COT 5310: Theory of Automata and Formal Languages

Lecture 6

Florida State University
Department of Computer Science

Slides Credit: Dr. Michael Mascagni, CS FSU

Enumeration Theorem

Definition. We write

$$W_n = \{x \in N \mid \Phi(x, n) \downarrow\}.$$

Then we have

Theorem 4.6. A set B is r.e. if and only if there is an n for which $B = W_n$.

Proof. This is simply by the definition of $\Phi(x, n)$. □

Note that

$$W_0, W_1, W_2, \ldots$$

is an enumeration of all r.e. sets.

The Set K

Let

$$K = \{n \in N \mid n \in W_n\}.$$

Now

$$n \in K \;\Leftrightarrow\; \Phi(n, n)\downarrow \;\Leftrightarrow\; \mathrm{HALT}(n, n)$$

This, K is the set of all numbers n such that program number n eventually halts on input n.

K Is r.e. but Not Recursive

Theorem 4.7. K is r.e. but not recursive.
Proof. By the universality theorem, $\Phi(n, n)$ is partially computable, hence K is r.e.

If $\bar{K}$ were also r.e., then by the enumeration theorem,

$$\bar{K} = W_i$$

for some i. We then arrive at

$$i \in K \;\Leftrightarrow\; i \in W_i \;\Leftrightarrow\; i \in \bar{K}$$

which is a contradiction. We conclude that K is not recursive. □

r.e. Sets and Primitive Recursive Predicates

Theorem 4.8. Let B be an r.e. set. Then there is a primitive recursive predicate $R(x,t)$ such that

$$B = \{x \in N \mid (\exists t)R(x,t)\}.$$

Proof. Let $B = W_n$. Then

$$B = \{x \in N \mid (\exists t)\mathrm{STP}^{(1)}(x,n,t)\}.$$

By Theorem 3.2, $\mathrm{STP}^{(1)}$ is primitive recursive. □

A r.e. Set Is the Range of A Primitive Recursive Function

Theorem 4.9. Let S be a nonempty r.e. set. Then there is a primitive recursive function $f(u)$ such that

$$S = \{f(x) \mid x \in N\} = \{f(0), f(1), f(2), \ldots\}$$

That is, S is the range of f.

Proof. By Theorem 4.8

$$S = \{x \in N \mid (\exists t)R(x,t)\}$$

where R is primitive recursive. Let x_0 be some fixed member of S (say, the smallest), and let

$$f(u) = \begin{cases} l(u) & \text{if } R(l(u), r(u)) \\ x_0 & \text{otherwise.} \end{cases}$$

Clearly f is primitive recursive. It follows that the range of f is a subset of S. Conversely, if $x \in S$, then $R(x, t_0)$ is true for some t_0. Then $f(\langle x, t_0 \rangle) = l(\langle x, t_0 \rangle) = x$. That is, S is a subset of the range of f. We conclude $S = \{f(n) \mid x \in N\}$. □

The Range of A Partially Computable Function Is r.e.

Theorem 4.10. Let $f(x)$ be a partially computable function and let $S = \{f(x) \mid f(x)\downarrow\}$. Then S is r.e.

Proof. Let

$$g(x) = \begin{cases} 0 & \text{if } x \in S \\ \uparrow & \text{otherwise.} \end{cases}$$

Clearly $S = \{x \mid g(x)\downarrow\}$. It suffices to show that g is partially computable. Let $\mathscr{P}$ be a program that computes f and let $\#(\mathscr{P}) = p$. Then the following program computes $g(x)$:

[A] IF $\sim \text{STP}^{(1)}(Z, p, T)$ GOTO B
$V \leftarrow f(Z)$
IF $V = X$ GOTO E
[B] $Z \leftarrow Z + 1$
IF $Z \leq T$ GOTO A
$T \leftarrow T + 1$
$Z \leftarrow 0$
GOTO A

Recursively Enumerable Sets, Revisited

Theorem 4.11. Suppose that $S \neq \emptyset$. Then the following statements are all equivalent:

1. S is r.e.
2. S is the range of a primitive recursive function;
3. S is the range of a recursive function;
4. S is the range of a partially recursive function.

Proof. By Theorem 4.9, 1. implies 2. Obviously, 2. implies 3., and 3. implies 4. By Theorem 4.10, 4. implies 1. Hence all four statements are equivalent. □

The Parameter Theorem

The Parameter theorem (which has also been called the $s - m - n$ *theorem*) relates the various functions $\Phi^{(n)}(x_1, x_2, \ldots, x_n, y)$ for different values of n.

Theorem 5.1. For each $n, m > 0$, there is a primitive recursive function $S^n_m(u_1, u_2, \ldots, u_n, y)$ such that

$$\Phi^{(m+n)}(x_1, \ldots, x_m, u_1, \ldots, u_n, y) = \Phi^{(m)}(x_1, \ldots, x_m, S^n_m(u_1, \ldots, u_n, y))$$

The Parameter Theorem, Continued

$$\Phi^{(m+n)}(x_1, \ldots, x_m, u_1, \ldots, u_n, y) = \Phi^{(m)}(x_1, \ldots, x_m, S_m^n(u_1, \ldots, u_n, y))$$

Suppose the values for variables $u_1, \ldots, u_n$ are fixed and we have in mind some particular value of y. Then left hand side of the above equation is a partially computable function f of m arguments $x_1, \ldots, x_m$.

Let q be the number of a program that computes this function of m variables, we have

$$\Phi^{(m+n)}(x_1, \ldots, x_m, u_1, \ldots, u_n, y) = \Phi^{(m)}(x_1, \ldots, x_m, q)$$

The parameter theorem tells us that not only does there exist such a number q, but it can be obtained from $u_1, \ldots, u_n, y$ by using a primitive recursive function S_m^n.

The Parameter Theorem, Proof

The proof is by a mathematical induction on n. For $n = 1$, we need to show that there is a primitive recursive function $S_m^1(u, y)$ such that

$$\Phi^{(m+1)}(x_1, \ldots, x_m, u, y) = \Phi^{(m)}(x_1, \ldots, x_m, S_m^1(u, y))$$

Let $\mathscr{P}$ be the program such that $\#(\mathscr{P}) = y$. Then $S_m^1(u, y)$ can be taken to the number of the program which first gives variable X_{m+1} the value u and then proceeds to carry out $\mathscr{P}$.

The Parameter Theorem, Proof

X_{m+1} will be given the value u by the program:

$$\left.\begin{array}{l} X_{m+1} \leftarrow X_{m+1} + 1 \\ \vdots \\ X_{m+1} \leftarrow X_{m+1} + 1 \end{array}\right\} u$$

The number of the instruction $\boxed{X_{m+1} \leftarrow X_{m+1} + 1}$ is $\langle 0, \langle 1, 2m+1 \rangle \rangle = 16m + 10$. So we may take

$$S_m^1(u, y) = [(\prod_{i=1}^{u} p_i)^{16m+10} \cdot (\prod_{j=1}^{Lt(y+1)} p_{u+j}^{(y+1)_j})] \dot{-} 1$$

as the primitive recursive function.

The Parameter Theorem, Proof

To complete the proof, suppose the result is known for $n = k$. Then we have

$$\begin{aligned} & \Phi^{(m+k+1)}(x_1, \ldots, x_m, u_1, \ldots, u_k, u_{k+1}, y) \\ = \;& \Phi^{(m+k)}(x_1, \ldots, x_m, u_1, \ldots, u_k, S^1_{m+k}(u_{k+1}, y)) \\ = \;& \Phi^{(m)}(x_1, \ldots, x_m, S^k_m(u_1, \ldots, u_k, S^1_{m+k}(u_{k+1}, y))) \end{aligned}$$

using first the result for $n = 1$ and then the induction hypothesis. By now, if we define

$$S^{k+1}_m(u_1, \ldots, u_k, u_{k+1}, y) = S^k_m(u_1, \ldots, u_k, S^1_{m+k}(u_{k+1}, y))$$

we have the desired result.

The Parameter Theorem, Examples

Is there a computable function $g(u, v)$ such that

$$\Phi_u(\Phi_v(x)) = \Phi_{g(u,v)}(x)$$

for all u, v, x?
Yes! Note that

$$\Phi_u(\Phi_v(x)) = \Phi(\Phi(x, v), u)$$

is a partially computable function of x, u, v. Hence, we have

$$\Phi(\Phi(x, v), u) = \Phi^{(3)}(x, u, v, z_0)$$

for some number z_0. By the parameter theorem,

$$\Phi^{(3)}(x, u, v, z_0) = \Phi(x, S_1^2(u, v, z_0)) = \Phi_{S_1^2(u,v,z_0)}(x).$$