Using Communication Complexity

to Prove Languages Not Regular

Exposition by William Gasarch

1 Introduction

We use communication complexity to (1) show that certain languages are not regular, and (2) show lower bounds on the number of states for some regular languages. The techniques are not new. They are essentially in papers by Birget [1] and Galister and Shallit [2].

For more on communication complexity see [3]. We only prove as much as we need to show certain languages are not regular.

2 Communication Complexity

Def 2.1 Let $A \subseteq \{0,1\}^n \times \{0,1\}^n$. Imagine that Alice has $x \in \{0,1\}^n$ and Bob has $y \in \{0,1\}^n$. They want to determine if $(x,y) \in A$. The *Communication Complexity of* A is the minimum number of bits they need to communicate in order for them both to know if $(x,y) \in A$. We allow them unlimited computation. That is Alice and Bob can do anything they want privately (e.g, Solve SAT, Solve the HALTING problem) – our concern will only be with communication. We denote this $D(A)$ since our protocol is deterministic. We will assume that in each round the communicating player sends either a 0 or a 1.

Examples

1) For any set A, $D(A) \leq n+1$ since Alice can just GIVE Bob x, Bob determines if $(x,y) \in A$ and then sends the answer to Alice.

2) Consider the set

$MAJ = \{(x,y) \in \{0,1\}^n \times \{0,1\}^n : \#_1(xy) \geq n/2\}$

$D(MAJ) \leq \log(n)$ by having Alice send Bob $\#_1(x)$.

3) Consider the set

$EQ = \{(x,x) \in \{0,1\}^n \times \{0,1\}^n\}$

We will later show that $D(EQ) \geq n+1$. There is a randomized algorithm that takes $O(\log n)$ bits [4]; however, that is now our concern here. The slides (on the course website) show this protocol.

3 Lower Bound on $D(EQ)$

The following theorem is due to Yao [5].

Def 3.1 Let P be a protocol for $A \subseteq \{0,1\}^n \times \{0,1\}^n$. Let $(x,y) \in \{0,1\}^n \times \{0,1\}^n$. If we run P on (x,y) Alice says a_1 then Bob says b_1, etc, until it ends. The string $a_1 b_1 a_2 b_2 \cdots a_m b_m$ (it might end on a_m) is called *The Transcript of* $P(x,y)$ and denoted $T_{x,y}$.

Theorem 3.2 $D(EQ) \geq n+1$.

Proof:

Claim: If $x \neq y$ then $T_{x,x} \neq T_{y,y}$.

Proof of Claim: Assume, by way of contradiction, that $x \neq y$ and

$$T_{x,x} = T_{y,y} = a_1 b_1 a_2 b_2 \cdots a_m b_m$$

Now look at $T_{x,y}$.

Alice has x so she sends a bit a_1. Bob sees a_1 and has y so he sends b_1. Etc. Hence

$$T_{x,y} = a_1 b_1 a_2 b_2 \cdots a_m b_m$$

Since $x = x$ the transcript $T_{x,x}$ accepts (x,x). Hence the transcript $T_{x,y} = T_{x,x}$ accepts (x,y). This is a contradiction since $x \neq y$.

End of Proof of Claim

So all T_x are different. Hence there are $\geq 2^n$ transcripts. There must be at least 1 rejecting leave. So there are at least $2^n + 1$ transcripts. Hence some transcript must be of length $\geq n + 1$. ∎

4 A General Technique

The following theorem is due to Yao [5]. All the ideas for it are in the proof of Theorem 3.2.

Def 4.1 Let $A \subseteq \{0,1\}^n \times \{0,1\}^n$. A *fooling set* is a set (x_1, y_1), ..., (x_M, y_M) such that (1) for all i, $(x_i, y_i) \in A$, and (2) for all $i < j$, $(x_i, y_j) \notin A$.

Theorem 4.2 *Let $A \subseteq \{0,1\}^n \times \{0,1\}^n$. If there is a fooling set of size M then $D(f) \geq \lceil \log_2(M) \rceil$.*

5 Lower Bound on $D(MAJ)$

The following theorem is due to Yao [5].

Theorem 5.1 $D(MAJ) \geq \log n$.

Proof: We assume n is even (the n odd case is similar).

We find a fooling set of size $n/2 + 1$:

$(0^{n/2}, 1^{n/2})$

$(0^{n/2-1}1, 10^{n/2-1})$

$(0^{n/2-2}1^2, 1^20^{n/2-2})$

$(0^{n/2-3}1^3, 1^30^{n/2-3})$

$\vdots$

$(1^{n/2}, 1^{n/2})$

Clearly every ordered pair is in MAJ. If $i < j$ then $(1^i0^{n/2-i}, 1^{n/2-j}0^j) \notin MAJ$ since the number of 1's is $n/2 + i - j < n/2$, ∎

6 An Example of a prove that a Language is not Regular Using Comm Comp

Theorem 6.1 *The language $L = \{ww : w \in \{0,1\}^*\}$ is not regular. Hence $\overline{L}$ is also no regular.*

Proof: Assume, by way of contradiction, that L is regular via DFA M which has s (a constant!) states. Both Alice and Bob are given M. We now give a protocol that shows $D(EQ) = O(1)$ which contradicts $D(EQ) \geq n + 1$.

Alice gets x, Bob gets y, both of length n. Alice runs $M(x)$ and sees that it ends in state p. Alice sends p to Bob. Note that $|p| = \lg(s) + O(1) = O(1)$. Then Bob runs M on y starting at state p. If the result is a final state then he'll send Alice a 1 (for YES $x = y$), otherwise he'll send Alice a 0 (for NO $x \neq y$). This shows $D(EQ) = O(1)$ which contradicts Theorem 3.2. ∎

7 General Technique for Proving a Language is Not Regular

We generalize the technique from Section 6.

Theorem 7.1 *Let L be a language. Let $n \in \mathsf{N}$. Let*

$$L_n = \{(x, y) : |x| = |y| = n \wedge xy \in L\}.$$

Let f be an increasing function (it might increase very slowly). If for infinitely many n there exists a fooling set for L_n of size $\geq f(n)$ then L is not regular, and $\overline{L}$ is not regular.

Proof:

Since for infinitely many n there exists a fooling set for L_n of size $\geq f(n)$, for infinitely many n $D(L_n) \geq f(n)$. In particular $D(L_n)$ is NOT $O(1)$.

Assume, by way of contradiction, that L is regular via DFA M which has s (a constant!) states. Both Alice and Bob are given M. We now give a protocol that shows $D(L_n) = O(1)$.

Alice gets x, Bob gets y, both of length n. Alice runs $M(x)$ and sees that it ends in state p. Alice sends p to Bob. Note that $|p| = \lg(s) + O(1) = O(1)$. Then Bob runs M on y starting at state p. If the result is a final state then he'll send Alice a 1 (for YES $x = y$), otherwise he'll send Alice a 0 (for NO $x \neq y$). This shows $D(EQ) = O(1)$ which contradicts Theorem 3.2. ∎

8 More Examples

Theorem 8.1 *The alphabet is* $\{a, b\}$

1. $L_{sq:a,b} = \{w : \#_a(w)$ *is a square* $\}$ *is not regular.*
2. $L_{sq:a} = \{a^{n^2} : n \in \mathsf{N}\}$ *is not regular.*
3. $\overline{L_{sq:a,b}}$ *and* $\overline{L_{sq:a}}$ *are not regular.*

Proof:

1) Let $L_{sq:a,b,n^2} = \{(x, y) : |x| = |y| = n^2 \wedge xy \in L_{sq:a,b}\}$

We will only be looking at

We produce, for every n, a fooling set for $L_{sq:q,b,n^2}$ of size n.

(a^{n^2}, b^{n^2})

$(a^{n^2-1}b, ab^{n^2-1})$

$(a^{n^2-2}b^2, a^2b^{n^2-2})$

$\vdots$

$(a^{n^2-i}b^i, a^ib^{n^2-i})$

$\vdots$

$(a^{n^2-n}b^n, a^nb^{n^2-n})$

All of the ordered pairs are in $L_{sq:a,b,n^2}$. We want that if you take the LHS of any of them with the RHS of any other one the resulting pair is NOT in $L_{sq:a,b,n^2}$. Let $i < j$.

$(a^{n^2-i}b^i, a^j b^{n^2-j})$

We want that

$n^2 - i + j$ is NOT a square.

So we want

$n^2 - i + j < (n+1)^2 = n^2 + 2n + 1$

$j - i < 2n + 1$

What is the largest $j - i$ can be? The largest j can be is n The smallest i can be is 0.

$$j - i \leq n - 1 < 2n + 1.$$

Therefore $n^2 - i + j$ is NOT a square. Hence we have a fooling set.

2) Assume $L_{sq:a}$ is regular. Let M be its DFA. Attach to every state a self loop for whenever b is the input. This is a DFA for $L_{sq:a,b}$. This contradicts part 1. ∎

We leave the proof of the following theorem to the reader

Theorem 8.2 *Let $A \subseteq \mathsf{N}$ such that A is infinite, co-infinite, and there exists arbitrarily large gaps in A (formally: for all n there exists n_1 such that $n_1, n_1 + 1, \ldots, n_1 + n \notin A$).*

1. *$L_{A:a,b} = \{w : \#_a(w) \in A\}$ is not regular.*
2. *$L_{A:a} = \{a^i : i \in A\}$ is not regular.*
3. *$\overline{L_{A:a,b}}$ and $\overline{L_{A:a}}$ are not regular.*

9 Number of States

Lets look at a finite version of the set $\{ww : w \in \Sigma^*\}$.

Let

$$L_n = \{ww : |w| = n\}$$

Theorem 9.1

1. *There is a DFA for L_n with $2^{n+1}+1$ states.*

2. *Any DFA for L_n has at least 2^{n+1} states.*

Proof:

1) Let s be the start state. For every $w \in \{0,1\}^{\leq n}$ there is a state. On input w the DFA ends in state w. For every $w \in \{0,1\}^n$ there is a chain of n states so that if w is the next n letters you go to an accept state. Anything else goes to a reject state. There is only 1 reject and 1 accept state. The total number of states is $|\{0,1\}^{\leq n}| + 2 = 2^{n+1} - 1 + 2 = 2^{n+1} + 1$.

2) Let M be a DFA for L_n. Let s be the number of states. By the proof of Theorem 8.1 M can be used to show that $D(EQ) \leq \lg(s)$. By Theorem 3.2 $D(EQ) \geq n+1$. Hence

$$\lg(s) \geq n+1$$

$$s \geq 2^{n+1}$$

∎

Lower bounds on the number of states for other regular languages can be obtained in a similar manner.

10 Limitations of the Method

Let L be a langauge and $L_n = \{(x,y) : |x| = |y| = n \wedge xy \in L\}$. We have essentially proven and used the following:

If L_n has nonconstant size fooling set then L is not regular.

If L is regular then $\overline{L}$ is regular.

If L is regular then inserting a new letter anywhere you want keeps the language regular.

Can every non-regular language be proven regular with these techniques? No.

Let X be *any* subset of N. Let

$$L^X = \{w : \#_a(w) \equiv 1 \pmod 2 \vee \#_a(w)/2 \in X\}.$$

Let

$$L_n^X = \{(x, y) : xy \in L^X\}$$

For any value of X this set does not have a large fooling set. If X is not decidable then L^X is not regular. This is not quite a counterexample. We need to show that $D(L^X)$ is nonconstant which may be possible without the fooling set method.

Another counterexample, but its really stupid, is the following

$$L^Z = \{xy : |x| = |y| \wedge x \in Z\}$$

$$L_n^Z = \{(x, y) : |x| = |y| \wedge xy \in Z\}$$

For any Z, $D(L_n^Z) = 1$: Alice just tells Bob if $x \in Z$ or not. If Z is the halting set or some other non-regular set, then L^Z is not regular.

References

[1] J.-C. Birget. Intersection and union of regular languages and state complexity. *Information Processing Letters*, 28, 1992.

[2] I. Glaister and J. Shallit. A lower bound technique for the size of nondterministic finite automata. *Information Processing Letters*, 52, 1996.

[3] E. Kushilevitz and N. Nisan. *Communication Complexity.* Cambridge University Press, Cambridge, England, 1997.

[4] Mehlhorn and Schmidt. Las vegas is better than determinism in vlsi and distributed computing. In *Proceedings of the Twenty-fourth Annual ACM Symposium on the Theory of Computing,* Victoria, British Columbia, Canada, New York, 1982. ACM.

[5] A. Yao. Some complexity questions related to distributive computing. In *Proceedings of the Eleventh Annual ACM Symposium on the Theory of Computing,* Atlanta GA, 1979.