Introducción

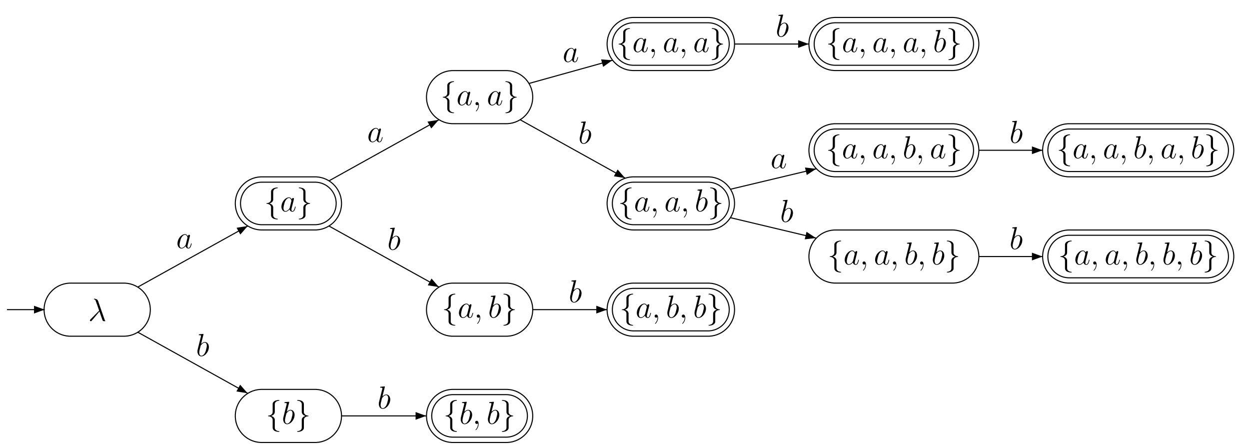
El problema de detectar en que posiciones aparece una o un conjunto de distintas palabras (usualmente denominadas patrones) en una palabra más larga x (texto) se conoce como *String Matching* o *Pattern Matching*. Este problema es de grán interés algoritmico y tiene utilidad práctica en campos como, por ejemplo, la Biología Molecular o la Genética, ya que permite el procesamiento rápido de secuencias biológicas.

Una primera aproximación (*naive*) al problema conlleva buscar cada patrón en la secuencia lo que supone un coste de $\mathcal{O}(n \cdot |p| \cdot |x|)$, donde n es el número de patrones a localizar, $|p|$ es la longitud del patrón más largo y $|x|$ es la longitud del texto.

Dado un conjunto M de palabras sobre determinado alfabeto Σ, el *árbol aceptor de prefijos* para M ($AAP(M)$) es un autómata determinista que acepta exclusivamente M. Por ejemplo, dado el conjunto:

$$M = \{\{a\}, \{b, b\}, \{a, a, a\}, \{a, a, b\}, \{a, b, b\}, \{a, a, a, b\}, \{a, a, b, a\}, \{a, a, b, a, b\}, \{a, a, b, b, b\}\}$$

el $AAP(M)$ sería el siguiente:

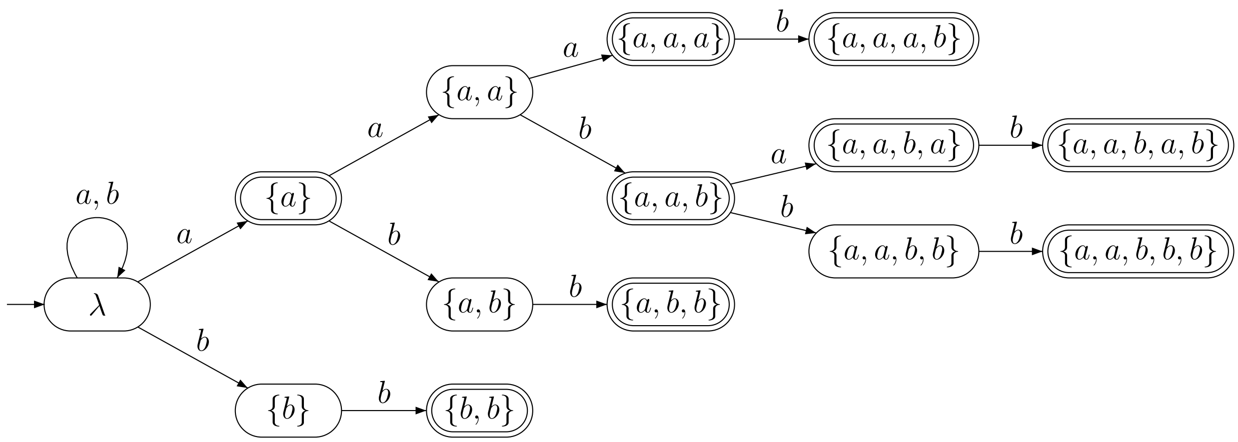


Nótese que es fácil construir el AAP de un conjunto de M de palabras si consideramos los prefijos de todas las palabras en M. Intuitívamente, para aceptar una palabra cualquiera, es necesario procesar cada símbolo de ella y en el orden correcto.

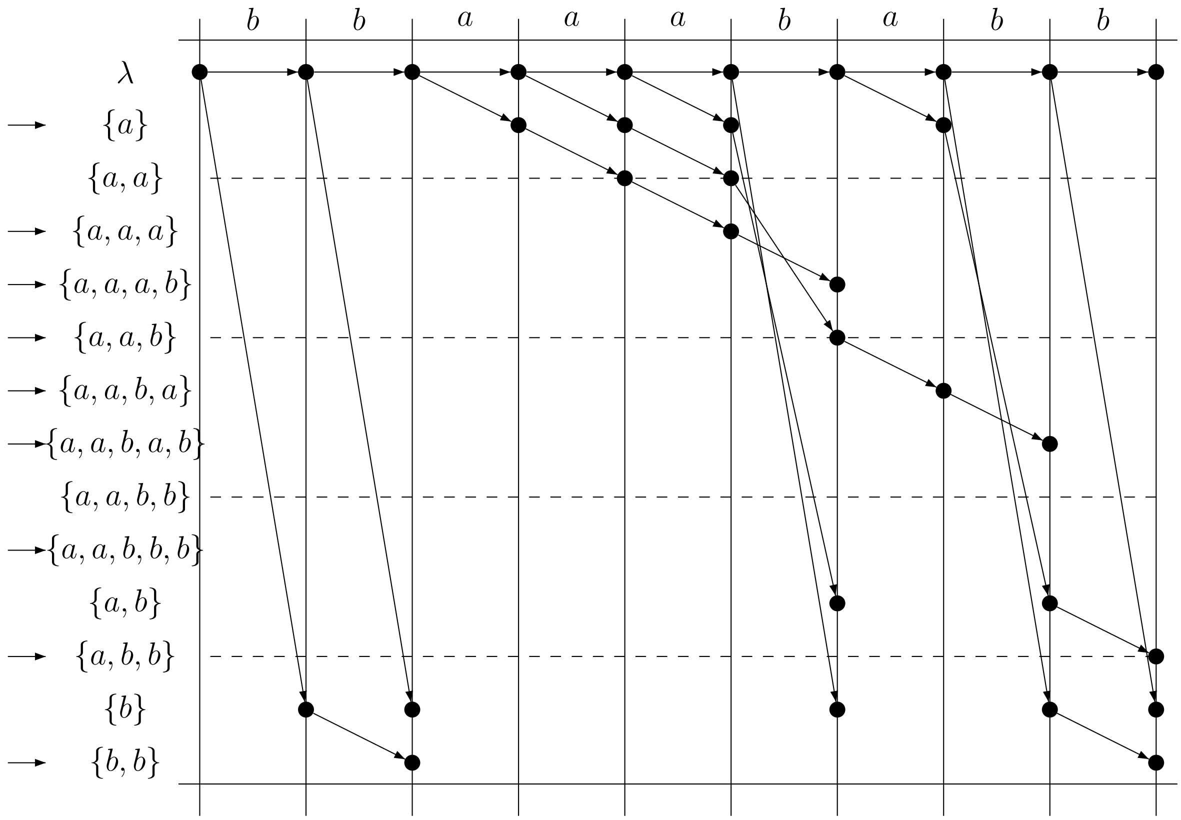
Formalmente, el $AAP(M)$ se define como el autómata $A = (Q, \Sigma, \delta, q_0, F)$ donde:

- $Q = \{x \in \Sigma^* \ : \ x \in Pref(M)\}$
- $q_0 = \lambda$
- $F = M$
- $\delta(x, a) = xa$ si $xa \in Q$, estando indefinida en caso contrario.

Este autómata puede modificarse fácilmente para obtener un NFA que acepte Σ^*M. Nótese que para ello basta añadir un bucle sobre el estado inicial con todos los símbolos del alfabeto. A continuación se muestra el NFA obtenido a partir del ejemplo anterior.



A partir de este autómata, es posible detectar todas las posiciones donde aparece una palabra de M en un texto x. Para ello basta realizar un análisis no determinista modificado ligeramente. Esta modificación consiste en detectar cada vez que se alcanza un estado final en el conjunto de estados activos. Por ejemplo, considerando el texto $x = \{b, b, a, a, a, b, a, b, b\}$, el análisis no determinista puede representarse como sigue:



En este diagrama puede verse que, después de analizar el segundo símbolo se alcanzan los estados $\{b\}$ y $\{b, b\}$. El hecho de que el estado $\{b, b\}$ sea final indica que se ha detectado un patrón del conjunto M (el patrón bb). Del mismo modo, como ejemplos: después de analizar bba y $bbaa$ se alcanza (entre otros) el estado $\{a\}$ que indica que se ha detectado el patrón a; cuando se analiza $bbaaa$ se alcanzan los estados finales $\{a\}$ y $\{a, a, a\}$ que indican que se han detectado los patrones a y aaa, y así sucesivamente.

Ejercicios

Ejercicio 1

Implementar un módulo Mathematica que, tomando un conjunto de palabras M como entrada, devuelva el árbol aceptor de prefijos de ese conjunto.

Ejercicio 2

Implementar un módulo Mathemática que, tomando un conjunto de palabras M como entrada, devuelva un AFN que acepte el lenguaje Σ^*M.

Ejercicio 3

Implementar un módulo Mathematica para, dados un conjunto de patrones M y un texto x, construya un AFN que acepte el lenguaje $\Sigma^* M$ y lo utilice para, realizando un análisis eficiente del texto x, devuelva las posiciones de x en las que aparece un patrón en M y cuál es.

Ejemplo: Dados:

$$x = \{b, a, b, a, a, b, b, a, b, b, b, a, b, b, a, b, a, a, a, a, a, b, b, a, a, b, b, a, b, a\}$$
$$M = \{\{b, b\}, \{a, b, b, b\}, \{b, b, a, b\}, \{a, a, a, a\}\}$$

el módulo debería devolver:

$\{\{6, \{b, b\}\}, \{6, \{b, b, a, b\}\}, \{9, \{b, b\}\}, \{10, \{b, b\}\}, \{10, \{b, b, a, b\}\}, \{8, \{a, b, b, b\}\},$
$\{13, \{b, b\}\}, \{13, \{b, b, a, b\}\}, \{17, \{a, a, a, a\}\}, \{18, \{a, a, a, a\}\}, \{22, \{b, b\}\},$
$\{26, \{b, b\}\}, \{26, \{b, b, a, b\}\}\}$

Nota: Para resolver el ejercicio se recomienda modificar el ejercicio de la práctica 2 que aborda el análisis de una palabra en un autómata no determinista.

Bibliografía

Maxime Crochemore, Christophe Hancart and Thierry Lecroq ALGORITHMS ON STRINGS. *Cambridge University Press*, 2007.