Relación entre los lenguajes definidos

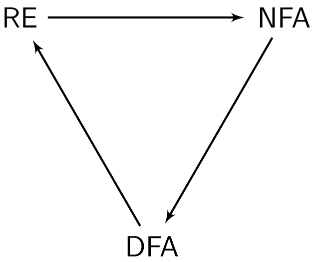
Horst H. von Brand
vonbrand@inf.utfsm.cl

Departamento de Informática
Universidad Técnica Federico Santa María

Contenido

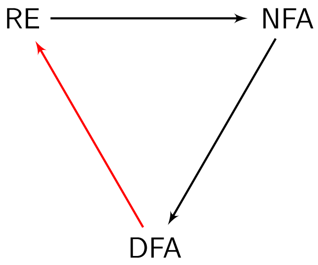
Plan de batalla

Estamos demostrando las construcciones explícitas indicadas por el diagrama:



Manos a la obra

Tercer paso:



DFA $\to$ RE

Las construcciones anteriores son de gran importancia práctica: (variantes de) expresiones regulares son representaciones cómodas de patrones de búsqueda; si algún patrón se usa muy frecuentemente vale la pena pagar el costo de crear un DFA, más eficiente de interpretar.

Cerrar el ciclo, como haremos ahora, es solo para satisfacer nuestra curiosidad intelectual.

Idea de la construcción

Lo que haremos será construir expresiones regulares para las palabras que llevan al DFA de un estado a otro, para todas las combinaciones de estados. Teniendo las anteriores, basta elegir las que llevan al DFA del estado inicial a algún estado final y unirlas.

Para obtener una construcción sistemática, ordenaremos los estados (el orden es totalmente arbitrario; claro que el resultado dependerá del orden elegido, pero todos dan expresiones equivalentes). Para simplificar, el conjunto de estados será simplemente $Q = \{1, 2, \dots, n\}$.

Definición de $R_{ij}^{(k)}$

Sea $M = (Q, \Sigma, \delta, q_0, F)$ un DFA, donde sin pérdida de generalidad $Q = \{1, 2, \ldots, n\}$. Construiremos las expresiones regulares $R_{ij}^{(k)}$ con el significado siguiente:

$$R_{ij}^{(k)} = \{\text{llevan a } M \text{ de } i \text{ a } j\text{, pasando a lo más por } k \text{ entremedio}\}$$

La restricción es sobre los estados intermedios, debemos calcular $R_{ij}^{(k)}$ para *todas* las combinaciones de estados i y j.
Las expresiones $R_{ij}^{(n)}$ no tienen restricciones, son las que buscamos para construir la expresión regular final.

Advertencia

Note que si son n estados, son n^3 expresiones $R_{ij}^{(k)}$ a construir; veremos que las expresiones se complican rápidamente conforme crece k. Es una construcción puramente de interés teórico. ***No desarrollaremos ningún ejemplo.*** No insista. Concéntrese en entender porqué funciona.

Cálculo de las expresiones

Base: Las expresiones $R_{ij}^{(0)}$ corresponden a ir directamente del estado i al j (no tenemos permiso de visitar estados entremedio).

$$R_{ij}^{(0)} = \begin{cases} \varepsilon & i = j \\ (a_1 \mid a_2 \mid \cdots \mid a_r) & \delta(i, a_s) = j \\ \varnothing & \text{caso contrario} \end{cases}$$

Cálculo de las expresiones

Inducción: Para ir de i a j pasando a lo más por el estado k entremedio podemos:

- Nunca pasar por k, representado por $R_{ij}^{(k-1)}$
- Visitar k una o más veces. Es ir de i a k ($R_{ik}^{(k-1)}$), posiblemente visitar k varias veces más ($(R_{kk}^{(k-1)})^*$), finalmente ir de k a j ($R_{kj}^{(k-1)}$)

En resumen, para $1 \leq k \leq n$:

$$R_{ij}^{(k)} = R_{ij}^{(k-1)} \mid R_{ik}^{(k-1)} (R_{kk}^{(k-1)})^* R_{kj}^{(k-1)}$$

Resumen

- Tres formas de describir lenguajes: RE, DFA, NFA. ¿Cómo se relacionan?
- Plan de batalla: demostrar un ciclo de construcciones, RE → NFA → DFA → RE
- Completamos la tarea.
- Los primeros dos pasos son de gran importancia práctica (y hay buenos algoritmos), el último interesa por razones puramente teóricas.

Preguntas pendientes:

- ¿Son estos todos los lenguajes que hay?
- ¿Cómo construir el DFA más simple (mínimo número de estados) para el lenguaje?