{pop} : \textbf{PIEO}(S, D, F) \times F \rightharpoonup S \times \textbf{PIEO}(S, D, F)$$
$$\text{push} : \textbf{PIEO}(S, D, F) \times S \times D \times \textbf{Rk} \to \textbf{PIEO}(S, D, F)$$
$$\text{proj} : \textbf{PIEO}(S, D, F) \times F \to \textbf{PIFO}(S)$$

Maps push and pop are as usual. The *projection* $\text{proj}(p, f)$ is the PIFO of entries in p with data satisfying f. These three maps play nicely together:

$$\text{pop}(p, f) \text{ is undefined} \iff \text{pop}(\text{proj}(p, f)) \text{ is undefined} \tag{1}$$

$$\text{pop}(p, f) = (\text{pkt}, p') \implies \text{pop}(\text{proj}(p, f)) = (\text{pkt}, \text{proj}(p', f)) \tag{2}$$

$$\text{proj}(\text{push}(p, s, d, r), f) = \begin{cases} \text{push}(\text{proj}(p, f), s, r) & f(d) \text{ holds true} \\ \text{proj}(p, f) & \text{otherwise} \end{cases} \tag{3}$$

We consider PIEOs p, p' equal if, for all $f \in F$, $\text{proj}(p, f) = \text{proj}(p', f)$, i.e. their projections are always equal. For PIEO p, entry $s \in S$, and predicate $f \in F$, we write

(1) $|p|$ for the number of entries in p
(2) $|p|_f$ for the number of entries in p satisfying f
(3) $|p|_{s,f}$ for the number of times s occurs in p with associated $d \in D$ such that $f(d)$ holds

We fix an opaque set **Data** and a collection $\mathcal{F}$ of predicates defined on it. These predicates come with a total order $\leq$ and the property that, $\forall d \in \textbf{Data}$ and $f, f' \in \mathcal{F}$, $f \leq f' \wedge f(d) \implies f'(d)$.

Definition 1.2. The set of *PIEO trees* over $t \in \textbf{Topo}$, denoted $\textbf{PIEOTree}(t)$, is defined inductively by

$$\frac{p \in \textbf{PIEO}(\textbf{Pkt}, \textbf{Data}, \mathcal{F})}{\text{Leaf}(p) \in \textbf{PIEOTree}(*)} \qquad \frac{n \in \mathbb{N} \quad ts \in \textbf{Topo}^n \quad p \in \textbf{PIEO}(\{1, \ldots, n\}, \textbf{Data}, \mathcal{F}) \quad \forall i \in [1, n].\ qs[i] \in \textbf{PIEOTree}(ts[i])}{\text{Internal}(qs, p) \in \textbf{PIEOTree}(\text{Node}(ts))}$$

Definition 1.3. Define pop : $\textbf{PIEOTree}(t) \times \mathcal{F} \rightharpoonup \textbf{Pkt} \times \textbf{PIEOTree}(t)$ by

$$\frac{\text{pop}(p, f) = (\text{pkt}, p')}{\text{pop}(\text{Leaf}(p), f) = (\text{pkt}, \text{Leaf}(p'))} \qquad \frac{\text{pop}(p, f) = (i, p') \quad \text{pop}(qs[i], f) = (\text{pkt}, q')}{\text{pop}(\text{Internal}(qs, p), f) = (\text{pkt}, \text{Internal}(qs[q'/i], p'))}$$

Definition 1.4. Define push : $\textbf{PIEOTree}(t) \times \textbf{Pkt} \times \textbf{Data} \times \textbf{Path}(t) \to \textbf{PIEOTree}(t)$ by

$$\frac{\text{push}(p, \text{pkt}, d, r) = p'}{\text{push}(\text{Leaf}(p), \text{pkt}, d, r) = \text{Leaf}(p')} \qquad \frac{\text{push}(p, i, d, r) = p' \quad \text{push}(qs[i], \text{pkt}, d, pt) = q'}{\text{push}(\text{Internal}(qs, p), \text{pkt}, d, (i, r) :: pt) = \text{Internal}(qs[q'/i], p')}$$

Definition 1.5. Let $t \in \textbf{Topo}$. A *control* over t is a triple (s, q, z), where $s \in \text{St}$ is the *current state*, q is a PIEO tree of topology t, and

$$z : \text{St} \times \textbf{Pkt} \to \textbf{Data} \times \textbf{Path}(t) \times \text{St}$$

is a function called the *scheduling transaction*.

2 Well-Formedness

Definition 2.1. Fix $f \in \mathcal{F}$. Define $|\cdot|_f : \textbf{PIEOTree}(t) \to \mathbb{N}$ by

$$|\operatorname{Leaf}(p)|_f = |p|_f \qquad\qquad |\operatorname{Internal}(qs, p)|_f = \sum_{i=1}^{|qs|} |qs[i]|_f$$

We say $q \in \textbf{PIEOTree}(t)$ is *well-formed* w.r.t. f, denoted $\models_f q$, if it adheres to the following rules.

$$\frac{}{\vdash_f \operatorname{Leaf}(p)} \qquad\qquad \frac{\forall i \in [1, |qs|], \ \vdash_f qs[i] \wedge |p|_{i,f} = |qs[i]|_f}{\vdash_f \operatorname{Internal}(qs, p)}$$

We say q is f-well-formed, denoted $\vdash_f q$, if for all $f' \geq f$, $\models_{f'} q$.

Theorem 2.2. *Let* $t \in \textbf{Topo}$, $\text{pkt} \in \textbf{Pkt}$, $d \in \textbf{Data}$, $f, f' \in \mathcal{F}$, *and* $q \in \textbf{PIEOTree}(t)$ *such that* $\vdash_f q$.
  (1) *If* $pt \in \textbf{Path}(t)$, *then* $\operatorname{push}(q, \text{pkt}, d, pt)$ *is well-defined and* $\vdash_f \operatorname{push}(q, \text{pkt}, d, pt)$.
  (2) *If* $|q|_{f'} > 0$ *and* $f' \geq f$, *then* $\operatorname{pop}(q, f')$ *is well-defined and* $\vdash_{f'} q'$, *where* $\operatorname{pop}(q, f') = (\text{pkt}, q')$.

Proof. TBD $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

3 Projection

Definition 3.1. For $f \in \mathcal{F}$, define $\operatorname{proj}_f : \textbf{PIEOTree}(t) \to \textbf{PIFOTree}(t)$ by

$$\frac{p' = \operatorname{proj}(p, f)}{\operatorname{proj}_f(\operatorname{Leaf}(p)) = \operatorname{Leaf}(p')} \qquad \frac{p' = \operatorname{proj}(p, f) \qquad \forall i \in [1, |qs|], \ qs'[i] = \operatorname{proj}_f(qs[i])}{\operatorname{proj}_f(\operatorname{Internal}(qs, p)) = \operatorname{Internal}(qs', p')}$$

Lemma 3.2. *For* $q, q' \in \textbf{PIEOTree}(t)$,

$$\forall f \in \mathcal{F}, \ \operatorname{proj}_f(q) = \operatorname{proj}_f(q') \implies q = q'$$

Proof. Suppose $\operatorname{proj}_f(q) = \operatorname{proj}_f(q')$ for all $f \in \mathcal{F}$. We'll proceed by induction on t to show $q = q'$.
 (Leaf) For $t = *$, let $q = \operatorname{Leaf}(p)$ and $q' = \operatorname{Leaf}(p')$. Since

$$\operatorname{proj}_f(q) = \operatorname{Leaf}(\operatorname{proj}(p, f)) = \operatorname{Leaf}(\operatorname{proj}(p', f)) = \operatorname{proj}_f(q')$$

we know $\operatorname{proj}(p, f) = \operatorname{proj}(p', f)$ for all $f \in \mathcal{F}$. By Definition 1.1, $p = p'$ and hence $q = q'$.
 (Node) For $t = \operatorname{Node}(ts)$ and $n = |ts|$, let $q = \operatorname{Internal}(qs, p)$ and $q' = \operatorname{Internal}(qs', p')$. Notice

$$\operatorname{proj}(p, f) = \operatorname{proj}(p', f)$$
$$\operatorname{proj}_f(qs[i]) = \operatorname{proj}_f(qs'[i]) \qquad\qquad\qquad\qquad\qquad (i = 1, \dots, n)$$

for all $f \in \mathcal{F}$. Hence, $p = p'$ via Definition 1.1 and $qs = qs'$ by the inductive hypothesis, i.e. $q = q'$.
$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Lemma 3.3. *For* $q \in \textbf{PIEOTree}(t)$ *and* $f \in \mathcal{F}$, $\operatorname{pop}(q, f)$ *is undefined if and only if* $\operatorname{pop}(\operatorname{proj}_f(q))$ *is undefined.*

Proof. We'll do induction on t.
 (Leaf) For $t = *$, let $q = \operatorname{Leaf}(p)$ and $\operatorname{proj}_f(q) = \operatorname{Leaf}(p')$. By Equation (1) in Definition 1.1,

$$\operatorname{pop}(q, f) \text{ is undefined} \iff \operatorname{pop}(p, f) \text{ is undefined}$$
$$\iff \operatorname{pop}(p') \text{ is undefined}$$
$$\iff \operatorname{pop}(\operatorname{proj}_f(q)) \text{ is undefined}$$

 (Node) For $t = \operatorname{Node}(ts)$, let $q = \operatorname{Internal}(qs, p)$ and $\operatorname{proj}_f(q) = \operatorname{Internal}(qs', p')$. As before,

$$\operatorname{pop}(p, f) \text{ is undefined} \iff \operatorname{pop}(p') \text{ is undefined}$$

by Equation (1) in Definition 1.1 and

$$\operatorname{pop}(qs[i], f) \text{ is undefined} \iff \operatorname{pop}(qs'[i]) \text{ is undefined} \ \forall i \in [1, |ts|]$$

by the inductive hypothesis. Hence, using Equation ([2](#)) in Definition [1.1](#),

$$\text{pop}(q, f) \text{ is undefined} \iff \text{pop}(p, f) \text{ is undefined} \lor (\text{pop}(p, f) = (i, \_) \land \text{pop}(qs[i], f) \text{ is undefined})$$
$$\iff \text{pop}(p') \text{ is undefined} \lor (\text{pop}(p') = (i, \_) \land \text{pop}(qs'[i]) \text{ is undefined})$$
$$\iff \text{pop}(\text{proj}_f(q)) \text{ is undefined}$$

□

Lemma 3.4. *For $q \in \textbf{PIEOTree}(t)$ and $f \in \mathcal{F}$,*

$$\text{pop}(q, f) = (\text{pkt}, q') \implies \text{pop}(\text{proj}_f(q)) = (\text{pkt}, \text{proj}_f(q'))$$

Proof. More induction on t!

(Leaf) For $t = *$, let

$$q = \text{Leaf}(p_1) \qquad\qquad \text{proj}_f(q) = \text{Leaf}(p_2)$$
$$q' = \text{Leaf}(p_1') \qquad\qquad \text{proj}_f(q') = \text{Leaf}(p_2')$$

By Equation ([2](#)) in Definition [1.1](#),

$$\text{pop}(q, f) = (\text{pkt}, q') \implies \text{pop}(p_1, f) = (\text{pkt}, p_1')$$
$$\implies \text{pop}(p_2) = \text{pop}(\text{proj}(p_1, f)) = (\text{pkt}, \text{proj}(p_1', f)) = (\text{pkt}, p_2')$$
$$\implies \text{pop}(\text{proj}_f(q)) = (\text{pkt}, \text{proj}_f(q'))$$

(Node) For $t = \text{Node}(ts)$, Let

$$q = \text{Internal}(qs_1, p_1) \qquad\qquad \text{proj}_f(q) = \text{Internal}(qs_2, p_2)$$
$$q' = \text{Internal}(qs_1', p_1') \qquad\qquad \text{proj}_f(q') = \text{Internal}(qs_2', p_2')$$

Using Equation ([2](#)) in Definition [1.1](#) again and the inductive hypothesis,

$$\text{pop}(q, f) = (\text{pkt}, q') \implies \text{pop}(p_1, f) = (i, p_1') \land \text{pop}(qs_1[i], f) = (\text{pkt}, qs_1'[i])$$
$$\implies \text{pop}(\text{proj}(p_1, f)) = (i, \text{proj}(p_1', f)) \land \text{pop}(\text{proj}_f(qs_1[i])) = (\text{pkt}, \text{proj}_f(qs_1'[i]))$$
$$\implies \text{pop}(p_2) = (i, p_2') \land \text{pop}(qs_2[i]) = (\text{pkt}, qs_2'[i])$$
$$\implies \text{pop}(\text{proj}_f(q)) = (\text{pkt}, \text{proj}_f(q'))$$

□

Lemma 3.5. *For $q \in \textbf{PIEOTree}(t)$, $\text{pkt} \in \textbf{Pkt}$, $d \in \textbf{Data}$, $pt \in \textbf{Path}(t)$, and $f \in \mathcal{F}$,*

$$\text{proj}_f(\text{push}(q, \text{pkt}, d, pt)) = \begin{cases} \text{push}(\text{proj}_f(q), \text{pkt}, pt) & f(d) \text{ holds true} \\ \text{proj}_f(q) & \text{otherwise} \end{cases}$$

Proof. Even more induction on t!

(Leaf) For $t = *$, let $q = \text{Leaf}(p)$ and $pt = r$. By Equation ([3](#)) in Definition [1.1](#).

$$\text{proj}_f(\text{push}(q, \text{pkt}, d, pt)) = \text{Leaf}(\text{proj}(\text{push}(p, \text{pkt}, d, r), f))$$
$$= \begin{cases} \text{Leaf}(\text{push}(\text{proj}(p, f), \text{pkt}, r)) & f(d) \text{ holds true} \\ \text{Leaf}(\text{proj}(p, f)) & \text{otherwise} \end{cases}$$
$$= \begin{cases} \text{push}(\text{proj}_f(q), \text{pkt}, pt) & f(d) \text{ holds true} \\ \text{proj}_f(q) & \text{otherwise} \end{cases}$$

(Node) For $t = \mathsf{Node}(ts)$, let $pt = (i, r) :: pt'$ and

$$q = \mathsf{Internal}(qs, p)$$
$$\mathsf{proj}_f(q) = \mathsf{Internal}(qs', p')$$
$$\mathsf{push}(\mathsf{proj}_f(q), \mathsf{pkt}, pt) = \mathsf{Internal}(qs'', p'')$$
$$\mathsf{proj}_f(\mathsf{push}(q, \mathsf{pkt}, d, pt)) = \mathsf{Internal}(qs''', p''')$$

By Equation (3) in Definition 1.1 and the inductive hypothesis,

$$p''' = \begin{cases} \mathsf{push}(\mathsf{proj}(p, f), i, r) & f(d) \text{ holds true} \\ \mathsf{proj}(p, f) & \text{otherwise} \end{cases}$$

$$= \begin{cases} \mathsf{push}(p', i, r) & f(d) \text{ holds true} \\ p' & \text{otherwise} \end{cases} = \begin{cases} p'' & f(d) \text{ holds true} \\ p' & \text{otherwise} \end{cases}$$

$$qs'''[i] = \begin{cases} \mathsf{push}(\mathsf{proj}_f(qs[i]), \mathsf{pkt}, pt') & f(d) \text{ holds true} \\ \mathsf{proj}_f(qs[i]) & \text{otherwise} \end{cases}$$

$$= \begin{cases} \mathsf{push}(qs'[i], \mathsf{pkt}, pt') & f(d) \text{ holds true} \\ qs'[i] & \text{otherwise} \end{cases} = \begin{cases} qs''[i] & f(d) \text{ holds true} \\ qs'[i] & \text{otherwise} \end{cases}$$

By inspection of Definition 1.4 and Definition 3.1,

$$qs'[j] = qs''[j] = qs'''[j] = \mathsf{proj}_f(qs[j])$$

for all $j \in [1, |ts|]$ such that $j \neq i$. Hence,

$$qs''' = \begin{cases} qs'' & f(d) \text{ holds true} \\ qs' & \text{otherwise} \end{cases}$$

Putting everything together,

$$\mathsf{proj}_f(\mathsf{push}(q, \mathsf{pkt}, d, pt)) = \mathsf{Internal}(qs''', p''')$$

$$= \begin{cases} \mathsf{Internal}(qs'', p'') & f(d) \text{ holds true} \\ \mathsf{Internal}(qs', p') & \text{otherwise} \end{cases}$$

$$= \begin{cases} \mathsf{push}(\mathsf{proj}_f(q), \mathsf{pkt}, pt) & f(d) \text{ holds true} \\ \mathsf{proj}_f(q) & \text{otherwise} \end{cases}$$

$\square$

4 Embedding & Simulation

Definition 4.1. Let $t_1, t_2 \in \mathbf{Topo}$. We call a relation $R \subseteq \mathbf{PIEOTree}(t_1) \times \mathbf{PIEOTree}(t_2)$ a *simulation* if, for all $\mathsf{pkt} \in \mathbf{Pkt}$, $f \in \mathcal{F}$, and $q_1 \, R \, q_2$,

(1) If $\mathsf{pop}(q_1, f)$ is undefined, then so is $\mathsf{pop}(q_2, f)$
(2) If $\mathsf{pop}(q_1, f) = (\mathsf{pkt}, q_1')$, then $\mathsf{pop}(q_2, f) = (\mathsf{pkt}, q_2')$ such that $q_1' \, R \, q_2'$.
(3) For all $pt_1 \in \mathbf{Path}(t_1)$ and $d \in \mathbf{Data}$, there exists $pt_2 \in \mathbf{Path}(t_2)$ such that

$$\mathsf{push}(q_1, \mathsf{pkt}, d, pt_1) \, R \, \mathsf{push}(q_2, \mathsf{pkt}, d, pt_2)$$

If such a simulation exists, we say that q_1 is *simulated* by q_2, and we write $q_1 \preccurlyeq q_2$.

Remark 4.2. For all further discussion, we assume our embeddings are injective.

Definition 4.3. For $t_1, t_2 \in \mathbf{Topo}$, let f be an embedding from t_1 to t_2. We lift f to a map $\overline{f}$ from $\mathbf{PIEOTree}(t_1)$ to $\mathbf{PIEOTree}(t_2)$ inductively.

- For $t_1 = *$, define $\overline{f}(q) = q$. This is well-defined by [MLF$^+$23, Lemma 5.2].

- For $t_1 = \text{Node}(ts_1)$, $n = |ts_1|$, $q = \text{Internal}(qs, p)$, construct $\overline{f}_\alpha(q) \in \textbf{PIEOTree}(t_2/\alpha)$ for each prefix α of $f(i)$ for some $i \in [1, n]$. Inductively, we'll build up from $f(i)$'s to ϵ and set $\overline{f}(q) = \overline{f}_\epsilon(q)$.
  - Let $\alpha = f(i)$ for some $i \in [1, n]$. We'll set $\overline{f}_\alpha(q) = \overline{f}_i(qs[i])$, where f_i embeds t_1/i into $t_2/f(i)$ as per [MLF+23, Lemma 5.2]. This well-defined by the injectivity of f.
  - Let α point to a transient node, say with m children. For $1 \leq j \leq m$ such that $\alpha \cdot j$ is not a prefix of some $f(i)$, define $\overline{f}(q)_{\alpha \cdot j}$ to be the PIEO tree with empty PIEOs on all leaves and internal nodes. With this and recursion, we know $\overline{f}(q)_{\alpha \cdot j} \in \textbf{PIEOTree}(t_2/(\alpha \cdot j))$ for all $j \in [1, m]$. We create a new PIEO p_α as follows:
    (1) Start with p_α empty
    (2) For each i in p such that $\alpha \cdot j$ is a prefix of $f(i)$, push j into p_α with i's data and rank
    Finally, for all $j \in [1, m]$, set $qs_\alpha[j] = \overline{f}(q)_{\alpha \cdot j}$ and $\overline{f}(q)_\alpha = \text{Internal}(qs_\alpha, p_\alpha)$.

Theorem 4.4. *The following diagram commutes*

$$
\begin{array}{ccc}
\textbf{PIEOTree}(t_1) & \xrightarrow{\ \overline{f}\ } & \textbf{PIEOTree}(t_2) \\
\downarrow {\scriptstyle \text{proj}_g} & & \downarrow {\scriptstyle \text{proj}_g} \\
\textbf{PIFOTree}(t_1) & \xrightarrow{\ \widehat{f}\ } & \textbf{PIFOTree}(t_2)
\end{array}
$$

In other words, for $q \in \textbf{PIEOTree}(t_1)$ and $g \in \mathcal{F}$, $\text{proj}_g(\overline{f}(q)) = \widehat{f}(\text{proj}_g(q))$.

Proof. We'll proceed by induction on t_1. Suppose $t_1 = *$ and $q = \text{Leaf}(p)$. By [MLF+23, Lemma 5.3], $t_2 = *$ as well. By Definition 4.3 and [MLF+23, Definition 5.4], both $\overline{f}$ and $\widehat{f}$ are the identity. Hence,

$$\text{proj}_g(\overline{f}(q)) = \text{proj}_g(q) = \widehat{f}(\text{proj}_g(q))$$

Suppose $t_1 = \text{Node}(ts)$ and $q = \text{Internal}(qs, p)$. For any prefix α of $f(i)$ for $i \in [1, |ts|]$, we'll show

$$\text{proj}_g(\overline{f}(q)_\alpha) = \widehat{f}(\text{proj}_g(q))_\alpha \qquad\qquad (*)$$

by inverse induction on α. Instantiating Equation $(*)$ with $\alpha = \epsilon$ yields the desired result.

- For $\alpha = f(i)$, Equation $(*)$ holds by the outer inductive hypothesis because

$$\text{proj}_g(\overline{f}(q)_\alpha) = \text{proj}_g(\overline{f}_i(q)) = \widehat{f}_i(\text{proj}_g(q)) = \widehat{f}(\text{proj}_g(q))_\alpha$$

- Suppose α is some strict prefix of $f(i)$, pointing to a node with m children. Let

$$\text{proj}_g(\overline{f}(q)) = \text{Internal}(qs', p') \qquad \text{and} \qquad \widehat{f}(\text{proj}_g(q)) = \text{Internal}(qs'', p'')$$

There's two parts to showing Equation $(*)$, namely $qs' = qs''$ and $p' = p''$.
  - For all $j \in [1, m]$,

    $$\text{proj}_g(\overline{f}(q)_{\alpha \cdot j}) = \widehat{f}(\text{proj}_g(q))_{\alpha \cdot j}$$

    For j such that $\alpha \cdot j$ is a prefix of some $f(i)$, this follows from the inner inductive hypothesis. For all other j, notice the LHS and RHS are both PIFO trees of topology t_2, with empty PIFOs on all leaves and internal nodes. Hence, $qs'[j] = qs''[j]$ for all $j \in [1, m]$, i.e. $qs' = qs''$.
  - By inspection, it's clear following the construction for p_α from Definition 4.3 and then computing the projection $\text{proj}(p_\alpha, g)$ yields the same result as following the recipe for p_α from [MLF+23, Definition 5.4] on the projection $\text{proj}_g(q)$: that is, exactly when we filter out elements not satisfying g does not matter. Hence, $p' = p''$.

$\square$

Lemma 4.5. *Let $t_1, t_2 \in \textbf{Topo}$ and f be an embedding of t_1 inside t_2. For $g \in \mathcal{F}$,*

$$\text{pop}(q, g) \text{ is undefined} \implies \text{pop}(\overline{f}(q), g) \text{ is undefined}$$

Proof. Suppose $\text{pop}(q, g)$ is undefined. Applying both Lemma 3.3 and [MLF+23, Lemma 5.6], notice $\text{pop}(\widehat{f}(\text{proj}_g(q)))$ is undefined. By Theorem 4.4, $\widehat{f}(\text{proj}_g(q)) = \text{proj}_g(\overline{f}(q))$. Hence, $\text{pop}(\text{proj}_g(\overline{f}(q)))$ is undefined. Applying Lemma 3.3 once more, $\text{pop}(\overline{f}(q), g)$ is undefined. $\square$

Lemma 4.6. *Let $t_1, t_2 \in$ **Topo** and f be an embedding of t_1 inside t_2. For $g \in \mathcal{F}$,*

$$\mathrm{pop}(q, g) = (\mathrm{pkt}, q') \implies \mathrm{pop}(\overline{f}(q), g) = (\mathrm{pkt}, \overline{f}(q'))$$

Almost a clone of the proof [MLF$^+$23, Lemma 5.7].

Proof. We'll proceed by induction on t_1. Suppose $t_1 = *$. By [MLF$^+$23, Lemma 5.3], $t_2 = *$ as well. By Definition 4.3, $\overline{f}$ is the identity. Hence,

$$\mathrm{pop}(q, g) = (\mathrm{pkt}, q') \implies \mathrm{pop}(\overline{f}(q), g) = \mathrm{pop}(q, g) = (\mathrm{pkt}, q') = (\mathrm{pkt}, \overline{f}(q'))$$

Suppose $t_1 = \mathrm{Node}(ts)$. Let

$$q = \mathrm{Internal}(qs, p) \qquad\qquad q' = \mathrm{Internal}(qs', p') \qquad\qquad \mathrm{pop}(p, g) = (j, p')$$

For any prefix α of some $f(i)$ (where $i \in [1, |ts|]$), we'll show

$$\mathrm{pop}(\overline{f}(q)_\alpha, g) = (\mathrm{pkt}, \overline{f}(q')_\alpha) \qquad\qquad \text{if } \alpha \text{ is a prefix of } f(j) \qquad (\dagger)$$
$$\overline{f}(q)_\alpha = \overline{f}(q')_\alpha \qquad\qquad \text{otherwise}$$

by inverse induction on α. Instantiating Equation ($\dagger$) with $\alpha = \epsilon$ yields the desired result.

- Suppose $\alpha = f(i)$. If α is a prefix of $f(j)$, $i = j$ by injectivity and [MLF$^+$23, Definition 5.2, Equation (3)]. Recall $\mathrm{pop}(qs[j], g) = (\mathrm{pkt}, qs'[j])$. Hence, by the outer inductive hypothesis,

$$\mathrm{pop}(\overline{f}(q)_\alpha, g) = \mathrm{pop}(\overline{f}_j(qs[j]), g) = (\mathrm{pkt}, \overline{f}_j(qs'[j])) = (\mathrm{pkt}, \overline{f}(q')_\alpha)$$

- Once more, suppose $\alpha = f(i)$. If α is not a prefix of $f(j)$, then $i \neq j$. Since $qs[i] = qs'[i]$,

$$\overline{f}(q)_\alpha = \overline{f}_i(qs[i]) = \overline{f}_i(qs'[i]) = \overline{f}(q')_\alpha$$

- Suppose α is some strict prefix of $f(j)$, pointing to a node with m children. Let

$$\overline{f}(q)_\alpha = \mathrm{Internal}(qs_\alpha, p_\alpha) \qquad\qquad \overline{f}(q')_\alpha = \mathrm{Internal}(qs'_\alpha, p'_\alpha)$$

There exists $k \in [1, m]$ such that $\alpha \cdot k$ is a prefix of $f(j)$. By the inner inductive hypothesis,

$$qs_\alpha[i] = \overline{f}(q)_{\alpha \cdot i} = \overline{f}(q')_{\alpha \cdot i} = qs_\alpha[i] \text{ for } i \in [1, m] \text{ with } i \neq k \qquad (!)$$
$$\mathrm{pop}(qs_\alpha[k], g) = \mathrm{pop}(\overline{f}(q)_{\alpha \cdot k}, g) = (\mathrm{pkt}, \overline{f}(q')_{\alpha \cdot k}) = (\mathrm{pkt}, qs'[k])$$

Via the construction in Definition 4.3 and since $\mathrm{pop}(p, g) = p'$, $\mathrm{pop}(p_\alpha, g) = (k, p'_\alpha)$. Putting everything together, $\mathrm{pop}(\overline{f}(q)_\alpha, g) = (\mathrm{pkt}, \overline{f}(q')_\alpha)$, as desired.

- Suppose α is some strict prefix of some $f(i)$ but not $f(j)$, pointing to a node with m children. Let

$$\overline{f}(q)_\alpha = \mathrm{Internal}(qs_\alpha, p_\alpha) \qquad\qquad \overline{f}(q')_\alpha = \mathrm{Internal}(qs'_\alpha, p'_\alpha)$$

Since p and p' agree on all indices i such that $f(i)$ is a child of α, $p_\alpha = p'_\alpha$. For $i \in [1, m]$, since $\alpha \cdot i$ is not a prefix of $f(j)$, the inner inductive hypothesis yields

$$qs_\alpha[i] = \overline{f}(q)_{\alpha \cdot i} = \overline{f}(q')_{\alpha \cdot i} = qs'_\alpha[i] \qquad (!!)$$

Putting everything together, $\overline{f}(q)_\alpha = \overline{f}(q')_\alpha$, as desired.

NOTE: even when $\alpha \cdot i$ is not a prefix of any $f(i)$, Equation (!) and Equation (!!) hold! Both $\overline{f}(q)_{\alpha \cdot i}$ and $\overline{f}(q')_{\alpha \cdot i}$ would be PIEO trees with empty PIEOs on all leaf and internal nodes. $\qquad \square$

Lemma 4.7. *Let $t_1, t_2 \in$ **Topo** and f be an embedding of t_1 inside t_2. For $\mathrm{pkt} \in$ **Pkt**, $d \in$ **Data**, and $pt \in$ **Path**(t_1),*

$$\overline{f}(\mathrm{push}(q, \mathrm{pkt}, d, pt)) = \mathrm{push}(\overline{f}(q), \mathrm{pkt}, d, \widetilde{f}(pt))$$

Proof. Let $q_2 = \overline{f}(q_1)$. For $g \in \mathcal{F}$ such that $g(d)$ holds true,

(by Theorem 4.4) $\qquad \operatorname{proj}_g(\overline{f}(\operatorname{push}(q_1, \operatorname{pkt}, d, pt))) = \widehat{f}(\operatorname{proj}_g(\operatorname{push}(q_1, \operatorname{pkt}, d, pt)))$

(by Lemma 3.5) $\qquad\qquad\qquad\qquad\qquad\qquad = \widehat{f}(\operatorname{push}(\operatorname{proj}_g(q_1), \operatorname{pkt}, pt))$

(by [MLF$^+$23, Lemma 5.9]) $\qquad\qquad\qquad\qquad = \operatorname{push}(\widehat{f}(\operatorname{proj}_g(q_1)), \operatorname{pkt}, \widetilde{f}(pt))$

(by Theorem 4.4) $\qquad\qquad\qquad\qquad\qquad\quad = \operatorname{push}(\operatorname{proj}_g(q_2), \operatorname{pkt}, \widetilde{f}(pt))$

(by Lemma 3.5) $\qquad\qquad\qquad\qquad\qquad\quad = \operatorname{proj}_g(\operatorname{push}(q_2, \operatorname{pkt}, d, \widetilde{f}(pt)))$

For $g \in \mathcal{F}$ such that $g(d)$ does not hold true,

(by Theorem 4.4) $\qquad \operatorname{proj}_g(\overline{f}(\operatorname{push}(q_1, \operatorname{pkt}, d, pt))) = \widehat{f}(\operatorname{proj}_g(\operatorname{push}(q_1, \operatorname{pkt}, d, pt)))$

(by Lemma 3.5) $\qquad\qquad\qquad\qquad\qquad\qquad = \widehat{f}(\operatorname{proj}_g(q_1))$

(by Theorem 4.4) $\qquad\qquad\qquad\qquad\qquad\quad = \operatorname{proj}_g(q_2)$

(by Lemma 3.5) $\qquad\qquad\qquad\qquad\qquad\quad = \operatorname{proj}_g(\operatorname{push}(q_2, \operatorname{pkt}, d, \widetilde{f}(pt)))$

Overall, $\operatorname{proj}_g(\overline{f}(\operatorname{push}(q_1, \operatorname{pkt}, d, pt))) = \operatorname{proj}_g(\operatorname{push}(q_2, \operatorname{pkt}, d, \widetilde{f}(pt)))$ for all $g \in \mathcal{F}$. Hence,

$$\overline{f}(\operatorname{push}(q_1, \operatorname{pkt}, d, pt)) = \operatorname{push}(q_2, \operatorname{pkt}, d, \widetilde{f}(pt))$$

by Lemma 3.2, as desired. $\qquad\qquad\square$

Theorem 4.8. *Let $t_1, t_2 \in$ **Topo**. If f embeds t_1 into t_2, then*

$$R = \{(q, \overline{f}(q)) \mid q \in \textbf{PIEOTree}(t_1)\}$$

is a simulation.

Proof. By Lemma 4.5, Lemma 4.6, and Lemma 4.7, the conditions from Definition 4.1 hold. $\qquad\square$

References

[MLF+23] Anshuman Mohan, Yunhe Liu, Nate Foster, Tobias Kappé, and Dexter Kozen. Formal abstractions for packet scheduling, 2023.