14. Counter Machines

Counter languages are the family $\mathfrak{C} = \mathfrak{F}(P, 1)$ where P is the free abelian group on countably many generators $\{p_1, p_2, \ldots\}$. $L \subset \Sigma^*$ is a counter language if any of the following equivalent conditions hold.

1. $L = \rho(1)$ for some rational relation $\rho : P \to \Sigma^*$;
2. $L = \rho \cap \Sigma^*$ for some rational relation $\rho \subset P \times \Sigma^*$;
3. L is accepted by a finite automaton $\mathcal{A}$ over $P \times \Sigma^*$; that is, L is the set of words w for which there are successful paths in $\mathcal{A}$ with label $(1, w)$.

Figure 1 shows a counter automaton $\mathcal{A}$. We may think of $\mathcal{A}$ as having two counters. The edge label (p_1, a) means "Read a and increment counter 1" while a label $(1, b)$ would mean "Read b and make no change in the counters," and $(p_1 p_2^{-1}, \lambda)$ would tell the machine to increment counter 1 and decrement counter 2 but read no input. $\mathcal{A}$

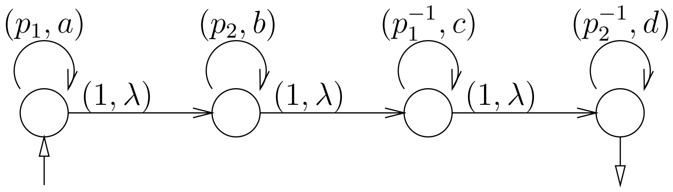

Figure 1. A counter automaton $\mathcal{A}$ accepting $\{a^i b^j c^i d^j\}$

accepts the inputs for which there is a computation beginning in the initial state with all counters empty and ending in the terminal state with all counters empty. It is clear that the labels of paths in $\mathcal{A}$ from initial to terminal vertex are

$$\{(p_1^i p_2^j p_1^{-k} p_2^{-m}, a^i b^j c^k d^m\} = \{(p_1^{i-k} p_2^{j-m}, a^i b^j c^k d^m\}$$

from which it follows that $\mathcal{A}$ accepts the desired language.

Usually counter automata can branch on empty stack, but ours cannot. If they could, they would have the same computing power as Turing machines.

Lemma 14.1. *$\mathfrak{C}$ contains all rational languages and is closed under the following operations.*

1. *Transduction and union;*
2. *Homomorphism, and inverse homomorphism;*
3. *Intersection;*
4. *Product.*

Proof. The first three assertions hold because $\mathfrak{C}$ is a family of languages. For the last two assume that L_1 and L_2 are counter languages defined using disjoint sets of generators from P. That is, there are subgroups P_1 and P_2 of P such that $P_1 \cap P_2 = 1$ and each $L_i = \rho_i \cap \Sigma^*$ where $\rho_i \subset P_i \times \Sigma^*$ is rational.

Earlier we proved that the composition of two rational relations $\sigma : M_1 \to \Sigma^*$ and $\tau : \Sigma^* \to M_2$ is rational. A slight modification of that proof shows that $\{(m_1, w, m_2) \mid w \in \rho(m_1), m_2 \in \tau(w)\}$ is a rational subset of $M_1 \times \Sigma^* \times M_2$. In our present situation this result means that $\mu = \{(q_1, w, q_2) \mid w \in \rho_1(q_1) \cap \rho_2(q_2)\}$ is a rational subset of $P_1 \times \Sigma^* \times P_2$. Hence μ is a rational subset of $P \times \Sigma^*$ and $\mu \cap \Sigma^* = L_1 \cap L_2$.

Finally $L_1 L_2 = (\rho_1 \cap \Sigma^*)(\rho_2 \cap \Sigma^*) = \rho_1 \rho_2 \cap \Sigma^*$. ☐

Theorem 14.2 (Interchange Lemma). *For each counter language L there is an integer n such that for every $w \in L$ and decomposition $w = w_0 v_1 w_1 \cdots v_n w_n$ with all $v_i \neq \lambda$, interchanging some pair of the v_i's yields a word in L.*

Proof. Let L be accepted by an automaton $\mathcal{A}$ with m vertices and take $n = 1 + m^2$. An accepted word w is part of the label of a successful path. At least two of the v_i's correspond to sub–paths beginning at the same vertex and ending at the same vertex. The word u obtained by interchanging these two sub–words is also part of the label of a successful path, and because P is abelian the first component of the label of this path is still 1. □

Exercise 14.3. *Formulate a pumping lemma for 1–counter languages, i.e. $\mathfrak{F}(Z, 0)$.*

Exercise 14.4. *Show that the word problem of G is a 1–counter language if and only if G is virtually cyclic. Hint: The word problem of a finitely generated group is context–free if and only if the group has a free subgroup of finite index.*

Problem 14.5. *Characterize groups whose word problems are counter languages.*

References

[1] S. Greibach, Remarks on blind and partially blind one–way multicounter machines, Theoretical Computer Science, **7**, 1978, 311-324.

[2] V. Mitrana and R. Stiebe, The accepting power of finite automata over groups, in *New trends in formal languages*, Lecture Notes in Computer Science, **1218**, Springer Verlag, Berlin, 1997, 39–48.