Parikh’s Theorem: Direct Automaton Construction

Cressida Hamlet & Marilyn George

Indian Institute of Science

cressida.hamlet@csa.iisc.ernet.in
marilyn.george@csa.iisc.ernet.in

November 7, 2014
Parikh’s Theorem

Theorem

Parikh image of a context-free language is semilinear (or)
Every CFL has the same Parikh image as of some regular language.

Parikh’s proof of theorem

- Given a CFG $G$, the proof produces (implicitly) an automaton or regular expression whose language has the same Parikh image as $G$.
- Constructed automata is of size $O(n^n)$, where $n$ - number of variables in the CNF of $G$.

Enhanced automaton construction

- Given a CFG $G$, construct an automaton explicitly.
- Instead of $O(n^n)$, only $O(4^n)$ states
Symbols and notations

If $G = (V, T, P, S)$, $\alpha, \beta \in (V \cup T)^*$

- degree of $G$, $m = -1 + \max\{|\gamma_V| : (A \rightarrow \gamma) \in P\}$
- $\Pi_V(\alpha)$ - Parikh image of $\alpha$ projected on $V$
- $\Pi_T(\alpha)$ - Parikh image of $\alpha$ projected on $T$
- $\alpha_V$ - $\alpha$ projected on $V$
- $\alpha_T$ - $\alpha$ projected on $T$
- transition $t(\alpha \Rightarrow \beta) = (\Pi_V(\alpha), \gamma_T, \Pi_V(\beta))$

$\alpha \Rightarrow \beta$ is a step if there exist

- a production $A \rightarrow \gamma$ and
- $\alpha_1, \alpha_2$ such that $\alpha = \alpha_1 A \alpha_2$ and $\beta = \alpha_1 \gamma \alpha_2$
Let $G = (V, T, P, S)$ and $n = |V|$

The *k-Parikh automaton* of $G$ is the NFA $M^k_G = (Q, T^*, \delta, q_0, \{q_f\})$ where

1. $Q = \{(x_1, ..., x_n) \in \mathbb{N}^n | \sum_{i=1}^{n} x_i \leq k\}$
2. $\delta = \{t(\alpha \Rightarrow \beta) | \alpha \Rightarrow \beta \text{ is a step and } \Pi_V(\alpha), \Pi_V(\beta) \in Q\}$
3. $q_0 = \Pi_V(S)$
4. $q_f = \Pi_V(w \in T^*) = (0, ..., 0)$

We can see that $M^k_G$ has exactly $\binom{n+k}{n}$ states.
Example

3-Parikh automaton of the grammar

\[ A_1 \rightarrow A_1 A_2 | a \]
\[ A_2 \rightarrow bA_2 aA_2 | cA_1 \]

eg transition:
\[(0, 2) \xrightarrow{c} (1, 1)\]
\[A_2 A_2 \Rightarrow cA_1 A_2\]
Theorem

If $G$ is a context-free grammar with $n$ variables and degree $m$, then $L(G)$ and $L(M^{nm+1}_G)$ have the same Parikh image.

- For the above grammar $G$, $n = 2$ and $m = 1$, which means $L(G) = \prod L(M^3_G)$.
- For a grammar in CNF, $m \leq 1$ which means $L(G) = \prod L(M^{n+1}_G)$
  states required $= \binom{2n+1}{n} \leq 2^{2n+1} \in O(4^n)$
Proof of $L(M_G^k) \subseteq \Pi L(G)$

Claim

If $k \geq 1$, let $q_0 \xrightarrow{\sigma} q$ be a run of $M_G^k$ on the word $\sigma \in T^*$, there exists a step sequence $S \Rightarrow^* \alpha$ satisfying

- $\Pi_V(\alpha) = q$
- $\Pi_T(\alpha) = \Pi_T(\sigma)$

If $\sigma \in L(M_G^k)$, then there is a run $q_0 \xrightarrow{\sigma} q_f$

By claim there exists a step sequence $S \Rightarrow^* \alpha$ satisfying

$\Pi_V(\alpha) = (0, \ldots, 0)$ and $\Pi_T(\alpha) = \Pi_T(\gamma)$

So $\alpha \in T^*$ and hence $\alpha \in L(G)$.

Since $\Pi_T(\alpha) = \Pi_T(\gamma)$, we have $\alpha =_\Pi \gamma$. 
Claim’s proof (by induction on length of path)

If $\ell = 0$ then $\sigma = \epsilon$, so $\alpha = S$, and $\Pi_V(S) = q_0$ and $\Pi_T(S) = \Pi_T(\epsilon)$.

For $\ell > 0$, $\sigma = \sigma'\gamma$ and $q_0 \xrightarrow{\sigma'} q' \xrightarrow{\gamma} q$ then according to I.H. $S \Rightarrow^* \alpha'$ where $\Pi_V(\alpha') = q'$ and $\Pi_T(\alpha') = \Pi_T(\sigma')$.

Since $q' \xrightarrow{\gamma} q$ is a transition there exists a production $A \rightarrow \gamma'$ where $\gamma'/T = \gamma$ and a step $\alpha_1 A \alpha_2 \Rightarrow \alpha_1 \gamma' \alpha_2$ s.t. $\Pi_V(\alpha_1 A \alpha_2) = q'$ and $\Pi_V(\alpha_1 \gamma' \alpha_2) = q$. 
Since $\Pi_V(\alpha') = q$, $\alpha' = \alpha_1'A\alpha_2'$
Say $\alpha = \alpha_1'\gamma'\alpha_2'$ so that $S \Rightarrow^* \alpha' \Rightarrow \alpha$ and

$$\Pi_V(\alpha) = \Pi_V(\alpha_1'\gamma'\alpha_2')$$
$$= \Pi_V(\alpha_1'\gamma'\alpha_2') - \Pi_V(A) + \Pi_V(\gamma')$$
$$= \Pi_V(\alpha') - \Pi_V(A) + \Pi_V(\gamma')$$
$$= \Pi_V(\alpha_1A\alpha_2') - \Pi_V(A) + \Pi_V(\gamma')$$
$$= \Pi_V(\alpha_1'\gamma'\alpha_2') = q$$

$$\Pi_T(\alpha) = \Pi_T(\alpha_1'\gamma'\alpha_2')$$
$$= \Pi_T(\alpha_1'\gamma'\alpha_2') + \Pi_T(\gamma')$$
$$= \Pi_T(\alpha') + \Pi_T(\gamma')$$
$$= \Pi_T(\alpha') + \Pi_T(\gamma)$$
$$= \Pi_T(\sigma') + \Pi_T(\gamma)$$
$$= \Pi_T(\sigma)$$
Proof idea of $L(G) \subseteq_\Pi L(M_G^{nm+1})$

**Definition**

A derivation $S \Rightarrow \alpha_0 \Rightarrow ... \Rightarrow \alpha_\ell$ has index $k$ if for all $i \in \{0, ..., \ell\}$, the word $(\alpha_i)/V$ has at most length $k$. The set of words derivable through derivations of index $k$ is denoted by $L_k(G)$.

Clearly $L_1(G) \subseteq L_2(G) \subseteq L_3(G)\ldots$ and $L(G) = \bigcup_{k \geq 1} L_k(G)$.

We can prove $L(G) \subseteq_\Pi L(M_G^{nm+1})$ by proving

- $L(G) \subseteq_\Pi L_{nm+1}(G)$ - will prove using
  - $Y(T) \subseteq_\Pi \bigcup_{i=0}^n Y(T^{(i)})$
  - $Y(T^{(k)}) \subseteq L_{km+1}(G)$

- $L_k(G) \subseteq_\Pi L(M_G^k)$

Cressida Hamlet & Marilyn George

Parikh's Theorem: Direct Automaton Construction
Yield of a parse tree $t$ is denoted by $Y(t)$
Set of yields of a set $\mathcal{T}$ of trees by $Y(\mathcal{T})$
A child of a $t$ is a subtree of $t$ whose root is a child of the root of $t$
A child is proper if its root is not a leaf.

The *dimension* $d(t)$ of a parse tree is defined as follows:
If $t$ has no proper children then $d(t) = 0$.
Otherwise let $t_1, t_2, \ldots t_r$ be the proper children of $t$ sorted s.t.
$d(t_1) \geq d(t_2) \geq \cdots \geq d(t_r)$. Then

$$d(t) = \begin{cases} 
    d(t_1) & \text{if } r=1 \text{ or } d(t_1) > d(t_2) \\
    d(t_1) + 1 & \text{if } d(t_1) = d(t_2)
\end{cases}$$

The set of parse trees with dimension $k$ is denoted by $\mathcal{T}^{(k)}$
The height of a tree, $h(t) > d(t)$.
Collapse Lemma

**Lemma**

\[ L(G) \subseteq_\Pi L_{nm+1}(G) \]

For the proof of the Collapse Lemma:

- Every word in \( L(G) \) is the yield of a parse tree - \( L(G) = Y(T) \)
- To prove: \( Y(T) \subseteq_\Pi L_{nm+1}(G) \)
- In Lemma 2.1, we show \( Y(T) \subseteq_\Pi \bigcup_{i=0}^{n} Y(T(i)) \)
- Then we prove \( \bigcup_{i=0}^{n} Y(T(i)) \subseteq L_{nm+1}(G) \)

Note, in Lemma 2.2 we prove the stronger result that parse trees of dimension \( k \geq 0 \) have derivations of index \( km + 1 \), i.e. for all \( k \geq 0 \)

\[ Y(T^{(k)}) \subseteq L_{km+1}(G) \]
Lemma 2.1: Proof - Preliminaries

- In this proof, we write \( t = t_1.t_2 \) to denote that
  - \( t_1 \) is a parse tree except exactly one leaf \( \ell \) is labelled by a variable, say \( A \), instead of a terminal
  - \( t_2 \) is a parse tree with root \( A \)
  - \( t \) is obtained from \( t_1 \) and \( t_2 \) by replacing leaf \( \ell \) of \( t_1 \) by \( t_2 \)
Lemma 2.1: Proof - Preliminaries

Lemma

\[ Y(\mathcal{T}) \subseteq \Pi \bigcup_{i=0}^{n} Y(\mathcal{T}^{(i)}) \]

- Trees \( t \) and \( t' \) are \( \Omega \)-equivalent if
  - They have the same number of nodes
  - The sets of variables in \( t \) and \( t' \) coincide
  - \( Y(t) =_{\Pi} Y(t') \)

- A tree is compact if \( d(t) \leq K(t) \), where \( K(t) \) denotes the number of variables in \( t \)

- Since \( K(t) \leq n \) for every \( t \), it suffices to show every tree is \( \Omega \)-equivalent to a compact tree to complete the proof of Lemma 2.1
Lemma 2.1: Proof of $\mathcal{Y}(\mathcal{T}) \subseteq \pi \bigcup_{i=0}^{n} \mathcal{Y}(\mathcal{T}^{(i)})$

We now define a recursive compactification procedure $\text{Compact}(t)$ that transforms a tree $t$ into an $\Omega$-equivalent compact tree:

1. If $t$ is compact then return $t$ and terminate.

2. If $t$ is not compact then
   
   (i) Let $t_1, \ldots, t_r$ be the proper children of $t$, $r \geq 1$.
   
   (ii) For every $1 \leq i \leq r$ : $t_i := \text{Compact}(t_i)$.

   Let $x$ be the smallest index $1 \leq x \leq r$ s.t. $K(t_x) = \max_i K(t_i)$

   (iii) Choose an index $y \neq x$ s.t. $d(t_y) = \max_i d(t_i)$

   (iv) Choose subtrees $t_x^a, t_x^b$ of $t_x$ and subtrees $t_y^a, t_y^b, t_y^c$ of $t_y$ s.t.

   a. $t_x := t_x^a.t_x^b$ and $t_y := t_y^a.(t_y^b.t_y^c)$; and
   
   b. the roots of $t_x^b, t_y^b$ and $t_y^c$ are labelled by the same variable.

   (v) $t_x = t_x^a.(t_y^b.t_x^b)$ ; $t_y = t_y^a.t_y^c$

   (Roughly, remove $t_y^b$ from $t_y$ and insert it into $t_x$)

   (vi) Goto (1).
Lemma 2.1: Proof of $Y(\mathcal{T}) \subseteq \mathcal{P} \bigcup_{i=0}^{n} Y(\mathcal{T}^{(i)})$

We now prove that assumptions made at (i), (iii) and (iv) in the algorithm about the existence of certain subtrees hold.

(i) If $t$ is not compact, then $t$ has at least one proper child.

Proof by contrapositive - Assume that $t$ has no proper child. Then, by the definition of dimension and $K(t)$, we have $d(t) = 0 \leq K(t)$, and so $t$ is compact.
Lemma 2.1: Proof of $Y(\mathcal{T}) \subseteq \bigcap_{i=0}^{n} Y(\mathcal{T}^{(i)})$

(iii) Assume that $t$ is not compact, has at least one proper child, and all its proper children are compact. Let $x$ be the smallest index $1 \leq x \leq r$ s.t. $K(t_x) = \max_i K(t_i)$. There exists an index $y \neq x$ s.t. $d(t_y) = \max_i d(t_i)$.

Let $1 \leq y \leq r$ (where possibly $x = y$) be an index s.t. $d(t_y) = \max_i d(t_i)$. We have

\[
\begin{align*}
d(t) &\leq d(t_y) + 1 \quad \text{(by definition of dimension and of $y$)} \\
&\leq K(t_y) + 1 \quad \text{(as $t_y$ is compact)} \\
&\leq K(t_x) + 1 \quad \text{(by definition of $x$)} \\
&\leq K(t) + 1 \quad \text{(as $t_x$ is a child of $t$)} \\
&\leq d(t) \quad \text{(as $t$ is not compact)} \quad (1)
\end{align*}
\]

Hence, all the inequalities in (1) are now equalities. In particular, $d(t) = d(t_y) + 1$, and by definition of dimension and of $y$, there exists $y' \neq y$ s.t. $d(t_y) = d(t_{y'})$. Now $x \neq y$ or $x \neq y'$, and wlog we can choose $y$ s.t. $y \neq x$. 
Lemma 2.1: Proof of $Y(\mathcal{T}) \subseteq \bigcap_{n} \bigcup_{i=0}^{n} Y(\mathcal{T}^{(i)})$

(iv) Assume that $t$ is not compact, all its proper children are compact, and it has two distinct proper children $t_x, t_y$ s.t. $K(t_x) = \max_i K(t_i)$ and $d(t_y) = \max_i d(t_i)$. There exist subtrees $t_x^a, t_x^b$ of $t_x$ and subtrees $t_y^a, t_y^b, t_y^c$ of $t_y$ satisfying conditions a. and b.

- By the equalities in (1), $K(t_y) = d(t_y)$. Also, $d(t_y) < h(t_y)$.
- It follows $K(t_y) < h(t_y)$; some path of $t_y$ from root to leaf visits at least two nodes labelled with the same variable, say $A$. Now $t_y$ can be factored into $t_y^a \cdot (t_y^b \cdot t_y^c)$ s.t. roots of $t_y^b$ and $t_y^c$ are labelled by $A$.
- From (1) we also have $K(t) = K(t_x)$; every variable in $t$ appears in $t_x$, and so $t_x$ has a node labelled $A$. $t_x$ can then be factored into $t_x = t_x^a \cdot t_x^b$ with root of $t_x^b$ labelled by $A$.

This concludes the proof that $\text{Compact}(x)$ is well-defined.
Lemma 2.1: Proof of $\mathcal{Y}(\mathcal{T}) \subseteq \bigcap_{i=0}^{n} \mathcal{Y}(\mathcal{T}^{(i)})$

**Lemma**

*If Compact(t) terminates and returns a tree t’, then t and t’ are $\Omega$-equivalent.*

- Proof by induction on the number of calls to Compact during the execution of Compact(t)
- If Compact is called only once, only (1) is executed, t is compact, no step modifies t, and hence this holds true
- If Compact is called more than once, the only lines that modify t are (ii) and (v)
- Consider line (ii) - By I.H, each call to Compact($t_i$) returns a compact tree $t'_i$ that is $\Omega$-equivalent to $t_i$. Let $t_1$ and $t_2$ be the values of t before and after the execution of $t_i := \text{Compact}(t_i)$. Then $t_2$ is the result of replacing $t_i$ by $t'_i$ in $t_1$. Since $t'_i$ is $\Omega$-equivalent to $t_i$, we have $t_2$ is $\Omega$-equivalent to $t_1$
Lemma 2.1: Proof of $\mathcal{Y}(\mathcal{T}) \subseteq \cap \bigcup_{i=0}^{n} \mathcal{Y}(\mathcal{T}^{(i)})$

- Consider line (v) - Let $t_1$ and $t_2$ be the values of $t$ before and after the execution of $t_x := t_x^a.(t_y^b.t_x^b)$ followed by $t_y := t_y^a.t_y^c$

- Since the subtree $t_y^b$ added to $t_x$ is removed from $t_y$, the Parikh-image of $\mathcal{Y}(t)$, the number of nodes of $t$, and the set of variables appearing in $t$ do not change.

- Hence $t_1$ and $t_2$ are $\Omega$-equivalent.

This completes the proof of the Lemma. We now have that if the procedure terminates, it returns an $\Omega$-equivalent tree. It remains to prove that it always terminates.
Lemma 2.1: Proof of $Y(\mathcal{T}) \subseteq \cap_{n} \bigcup_{i=0}^{n} Y(\mathcal{T}^{(i)})$

**Proof of termination:** By contradiction, assume there is a tree $t$ s.t. $Compact(t)$ does not terminate. Wlog assume that $t$ has a minimal number of nodes.

- In this case all calls to (ii) terminate, and execution contains infinitely many steps that do not belong to any deeper call in the call tree, and infinitely many executions of the block (iii) to (v).
- **Claim:** In all executions of this block, the index $x$ has the same value.
  - From the previous lemma, line (ii) does not change the number of nodes or set of variables in each of $t_1...t_r$. Hence, it preserves $K(t_1), ..., K(t_r)$.
  - Line (v) adds nodes to $t_x$ and does not change or remove nodes from any other proper children of $t$.
  - Therefore, the value of $K(t_x)$ does not decrease; for every $i \neq x$, $K(t_i)$ does not increase.
  - It follows that $x$ is still the smallest index satisfying $K(t_x) = \max_i K(t_i)$. 
Lemma 2.1: Proof of $Y(T) \subseteq \Pi \bigcup_{i=0}^{n} Y(T^{(i)})$

From the previous claim, each execution strictly decreases the number of nodes of some proper child $t_y, y \neq x$, and increases those of $t_x$. This contradicts the fact that all proper children of $t$ have a finite number of nodes. The algorithm will always terminate and return an $\Omega$-equivalent tree.

Hence we have that every parse tree in the $L(G)$ can be reduced to a compact tree, with $d(t) \leq K(t) \leq n$ and Lemma 2.1 is proved.
Lemma 2.2: Proof of $Y(T^{(k)}) \subseteq L_{km+1}(G)$

Lemma

For every $k \geq 0 : Y(T^{(k)}) \subseteq L_{km+1}(G)$.

In this proof, we will use the following notation:
If $D$ is a derivation $\alpha_0 \Rightarrow ... \Rightarrow \alpha_\ell$ and $w, w' \in (V \cup T)^*$, then we define $wDw'$ to be the step sequence $w\alpha_0w' \Rightarrow ... \Rightarrow w\alpha_\ell w'$.

Let $t$ be a parse tree s.t. $d(t) = k$. We show a derivation for $Y(t)$ of index $km+1$.

- Proof by induction on number of non-leaf nodes in $t$.
- Base case: $t$ has no proper child. Then $k = 0$ and $t$ represents a derivation $S \Rightarrow Y(t)$ of index 1.
Lemma 2.2: Proof of $Y(\mathcal{T}^{(k)}) \subseteq L_{km+1}(G)$

- Induction step: Assume $t$ has $r \geq 1$ proper children $t_1, ..., t_r$ where the root of $t_i$ is assumed to be labelled by $A^{(i)}$; i.e., we assume the topmost level of $t$ is induced by a rule

  $$S \rightarrow \gamma_0 A^{(1)} \gamma_1 ... \gamma_{r-1} A^{(r)} \gamma_r \text{ for } \gamma_i \in T^*$$

  Note that $r - 1 \leq m$

- At most one child $t_i$ has dimension $k$, while others have dimension at most $k - 1$. Wlog assume $d(t_1) \leq k$ and $d(t_2), ..., d(t_r) \leq k - 1$.

- By I.H., for all $1 \leq i \leq r$ there is a derivation $D_i$ for $Y(t_i)$ s.t. $D_1$ has index $km + 1$, and $D_2, ..., D_r$ have index $(k - 1)m + 1$. 

Cressida Hamlet & Marilyn George

Parikh’s Theorem: Direct Automaton Construction
Lemma 2.2: Proof of $Y(T^{(k)}) \subseteq L_{km+1}(G)$

For each $1 \leq i \leq r$, define the step sequence

$$D'_i := \gamma_0 A^{(1)} \gamma_1 \cdots \gamma_{i-2} A^{(i-1)} \gamma_{i-1} D_i \gamma_i Y(t_{i+1}) \gamma_{i+1} \cdots \gamma_{r-1} Y(t_r) \gamma_r$$

Now, $D'_1$ has index $km + 1$, and for $2 \leq i \leq r$, the step sequence $D'_i$ has index $(i - 1) + (k - 1)m + 1 \leq km + 1$.

By concatenating the step sequences $S \Rightarrow \gamma_0 A^{(1)} \gamma_1 \cdots \gamma_{r-1} A^{(r)} \gamma_r$ and $D_r, D_{r-1}, ..., D_1$ in that order, we obtain a derivation for $Y(t)$ of index $km + 1$. 
Lemma 2.3 [Collapse Lemma]: Proof

\[ L(G) \subseteq \Pi L_{nm+1}(G) \]

\[ L(G) = Y(\mathcal{T}) \]
\[ \subseteq \Pi \bigcup_{i=0}^{n} Y(\mathcal{T}^{(i)}) \quad \text{(Lemma 2.1)} \]
\[ \subseteq L_{nm+1}(G) \quad \text{(Lemma 2.2)} \]
Lemma 2.4: Proof

For every $k \geq 1 : L_k(G) \subseteq \prod L(M^k_G)$.

We show that if $S \Rightarrow^* \alpha$ is a prefix of a derivation of index $k$ then $M^k_G$ has a run $q_0 \xrightarrow{w} \Pi_V(\alpha)$ s.t. $w \in T^*$ and $\alpha/T = \Pi w$.

- Proof by induction on length $i$ of the prefix.
- Base Case: When $i = 0$. $\alpha = S$, $q_0 = \Pi_V(S)$ and $S/T = \varepsilon$; this holds.
- Induction Step: When $i > 0$. Since $S \Rightarrow^i \alpha$, there exist: $\beta_1 A \beta_2 \in (V \cup T)^*$ and a production $A \rightarrow \gamma$ s.t. $S \Rightarrow^{i-1} \beta_1 A \beta_2 \Rightarrow \alpha$ and $\beta_1 \gamma \beta_2 = \alpha$. 
Lemma 2.4: Proof of $L_k(G) \subseteq \Pi L(M^k_G)$

- By I.H. there exists a run of $M^k_G$ s.t. $q_0 \xrightarrow{w_1} \Pi_V(\beta_1A\beta_2)$ and $(\beta_1A\beta_2)/T = \Pi w_1$.
- By definition of $M^k_G$ and the fact that $S \Rightarrow^i \alpha$ is of index $k$; there exists a transition $(\Pi_V(\beta_1A\beta_2), \gamma/T, \Pi_V(\alpha))$.
- Hence, $q_0 \xrightarrow{w_1.\gamma/T} \Pi_V(\alpha)$. Also from $(\beta_1A\beta_2)/T = \Pi w_1$ and $\alpha = \beta_1\gamma\beta_2$; $\alpha/T = \Pi w_1.\gamma/T$ and this is proved.

Finally, if $\alpha \in T^*$ so that $S \Rightarrow^* \alpha$ is a derivation, then:
$q_0 \xrightarrow{w} \Pi_V(\alpha) = (0,...,0)$ where $(0,...,0)$ is an accepting state and $\alpha = \alpha/T = \Pi w$. 

Cressida Hamlet & Marilyn George
Parikh’s Theorem: Direct Automaton Construction
Proof of Correctness: $L(G) \subseteq \Pi L(M_{G}^{nm+1})$

Theorem

$L(G) \subseteq \Pi L(M_{G}^{nm+1})$

$L(G) \subseteq \Pi L_{nm+1}(G)$ (Collapse Lemma)

$L(G) \subseteq \Pi L(M_{G}^{nm+1})$ (Lemma 2.4)
Thank You!