Synchronization Graphs

R-KFC-A

1 Introduction

In this we examine parsing streams of tagged vertices into a canonical *synchronization graph*.

2 Background

Definition 1 (Dependency Relation)**.** A dependency relation $D \subseteq \Sigma \times \Sigma$ is a *symmetric* and *reflexive* relation on Σ.

Likewise an independence relation of D can be defined as the relative complement $I = (\Sigma \times \Sigma) \setminus D$.

Dependency relations are a general way of talking about *equivalence* relations between two streams of data $S_1, S_2 \in \Sigma^\star$, where we say $S_1 \equiv_D S_2$ if S_2 can be reached from S_1 (and vice versa) by appling permutations based on the independence relation I.

Definition 2 (Tree Dependence Relation)**.** A tree dependence relation $T \subseteq \Sigma \times \Sigma$ is a dependence relation such that (Σ, T) induces a graph with vertices Σ and edges T that forms a tree with a distinguished root $\sigma_\top$. Define the predecessor function with respect to this rooting, with $\mathbf{pred}(\sigma_\top) = \sigma_\top$; recursively define the depth function $\mathbf{depth}(\sigma_\top) = 0$ and $\mathbf{depth}(\sigma) = 1 + \mathbf{depth}(\mathbf{pred}(\sigma))$.

We define a *synchronization graph*, which is intended to model data streams that (1) are equipped with a dependence relationship and (2) have "synchronizing" (also: visibly pushdown / parallel / end-marker'd) behavior.

Definition 3 (Synchronization Graph)**.** A synchronization graph G is a directed acyclic graph with a unique *source* (top) vertex $\vee G$ and a unique *sink* (bot) vertex $\wedge G$. Recursively define G as follows:

i. (Base Case): A single vertex is a syncrhonization graph.

ii. (Sequential Concatenation): If G_1 and G_2 are synchronization graphs, then $G = G_1 \cdot u \cdot G_2$ is also a synchronization graph for a new vertex u, where

$$(\wedge G_1, u), (u, \vee G_2) \in G, \qquad \vee G = \vee G_1, \qquad \wedge G = \wedge G_2$$

iii. (Parallel Union): If G_1 and G_2 are synchronization graphs, then $G = u[G_1||G_2]v$ is also a synchronization graph for new vertices u, v where

$$(u, \vee G_1), (u, \vee G_2), (\wedge G_1, v), (\wedge G_2, v) \in G, \qquad \vee G = u, \qquad \wedge G = v$$

3 A Parsing Problem

In this problem setting, we are given:

i. A tree dependency relation T and its alphabet Σ

ii. A sequence of vertices $v_1, v_2, \ldots$ each labeled with an element of Σ.

The task is to generate a *canonical* synchronization graph.

4 A Basic Algorithm