

Autómatas finitos no-deterministas

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

Departamento de Informática
Universidad Técnica Federico Santa María

Contenido

No-determinismo

El modelo formal

Definición formal

Lenguaje aceptado

Resumen

Idea de no-determinismo

El *no-determinismo* es un concepto muy fundamental, bastante difícil de aprehender.

En la base, nuestro uso (¡hay otras acepciones!) significa que nuestro sistema (autómata, programa, ...) no siempre tiene definido completamente el camino a seguir, tiene *elecciones* que tomar en ciertos puntos. La idea es que la computación del caso tiene éxito si hay alguna secuencia de decisiones que llevan a aceptar, falla solo si no hay ninguna forma de llegar a aceptar.

Idea de no-determinismo

Hay varias maneras de entender este tipo de modelos:

Brujería: El sistema «adivina» cuál es la opción correcta. Nunca se equivoca; de haber varias opciones que llevan a la meta elige siempre el camino más corto.

Paralelismo: Siga todos los caminos en paralelo; si el camino elegido no resulta, esa computación falla silenciosamente. La primera en terminar exitosamente decide.

Probar sistemáticamente: Se exploran todas las posibilidades, y se retiene el camino exitoso más corto (de haberlo).

Elija algoritmo: Dejamos al sistema o al usuario la tarea de resolver el problema. Estamos especificando la *solución buscada*, no *cómo se resuelve* el problema.

Idea de no-determinismo

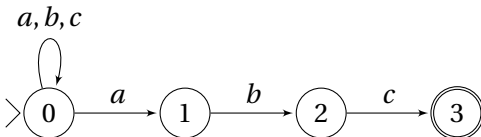
El sentido formal es el último (especificar lo que se busca, no cómo hallarlo), las otras alternativas suelen ser más intuitivas y resultan útiles en discusión informal.

Resultará importante más adelante el énfasis en «seguir el mejor camino». Sabemos que el mejor camino existe, en general no conocemos manera de hallarlo que no sea probar todas las opciones. En eso consiste el poder del no-determinismo.

Ejemplo de autómata no-determinista

La idea es similar al DFA que definimos antes, pero damos la opción de tomar caminos alternativos.

Sea el lenguaje de palabras sobre $\Sigma = \{a, b, c\}$ que terminan en abc . No tenemos cómo avanzar hasta el final, retroceder tres posiciones y revisar que coinciden. Pero podemos dejar que el autómata «adivine» que le quedan solo tres símbolos por leer:

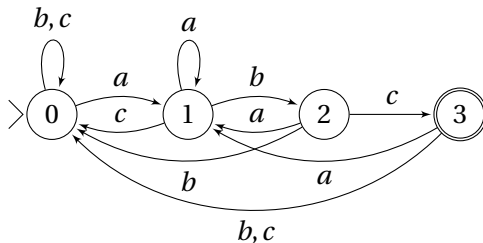


Ejemplo de autómata no-determinista

La interpretación de paralelismo es que en el estado 0 con a simultáneamente sigue en el estado 0 y otra tarea avanza al estado 1. Si ahora viene, digamos c , la segunda computación termina, solo queda la primera que vuelve al estado 0.

Autómata determinista equivalente

Determinista es recordar en los estados el prefijo de *abc* visto recién, si es *abc* y estamos al final, aceptamos.



El estado 0 es no haber visto nada, 1 significa que lo último visto es *a*, 2 es que lo último es *ab*, 3 corresponde a haber visto *abc*.

Modelo formal

Formalizamos el modelo discutido recién, lo que permitirá demostraciones más adelante.

Definición

Un *autómata finito no determinista* (abreviado NFA, por el inglés *Nondeterministic Finite Automaton*) $M = (Q, \Sigma, \delta, q_0, F)$ consta de:

Q : Conjunto finito de *estados*

Σ : *Alfabeto* de entrada

δ : *Función de transición*, $\delta: Q \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^Q$

q_0 : *Estado inicial*, $q_0 \in Q$

F : Conjunto de *estados finales*, $F \subseteq Q$

Comentarios

El único cambio con respecto al caso de autómatas finitos deterministas es la definición de la función de transición:

Transiciones espontáneas: Puede cambiar de estado leyendo ε , es decir nada. O sea, cambia de estado sin estímulo externo.

Resultado es un conjunto de estados: O sea, con la misma entrada el autómata puede ir a estados diferentes. Note también que el valor de δ puede ser el conjunto vacío, o sea, no hay salida desde ese estado.

Descripción instantánea

Definición

Sea $M = (Q, \Sigma, \delta, q_0, F)$ un NFA, donde suponemos $Q \cap \Sigma = \emptyset$. Una *descripción instantánea de M* es una palabra sobre $\Sigma \cup Q$:

$$\alpha q \beta$$

con $\alpha, \beta \in \Sigma^*$, $q \in Q$.

La idea es indicar que M leyó α , puede estar en el estado q y le queda por leer β .

Relación de transición

Definición

Sea $M = (Q, \Sigma, \delta, q_0, F)$ un NFA. La *relación de transición* de M entre descripciones instantáneas de M , anotada \vdash_M , se define por:

$$\begin{array}{ll} \alpha q x \beta \vdash_M \alpha x p \beta & \text{para todo } \alpha, \beta \in \Sigma^* \text{ y } x \in \Sigma \text{ si } p \in \delta(q, x) \\ \alpha q \beta \vdash_M \alpha p \beta & \text{para todo } \alpha, \beta \in \Sigma^* \text{ si } p \in \delta(q, \varepsilon) \end{array}$$

La idea es que M leyó α , está en el estado q y le queda por leer $x\beta$ (note que x es un símbolo o ε ahora); el primer paso es leer x desde q .

Lenguaje aceptado

Definición

Sea $M = (Q, \Sigma, \delta, q_0, F)$ un NFA, \vdash_M su relación de transición. El *lenguaje aceptado por M* es:

$$\mathcal{L}(M) = \{\alpha \in \Sigma^* : q_0 \alpha \vdash_M^* \alpha q_f \wedge q_f \in F\}$$

Note que la definición es idéntica a la para DFAs. Claro que allá había un único estado posible, acá posiblemente son muchos. Como el estado no es único, no es posible una definición como la que pasa por la función de transición extendida de un DFA.

DFA como NFA particulares

Intuitivamente, de sus descripciones gráficas, un DFA es simplemente un NFA especial: no tiene transiciones con ε ni hace uso de la licencia de tener varias transiciones desde un estado con el mismo símbolo, y puede simplemente omitir estados muertos (transiciones que no llevan a puerto).

Formalmente, dado el DFA $M = (Q, \Sigma, \delta, q_0, F)$ podemos construir el NFA $M' = (Q, \Sigma, \delta', q_0, F)$ con $\delta'(q, a) = \{\delta(q, a)\}$ para todo $q \in Q$ y $a \in \Sigma$ y $\delta'(q, \varepsilon) = \emptyset$ para todo $q \in Q$. Escépticos recalcitrantes demostrarán que las relaciones de transición \vdash_M y $\vdash_{M'}$ coinciden.

Resumen

- ▶ Definimos nuestra noción de no-determinismo. La noción de no-determinismo es extremadamente importante, familiarícese con ella.
- ▶ Definimos formalmente nuestro modelo de autómata finito no-determinista.
- ▶ Definimos el lenguaje aceptado por un NFA.
- ▶ Vimos un ejemplo de NFA y un DFA (mucho más complicado) que reconocen el mismo lenguaje.

Tenemos tres maneras de describir lenguajes: mediante expresiones regulares y ser aceptados por un DFA o un NFA. Una pregunta obvia es la relación entre ellas.