Contando las palabras aceptadas por un autómata

IIC3810

Una primera motivación

¿Cuál es la complejidad de este problema?

Dado un grafo G, queremos contar el número de subgrafos de G que son 3 colorables

La clase de funciones SpanP

Extendemos la noción de MTC para el caso no determinista

- Una MTC no determinista tiene un conjunto de estados finales, y una salida se considera válida si la máquina termina en un estado final

La clase de funciones SpanP

Extendemos la noción de MTC para el caso no determinista

- Una MTC no determinista tiene un conjunto de estados finales, y una salida se considera válida si la máquina termina en un estado final

Definición

Una función $f : \Sigma^* \to \mathbb{N}$ está en SpanP si y sólo si existe una MTC no determinista M con alfabeto de entrada Σ, que funciona en tiempo polinomial y tal que para cada $w \in \Sigma^*$:

$$f(w) \text{ es igual al número de salidas válidas de } M \text{ con entrada } w$$

La clase de funciones SpanP

Ejercicios

1. Sea #SUB-3-COL una función tal que, dado un grafo G, cuenta el número de subgrafos de G que son 3-coloreables. Muestre que #SUB-3-COL $\in$ SpanP

2. Demuestre que #P $\subseteq$ SpanP

$\# \cdot \mathcal{C}$: una visión unificada

$\mathcal{C}$ es una clase de complejidad para problemas de decisión

$\# \cdot \mathcal{C}$: una visión unificada

$\mathcal{C}$ es una clase de complejidad para problemas de decisión

Definición

Una función $f : \Sigma^* \to \mathbb{N}$ *está en* $\# \cdot \mathcal{C}$ *si existe una relación* $R \in \mathcal{C}$ *y un polynomio* p *tal que:*

- *Si* $(x, y) \in R$*, entonces* $|y| \leq p(|x|)$
- $f(x) = N_R(x)$ *para todo* $x \in \Sigma^*$

$\# \cdot \mathcal{C}$: una visión unificada

Claramente tenemos que $\#\mathrm{P} = \# \cdot \mathrm{P}$

$\# \cdot \mathcal{C}$: una visión unificada

Claramente tenemos que $\#\mathrm{P} = \# \cdot \mathrm{P}$

Proposición

$SpanP \;=\; \# \cdot NP$

$\# \cdot \mathcal{C}$: una visión unificada

Claramente tenemos que $\#\mathrm{P} = \# \cdot \mathrm{P}$

Proposición

$SpanP \; = \; \# \cdot NP$

Ejercicio

Demuestre la proposición

Un problema completo para SpanP

¿Qué variante del 3-CNF-SAT es completo para SpanP?

- Bajo reducciones parsimoniosas

Un problema completo para SpanP

¿Qué variante del 3-CNF-SAT es completo para SpanP?

- Bajo reducciones parsimoniosas

Dada una formula proposicional φ en 3-CNF con variables $\{x_1, \ldots, x_n\}$ y $k \in \{1, \ldots, n\}$, defina $\#$SUB-3-CNF-SAT(φ, k) como:

$$\Big|\Big\{\sigma : \{x_1, \ldots, x_k\} \to \{0,1\} \;\Big|\; \text{existe } \sigma' : \{x_{k+1}, \ldots, x_n\} \to \{0,1\} \text{ tal que } (\sigma \cup \sigma')(\varphi) = 1\Big\}\Big|$$

Un problema completo para SpanP

¿Qué variante del 3-CNF-SAT es completo para SpanP?

- Bajo reducciones parsimoniosas

Dada una formula proposicional φ en 3-CNF con variables $\{x_1, \ldots, x_n\}$ y $k \in \{1, \ldots, n\}$, defina $\#$SUB-3-CNF-SAT(φ, k) como:

$$\left| \left\{ \sigma : \{x_1, \ldots, x_k\} \to \{0,1\} \;\middle|\; \text{existe } \sigma' : \{x_{k+1}, \ldots, x_n\} \to \{0,1\} \text{ tal que } (\sigma \cup \sigma')(\varphi) = 1 \right\} \right|$$

Teorema

$\#$SUB-3-CNF-SAT es SpanP-completo bajo reducciones parsimoniosas

La clase de funciones SpanL

¿Recuerda las clases de complejidad L y NL?

- ¿Cómo se define el uso de espacio en una Máquina de Turing?

La clase de funciones SpanL

¿Recuerda las clases de complejidad L y NL?

- ¿Cómo se define el uso de espacio en una Máquina de Turing?

Extendemos la noción de MTC para el caso no determinista y con uso restringido de espacio

- Nuevamente consideramos un conjunto de estados finales, y una salida se considera válida si la máquina termina en un estado final

La clase de funciones SpanL

¿Recuerda las clases de complejidad L y NL?

- ¿Cómo se define el uso de espacio en una Máquina de Turing?

Extendemos la noción de MTC para el caso no determinista y con uso restringido de espacio

- Nuevamente consideramos un conjunto de estados finales, y una salida se considera válida si la máquina termina en un estado final

Definición

Una función $f : \Sigma^* \to \mathbb{N}$ está en SpanL si y sólo si existe una MTC no determinista M con alfabeto de entrada Σ, que funciona en espacio logarítmico y tal que para cada $w \in \Sigma^*$:

$f(w)$ es igual al número de salidas válidas de M con entrada w

La clase de funciones SpanL

Ejercicios

1. Demuestre que si $f \in \text{SpanL}$, entonces $L_f \in \text{P}$
2. Demuestre que $\text{SpanL} \subseteq \#\text{P}$
3. Demuestre que $\#\text{DNF-SAT} \in \text{SpanL}$
4. Demuestre que si $\text{SpanL} \subseteq \text{FP}$, entonces $\text{FP} = \#\text{P}$

¿Podemos utilizar la notación $\# \cdot \mathcal{C}$ para caracterizar SpanL?

¿Es cierto que $\mathrm{SpanL} = \# \cdot \mathrm{NL}$?

¿Podemos utilizar la notación $\# \cdot \mathcal{C}$ para caracterizar SpanL?

¿Es cierto que SpanL $= \# \cdot$ NL?

- Imponemos una restricción a la forma en que se procesa el segundo argumento y de una entrada (x, y)

¿Podemos utilizar la notación $\# \cdot \mathcal{C}$ para caracterizar SpanL?

¿Es cierto que $\text{SpanL} = \# \cdot \text{NL}$?

- Imponemos una restricción a la forma en que se procesa el segundo argumento y de una entrada (x, y)

Una 2-1-MT es una MT que tiene dos cintas de entrada

- La primera funciona de manera usual
- La segunda solo se puede leer una vez de izquierda a derecha

¿Podemos utilizar la notación $\# \cdot \mathcal{C}$ para caracterizar SpanL?

¿Es cierto que SpanL $= \# \cdot$ NL?

- Imponemos una restricción a la forma en que se procesa el segundo argumento y de una entrada (x, y)

Una 2-1-MT es una MT que tiene dos cintas de entrada

- La primera funciona de manera usual
- La segunda solo se puede leer una vez de izquierda a derecha

Usamos 2-1-MT cuando aceptamos relaciones en espacio logarítmico

¿Podemos utilizar la notación $\# \cdot \mathcal{C}$ para caracterizar SpanL?

Definición

Una función $f : \Sigma^* \to \mathbb{N}$ está en $\# \cdot NL$ si existe una relación R y un polynomio p tal que:

- *R es aceptada por una 2-1-MT no determinista y que funciona en espacio logarítmico*
- *Si $(x, y) \in R$, entonces $|y| \leq p(|x|)$*
- *$f(x) = N_R(x)$ para todo $x \in \Sigma^*$*

¿Podemos utilizar la notación $\# \cdot \mathcal{C}$ para caracterizar SpanL?

Definición

Una función $f : \Sigma^* \to \mathbb{N}$ está en $\# \cdot NL$ si existe una relación R y un polynomio p tal que:

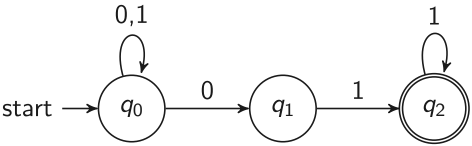
- *R es aceptada por una 2-1-MT no determinista y que funciona en espacio logarítmico*
- *Si $(x, y) \in R$, entonces $|y| \leq p(|x|)$*
- *$f(x) = N_R(x)$ para todo $x \in \Sigma^*$*

Ejercicio

Demuestre que SpanL $= \# \cdot NL$

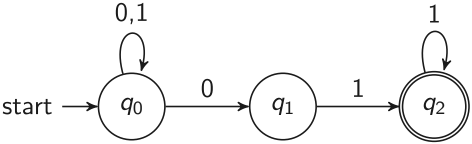
El problema #NFA

Para recordar: un automata finito no deterministico (NFA) con alfabeto $\{0, 1\}$



El problema #NFA

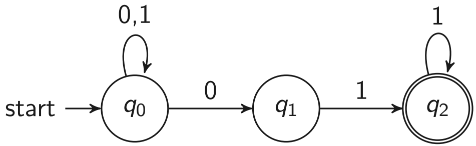
Para recordar: un automata finito no deterministico (NFA) con alfabeto $\{0, 1\}$



Queremos contar el número de palabras de un largo dado aceptados por un NFA

El problema #NFA

Para recordar: un automata finito no deterministico (NFA) con alfabeto $\{0, 1\}$

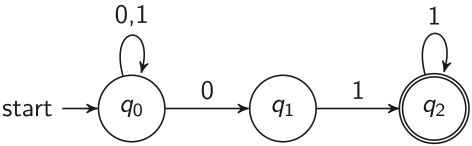


Queremos contar el número de palabras de un largo dado aceptados por un NFA

- ¿Cuántos palabras de largo 10 acepta el autómata mostrado arriba?

El problema #NFA

Para recordar: un automata finito no deterministico (NFA) con alfabeto $\{0, 1\}$



Queremos contar el número de palabras de un largo dado aceptados por un NFA

- ¿Cuántos palabras de largo 10 acepta el autómata mostrado arriba? 511

El problema #NFA

Dado un NFA $\mathcal{A}$ con alfabeto Σ y un número natural n en unario, defina $\#\text{NFA}(\mathcal{A}, n)$ como:

$$|\{w \in \Sigma^* \mid |w| = n \text{ y } w \text{ es aceptado por } \mathcal{A}\}|$$

El problema #NFA

Dado un NFA $\mathcal{A}$ con alfabeto Σ y un número natural n en unario, defina #NFA$(\mathcal{A}, n)$ como:

$$|\{w \in \Sigma^* \mid |w| = n \text{ y } w \text{ es aceptado por } \mathcal{A}\}|$$

En la entrada de #NFA:

- ¿Cómo es entregado $\mathcal{A}$?
- ¿Como es entregado n? ¿Por qué debe estar en unario?

La complejidad de #NFA

Teorema

#NFA es SpanL-completo bajo reducciones parsimoniosas

La complejidad de #NFA

Teorema

#NFA es SpanL-completo bajo reducciones parsimoniosas

Corolario

#NFA es #P-completo

La complejidad de #NFA

Ejercicios

1. Demuestre que #NFA está en SpanL

2. Demuestre que #NFA es SpanL-hard bajo reducciones parsimoniosas.

    2.1 Demuestre el corolario a partir de este resultado

Aproximando #NFA

En esta capítulo vamos a demostrar que #NFA admite un FPRAS

Aproximando #NFA

En esta capítulo vamos a demostrar que #NFA admite un FPRAS

Vamos a concluir que cada problema en SpanL admite un FPRAS

Aproximando #NFA

En esta capítulo vamos a demostrar que #NFA admite un FPRAS

Vamos a concluir que cada problema en SpanL admite un FPRAS

- Esto nos da una manera alternativa para construir un FPRAS para una función

Aproximando #NFA

En esta capítulo vamos a demostrar que #NFA admite un FPRAS

Vamos a concluir que cada problema en SpanL admite un FPRAS

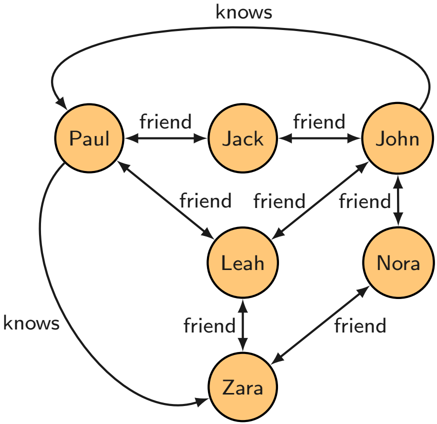
- Esto nos da una manera alternativa para construir un FPRAS para una función

Una de las motivaciones para demostrar que #NFA admite un FPRAS es tener una clase de complejidad definida por un modelo de máquina donde cada función admite un FPRAS

- Pero esta no es la única motivación

Una segunda motivación: bases de datos de grafos

G :



Una segunda motivación: bases de datos de grafos

L es un conjunto de etiquetas

Una segunda motivación: bases de datos de grafos

L es un conjunto de etiquetas

Un grafo con arcos etiquetados en L es una tupla $G = (N, A)$ donde

- N es un conjunto de nodos
- $A \subseteq N \times L \times N$ es un conjunto de arcos etiquetados en L

Una segunda motivación: bases de datos de grafos

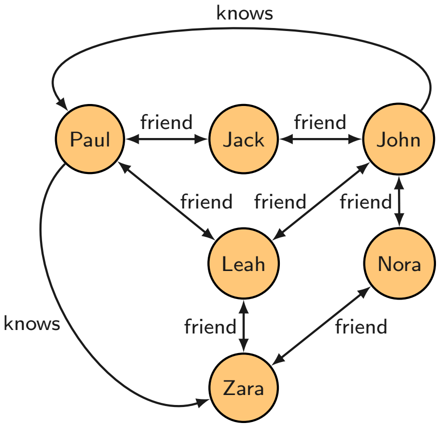
L es un conjunto de etiquetas

Un grafo con arcos etiquetados en L es una tupla $G = (N, A)$ donde

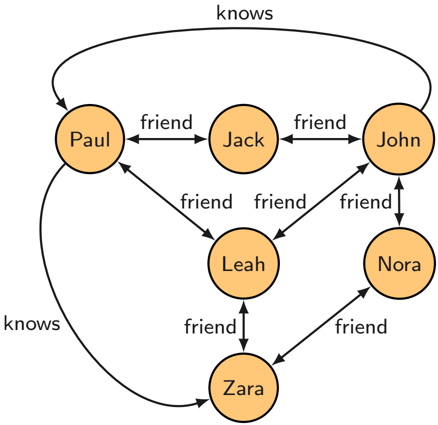
- N es un conjunto de nodos
- $A \subseteq N \times L \times N$ es un conjunto de arcos etiquetados en L

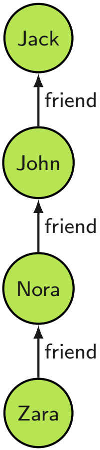
Una base de datos de grafos sobre L es un grafo con arcos etiquetados en L

Una consulta sobre G: $(\text{friend} + \text{knows})^*$

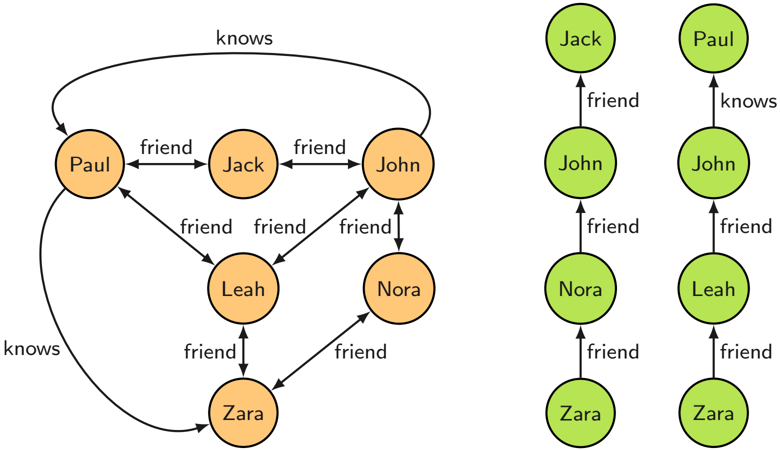


Una consulta sobre G: (friend + knows)*





Una consulta sobre G: (friend + knows)*



Una consulta sobre G: $(\text{friend} + \text{knows})^*$

Sea L un conjunto de etiquetas y $G = (N, A)$ una base de datos de grafos sobre L

Una consulta sobre G: $(\text{friend} + \text{knows})^*$

Sea L un conjunto de etiquetas y $G = (N, A)$ una base de datos de grafos sobre L

La secuencia $\pi = u_0, \ell_1, u_1, \ell_2, \ldots, \ell_n, u_n$ es un camino en G si

- $n \geq 0$
- $(u_i, \ell_{i+1}, u_{i+1}) \in A$ para cada $i \in \{0, \ldots, n-1\}$

El camino π va desde u_0 a u_n y su largo es n

Una consulta sobre G: $(\text{friend} + \text{knows})^*$

Una consulta sobre G es una expressión regular r sobre L

- Esta consulta es una regular path query (RPQ)

Una consulta sobre G: $(\text{friend} + \text{knows})^*$

Una consulta sobre G es una expressión regular r sobre L

- Esta consulta es una regular path query (RPQ)

Un camino $\pi = u_0, \ell_1, u_1, \ell_2, \ldots, \ell_n, u_n$ satisface la expresión regular r si y sólo si $\ell_1 \cdots \ell_n$ es un palabra en el lenguaje definido por r

Una consulta sobre G: $(\text{friend} + \text{knows})^*$

Una consulta sobre G es una expressión regular r sobre L

- Esta consulta es una regular path query (RPQ)

Un camino $\pi = u_0, \ell_1, u_1, \ell_2, \ldots, \ell_n, u_n$ satisface la expresión regular r si y sólo si $\ell_1 \cdots \ell_n$ es un palabra en el lenguaje definido por r

Utilizamos la siguiente notación:

$$[\![r]\!]_G \quad = \quad \{\pi \mid \pi \text{ es un camino en } G \text{ y } \pi \text{ satisface } r\}$$

Dos problemas fundamentales en bases de datos de grafos

- REACH: Dado una base de datos G, un nodo de partida s, un nodo de llegada t y una expresión regular r, determinar si existe un camino p desde s a t tal que $p \in [\![r]\!]_G$

Dos problemas fundamentales en bases de datos de grafos

- REACH: Dado una base de datos G, un nodo de partida s, un nodo de llegada t y una expresión regular r, determinar si existe un camino p desde s a t tal que $p \in [\![r]\!]_G$

- #PATH: Dado una base de datos G, un nodo de partida s, un nodo de llegada t, una expresión regular r y un largo n, contar el número de caminos p desde s a t tales que el largo de p es n y $p \in [\![r]\!]_G$

Y esto no es sólo teoría

neo4j

SQL/PGQ

Amazon Neptune

PuppyGraph

TigerGraph

DuckDB

Millenium DB

GQL Standard

La complejidad de REACH

Teorema

$REACH \in P$

La complejidad de REACH

Teorema

$REACH \in P$

Ejercicio

Demuestre la proposición considerando primero el caso sin una expresión regular como entrada

La complejidad de #PATH

Teorema

#PATH es SpanL-completo bajo reducciones parsimoniosas

La complejidad de #PATH

Teorema

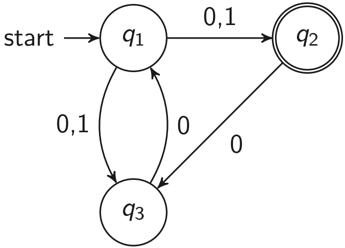
#PATH es SpanL-completo bajo reducciones parsimoniosas

Ejercicio

Demuestre que #PATH está SpanL suponiendo que la expresión regular r de entrada es dada como un NFA

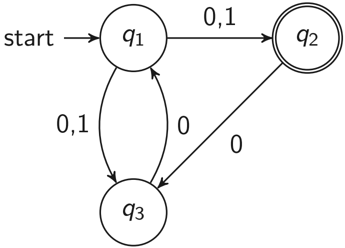
#PATH es SpanL-hard bajo reducciones parsimoniosas

Considere el siguiente NFA $\mathcal{A}$:



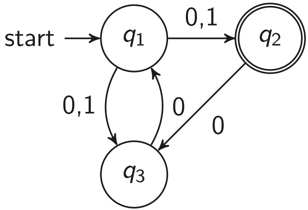
#PATH es SpanL-hard bajo reducciones parsimoniosas

Considere el siguiente NFA $\mathcal{A}$:

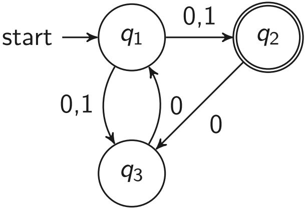


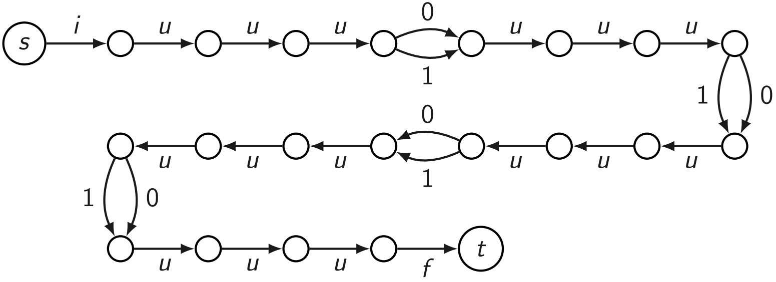
Suponga que necesitamos retornar el número de palabras de largo 4 aceptados por $\mathcal{A}$

Reducción parsimoniosa desde #NFA a #PATH



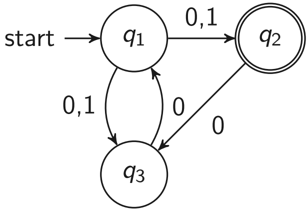
Reducción parsimoniosa desde #NFA a #PATH

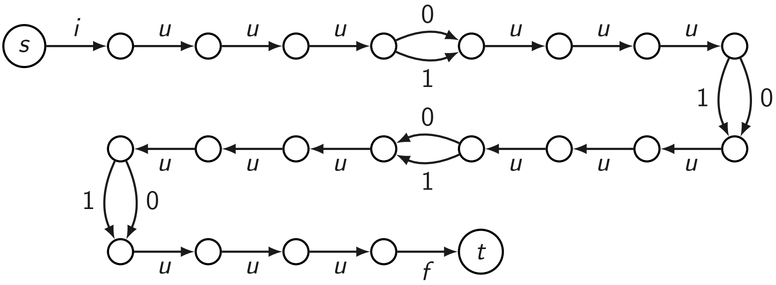




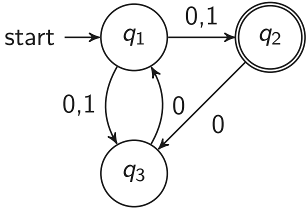
Reducción parsimoniosa desde #NFA a #PATH

q_j es el estado inicial: $\quad i/u^j$

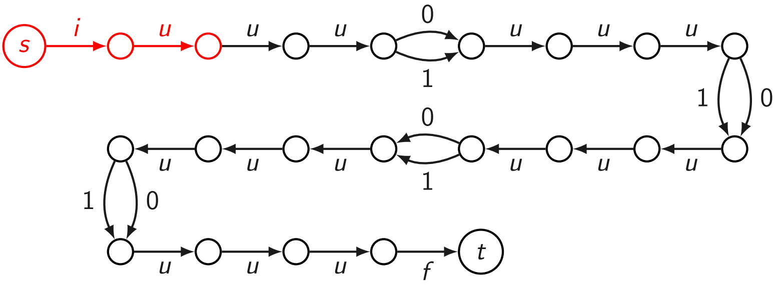




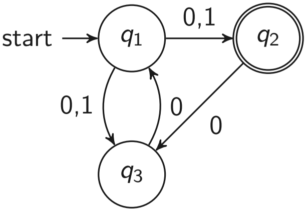
Reducción parsimoniosa desde #NFA a #PATH



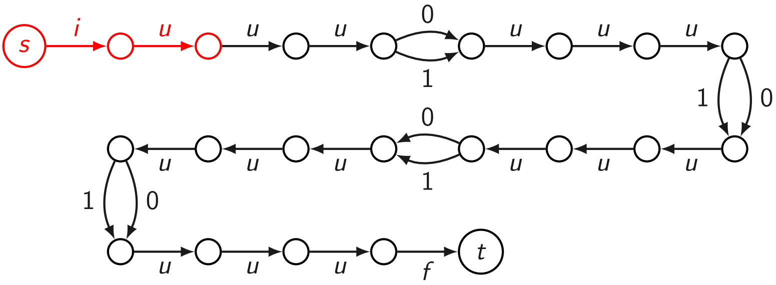
q_j es el estado inicial: $\quad i/u^j$



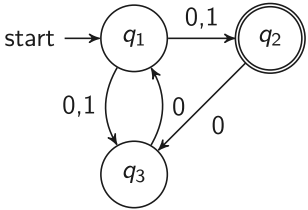
Reducción parsimoniosa desde #NFA a #PATH



q_j es el estado inicial:	i/u^j
(q_j, a, q_k) es una transición:	$u^{3-j}/a/u^k$

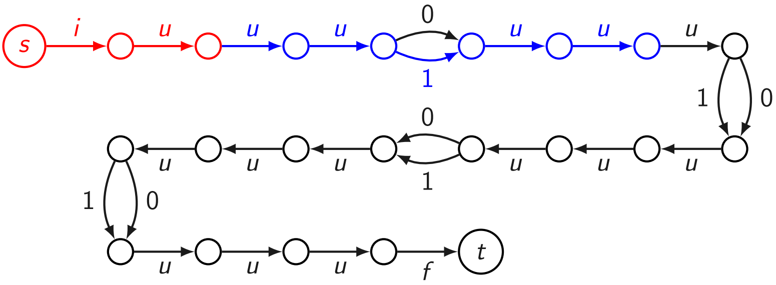


Reducción parsimoniosa desde #NFA a #PATH

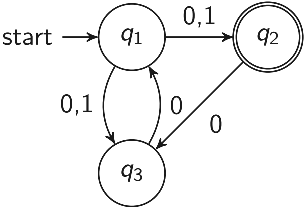


q_j es el estado inicial: i/u^j

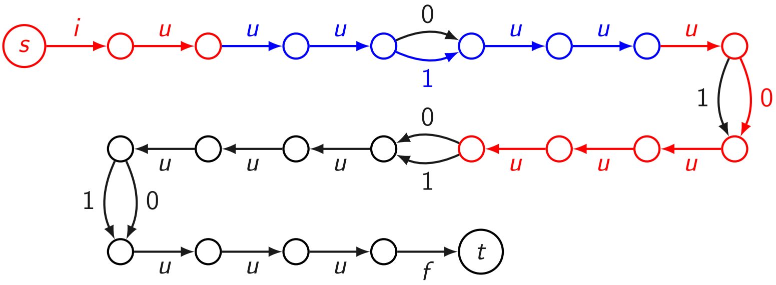
(q_j, a, q_k) es una transición: $u^{3-j}/a/u^k$



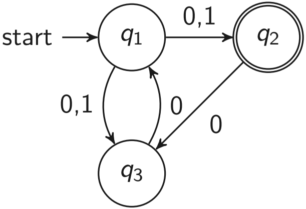
Reducción parsimoniosa desde #NFA a #PATH



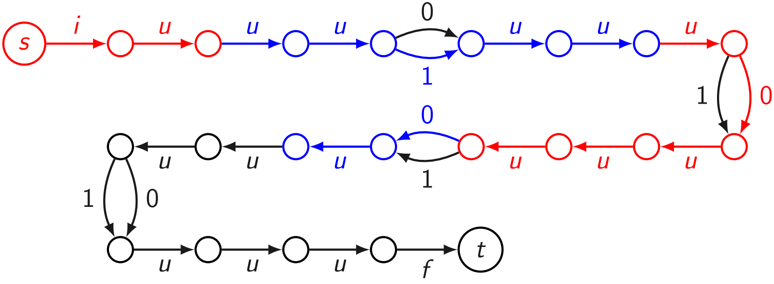
q_j es el estado inicial:	i/u^j
(q_j, a, q_k) es una transición:	$u^{3-j}/a/u^k$



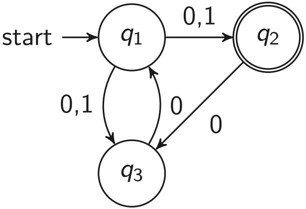
Reducción parsimoniosa desde #NFA a #PATH



q_j es el estado inicial:	i/u^j
(q_j, a, q_k) es una transición:	$u^{3-j}/a/u^k$

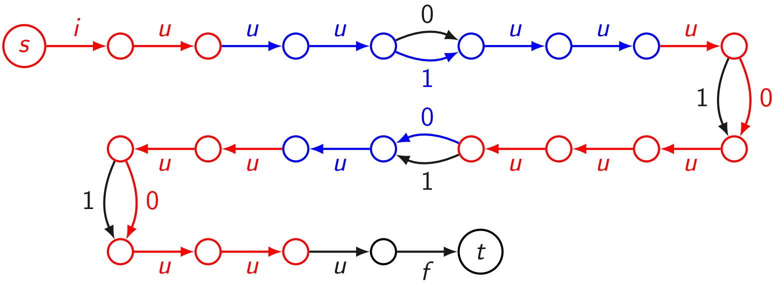


Reducción parsimoniosa desde #NFA a #PATH

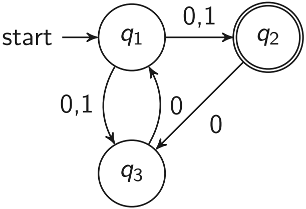


q_j es el estado inicial: $\quad i/u^j$

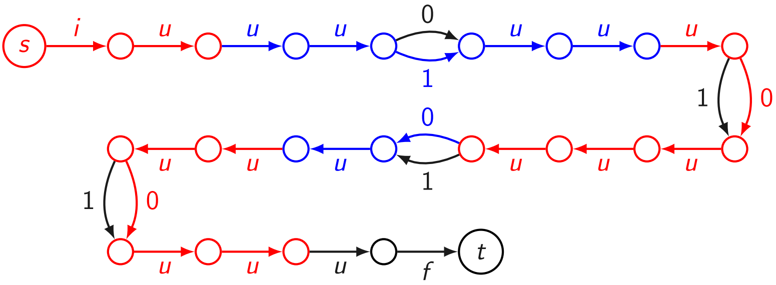
(q_j, a, q_k) es una transición: $\quad u^{3-j}/a/u^k$



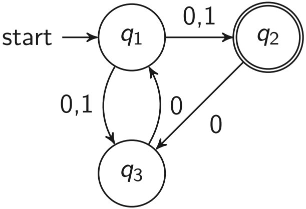
Reducción parsimoniosa desde #NFA a #PATH



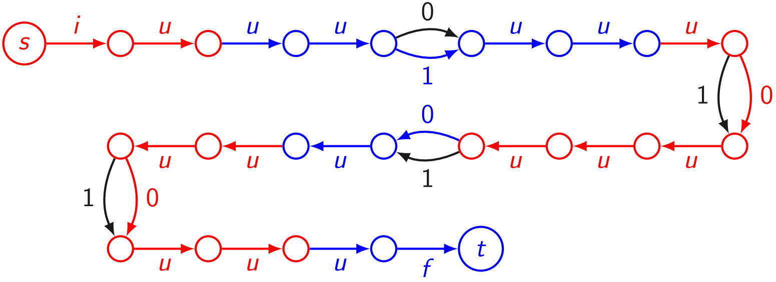
q_j es el estado inicial:	i/u^j
(q_j, a, q_k) es una transición:	$u^{3-j}/a/u^k$
q_j es el estado final:	u^{3-j}/f



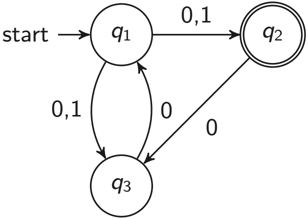
Reducción parsimoniosa desde #NFA a #PATH



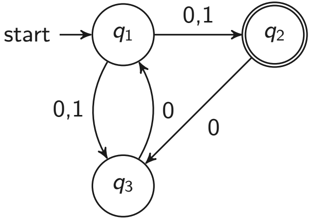
q_j es el estado inicial:	i/u^j
(q_j, a, q_k) es una transición:	$u^{3-j}/a/u^k$
q_j es el estado final:	u^{3-j}/f



Reducción parsimoniosa desde #NFA a #PATH

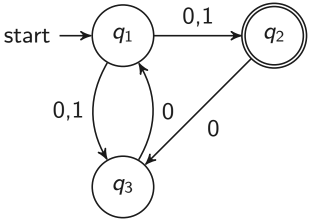


Reducción parsimoniosa desde #NFA a #PATH



Tenemos que $r = (i/u + u/u/1/u/u/ + u/0/u/u/u + 0/u + \ldots + u/f)^*$

Reducción parsimoniosa desde #NFA a #PATH



Tenemos que $r = (i/u + u/u/1/u/u/ + u/0/u/u/u + 0/u + \ldots + u/f)^*$

Número de palabras de largo 4 aceptado por $\mathcal{A}$

$=$

Número de caminos $p \in [\![r]\!]_G$ desde s hasta t de largo $21 = (5 \times 3 + 4 + 2)$

Aproximando #NFA: una segunda motivación

Como vamos a demostrar que #NFA admite un FPRAS, vamos a concluir que #PATH admite un FPRAS

Aproximando #NFA: una segunda motivación

Como vamos a demostrar que #NFA admite un FPRAS, vamos a concluir que #PATH admite un FPRAS

Tenemos un algoritmo de aproximación eficiente para un problema fundamental en bases de datos de grafos

Aproximando #NFA: una segunda motivación

Como vamos a demostrar que #NFA admite un FPRAS, vamos a concluir que #PATH admite un FPRAS

Tenemos un algoritmo de aproximación eficiente para un problema fundamental en bases de datos de grafos

- Este resultado es una segunda motivación para encontrar un FPRAS para #NFA

Nuestro objetivo es construir un FPRAS para #NFA

Sea $L(\mathcal{A})$ el lenguaje aceptado un NFA $\mathcal{A}$

- Y sea $L_n(\mathcal{A}) = \{w \in L(\mathcal{A}) \mid |w| = n\}$

Nuestro objetivo es construir un FPRAS para #NFA

Sea $L(\mathcal{A})$ el lenguaje aceptado un NFA $\mathcal{A}$

- Y sea $L_n(\mathcal{A}) = \{w \in L(\mathcal{A}) \mid |w| = n\}$

La definición de #NFA:

Entrada	:	Un NFA $\mathcal{A}$ sobre el alfabeto $\{0, 1\}$ y un largo n dado en unario
Salida	:	$\lvert L_n(\mathcal{A}) \rvert$

Nuestro objetivo es construir un FPRAS para #NFA

La entrada del FPRAS: $\mathcal{A}$, n and $\varepsilon \in (0, 1)$

Nuestro objetivo es construir un FPRAS para #NFA

La entrada del FPRAS: $\mathcal{A}$, n and $\varepsilon \in (0,1)$

La tarea es construir un número N que sea una $(1 \pm \varepsilon)$-aproximación de $|L_n(A)|$:

$$\Pr\Big((1-\varepsilon)|L_n(\mathcal{A})| \;\leq\; N \;\leq\; (1+\varepsilon)|L_n(\mathcal{A})|\Big) \;\;\geq\;\; \frac{3}{4}$$

Nuestro objetivo es construir un FPRAS para #NFA

La entrada del FPRAS: $\mathcal{A}$, n and $\varepsilon \in (0,1)$

La tarea es construir un número N que sea una $(1 \pm \varepsilon)$-aproximación de $|L_n(A)|$:

$$\Pr\Big((1-\varepsilon)|L_n(\mathcal{A})| \ \leq \ N \ \leq \ (1+\varepsilon)|L_n(\mathcal{A})|\Big) \ \geq \ \frac{3}{4}$$

Además, el número N tiene que ser calculado en tiempo $p(m, n, \frac{1}{\varepsilon})$, donde p es un polinomio y m es el número de estados de $\mathcal{A}$

Construyendo un FPRAS para #NFA

Suponga que $\mathcal{A} = (Q, \{0, 1\}, \Delta, I, F)$

- Q es un conjunto finito de estados
- $\Delta \subseteq Q \times \{0, 1\} \times Q$ es la relación de transición
- $I \subseteq Q$ es un conjunto de estados iniciales
- $F \subseteq Q$ es un conjunto de estados finales

Primer componente: desenrollando $\mathcal{A}$

Construimos un NFA $\mathcal{A}_{unroll}$ desde $\mathcal{A}$:

- Para cada estado $q \in Q$, se incluye los estados q^0, q^1, ..., q^n en $\mathcal{A}_{unroll}$
- Para cada transición $(p, a, q) \in \Delta$ e $i \in \{0, 1, \ldots, n-1\}$, se incluye la transición (p^i, a, q^{i+1}) en $\mathcal{A}_{unroll}$

Primer componente: desenrollando $\mathcal{A}$

Construimos un NFA $\mathcal{A}_{unroll}$ desde $\mathcal{A}$:

- Para cada estado $q \in Q$, se incluye los estados q^0, q^1, $\ldots$, q^n en $\mathcal{A}_{unroll}$
- Para cada transición $(p, a, q) \in \Delta$ e $i \in \{0, 1, \ldots, n-1\}$, se incluye la transición (p^i, a, q^{i+1}) en $\mathcal{A}_{unroll}$

Además, eliminamos desde $\mathcal{A}_{unroll}$ los estados no necesarios

- Cada estado q^i es parte de un camino desde un estado inicial p^0 $(p \in I)$ a un estado final r^n $(r \in F)$

Segundo componente: un sketch a ser usado en la estimación

Cada camino en $\mathcal{A}_{unroll}$ define una única palabra $w \in \{0,1\}^*$

- Llamamos a esta palabra w la etiqueta del camino

Segundo componente: un sketch a ser usado en la estimación

Cada camino en $\mathcal{A}_{unroll}$ define una única palabra $w \in \{0,1\}^*$

- Llamamos a esta palabra w la etiqueta del camino

Sea $L(q^i)$ el conjunto de palabras $w \in \{0,1\}^*$ tal que existe un camino desde un estado inicial p^0 a q^i con etiqueta w

- Nótese que $|w| = i$

Segundo componente: un sketch a ser usado en la estimación

Cada camino en $\mathcal{A}_{unroll}$ define una única palabra $w \in \{0,1\}^*$

- Llamamos a esta palabra w la etiqueta del camino

Sea $L(q^i)$ el conjunto de palabras $w \in \{0,1\}^*$ tal que existe un camino desde un estado inicial p^0 a q^i con etiqueta w

- Nótese que $|w| = i$

Además, para cada $X \subseteq Q$ defina: $L(X^i) = \bigcup_{q \in X} L(q^i)$

Segundo componente: un sketch a ser usado en la estimación

Cada camino en $\mathcal{A}_{unroll}$ define una única palabra $w \in \{0,1\}^*$

- Llamamos a esta palabra w la etiqueta del camino

Sea $L(q^i)$ el conjunto de palabras $w \in \{0,1\}^*$ tal que existe un camino desde un estado inicial p^0 a q^i con etiqueta w

- Nótese que $|w| = i$

Además, para cada $X \subseteq Q$ defina: $L(X^i) = \bigcup_{q \in X} L(q^i)$

La tarea es calcular una estimación de $|L(F^n)|$

Segundo componente: un sketch a ser usado en la estimación

En el siguiente análisis suponemos que $n \geq 2$ y $m \geq 2$

Segundo componente: un sketch a ser usado en la estimación

En el siguiente análisis suponemos que $n \geq 2$ y $m \geq 2$

- ¿Qué hacemos si $n \leq 1$ o $m \leq 1$?

Segundo componente: un sketch a ser usado en la estimación

En el siguiente análisis suponemos que $n \geq 2$ y $m \geq 2$

- ¿Qué hacemos si $n \leq 1$ o $m \leq 1$?

Sea $\kappa = \left\lceil \frac{nm}{\varepsilon} \right\rceil$

Segundo componente: un sketch a ser usado en la estimación

Para cada estado q^i mantenemos:

Segundo componente: un sketch a ser usado en la estimación

Para cada estado q^i mantenemos:

- $N(q^i)$: una $(1 \pm \kappa^{-2})^i$-aproximación de $|L(q^i)|$
  - Vale decir: $(1 - \kappa^{-2})^i |L(q^i)| \leq N(q^i) \leq (1 + \kappa^{-2})^i |L(q^i)|$

Segundo componente: un sketch a ser usado en la estimación

Para cada estado q^i mantenemos:

- $N(q^i)$: una $(1 \pm \kappa^{-2})^i$-aproximación de $|L(q^i)|$
    - Vale decir: $(1 - \kappa^{-2})^i |L(q^i)| \leq N(q^i) \leq (1 + \kappa^{-2})^i |L(q^i)|$
- $S(q^i)$: un multiconjunto de $2\kappa^7$ muestras uniformes de $L(q^i)$

Segundo componente: un sketch a ser usado en la estimación

Para cada estado q^i mantenemos:

- $N(q^i)$: una $(1 \pm \kappa^{-2})^i$-aproximación de $|L(q^i)|$
  - Vale decir: $(1 - \kappa^{-2})^i |L(q^i)| \leq N(q^i) \leq (1 + \kappa^{-2})^i |L(q^i)|$
- $S(q^i)$: un multiconjunto de $2\kappa^7$ muestras uniformes de $L(q^i)$

Estructura de datos a ser mantenida por el algoritmo:

$$\text{sketch}[i] \quad = \quad \{N(q^j), S(q^j) \mid j \in \{0, \ldots, i\} \text{ y } q^j \text{ es un estado de } \mathcal{A}_{unroll}\}$$

La plantilla del algoritmo

1. Construya $\mathcal{A}_{unroll}$ desde $\mathcal{A}$
2. Para cada estado $q^0 \in I^0$, sea $N(q^0) = |L(q^0)| = 1$ y sea $S(q^0)$ un multiconjunto que contiene $2\kappa^7$ veces la palabra vacía λ
3. Para $i = 0, \ldots, n-1$ y estado $q \in Q$:
   (a) Calcule $N(q^{i+1})$ dado sketch[i]
   (b) Genere de manera uniforme $2\kappa^7$ palabras en $L(q^{i+1})$ utilizando $N(q^{i+1})$ y sketch[i], y defina $S(q^{i+1})$ como el muticonjunto de las muestras obtenidas
4. Retorne una estimación de $|L(F^n)|$ dado sketch[n]

Calculando una estimación $N(F^n)$ de $|L(F^n)|$

Usamos notación $N(X^i)$ para la estimación de $|L(X^i)|$

Calculando una estimación $N(F^n)$ de $|L(F^n)|$

Usamos notación $N(X^i)$ para la estimación de $|L(X^i)|$

Esta estimación no es sólo usada en el último paso del algoritmo, sino que también en la construcción inductiva de sketch[i]:

Calculando una estimación $N(F^n)$ de $|L(F^n)|$

Usamos notación $N(X^i)$ para la estimación de $|L(X^i)|$

Esta estimación no es sólo usada en el último paso del algoritmo, sino que también en la construcción inductiva de sketch$[i]$:

...

3. Para $i = 0, \ldots, n-1$ y estado $q \in Q$:
   (a) Calcule $N(q^{i+1})$ dado sketch$[i]$
   (b) Genere de manera uniforme $2\kappa^7$ palabras en $L(q^{i+1})$ utilizando $N(q^{i+1})$ y sketch$[i]$, y defina $S(q^{i+1})$ como el muticonjunto de las muestras obtenidas

...

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Recuerde que $L(X^i) = \bigcup_{p \in X} L(p^i)$

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Recuerde que $L(X^i) = \bigcup_{p \in X} L(p^i)$

Primero note que $|L(X^i)| = \sum_{p \in X} |L(p^i)|$ **no es cierto** en general

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Recuerde que $L(X^i) = \bigcup_{p \in X} L(p^i)$

Primero note que $|L(X^i)| = \sum_{p \in X} |L(p^i)|$ **no es cierto** en general

- ¿Para que tipo de automatas es cierto esto?

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Recuerde que $L(X^i) = \bigcup_{p \in X} L(p^i)$

Primero note que $|L(X^i)| = \sum_{p \in X} |L(p^i)|$ **no es cierto** en general

- ¿Para que tipo de automatas es cierto esto?

Pero lo siguiente si es cierto dado un orden lineal $<$ sobre Q:

$$|L(X^i)| \;=\; \sum_{p \in X} \Big| L(p^i) \setminus \bigcup_{q \in X \,:\, q < p} L(q^i) \Big|$$

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Tenemos que:

$$|L(X^i)| \quad = \quad \sum_{p \in X} \left| L(p^i) \smallsetminus \bigcup_{q \in X \,:\, q < p} L(q^i) \right|$$

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Tenemos que:

$$
\begin{aligned}
|L(X^i)| &= \sum_{p\in X} \left| L(p^i) \smallsetminus \bigcup_{q\in X\,:\,q<p} L(q^i) \right| \\
&= \sum_{p\in X} \left| L(p^i) \right| \frac{\left| L(p^i) \smallsetminus \bigcup_{q\in X\,:\,q<p} L(q^i) \right|}{\left| L(p^i) \right|}
\end{aligned}
$$

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Tenemos que:

$$\begin{aligned} |L(X^i)| &= \sum_{p \in X} \Big| L(p^i) \setminus \bigcup_{q \in X \,:\, q < p} L(q^i) \Big| \\ &= \sum_{p \in X} \big| L(p^i) \big| \frac{\big| L(p^i) \setminus \bigcup_{q \in X \,:\, q < p} L(q^i) \big|}{\big| L(p^i) \big|} \end{aligned}$$

Entonces usamos la siguiente estimación:

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Tenemos que:

$$\begin{aligned}
|L(X^i)| &= \sum_{p \in X} \left| L(p^i) \setminus \bigcup_{q \in X \,:\, q < p} L(q^i) \right| \\
&= \sum_{p \in X} \left| L(p^i) \right| \frac{\left| L(p^i) \setminus \bigcup_{q \in X \,:\, q < p} L(q^i) \right|}{\left| L(p^i) \right|}
\end{aligned}$$

Entonces usamos la siguiente estimación:

$$\sum_{p \in X}$$

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Tenemos que:

$$\begin{aligned} |L(X^i)| &= \sum_{p \in X} \Big| L(p^i) \setminus \bigcup_{q \in X \,:\, q<p} L(q^i) \Big| \\ &= \sum_{p \in X} \big| L(p^i) \big| \frac{\big| L(p^i) \setminus \bigcup_{q \in X \,:\, q<p} L(q^i) \big|}{\big| L(p^i) \big|} \end{aligned}$$

Entonces usamos la siguiente estimación:

$$\sum_{p \in X} N(p^i)$$

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Tenemos que:

$$\begin{aligned} |L(X^i)| &= \sum_{p \in X} \Big| L(p^i) \setminus \bigcup_{q \in X \,:\, q<p} L(q^i) \Big| \\ &= \sum_{p \in X} \big| L(p^i) \big| \frac{\big| L(p^i) \setminus \bigcup_{q \in X \,:\, q<p} L(q^i) \big|}{\big| L(p^i) \big|} \end{aligned}$$

Entonces usamos la siguiente estimación:

$$\sum_{p \in X} N(p^i) \frac{\big| S(p^i) \setminus \bigcup_{q \in X \,:\, q<p} L(q^i) \big|}{\big| S(p^i) \big|}$$

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

Tenemos que:

$$\begin{aligned} |L(X^i)| &= \sum_{p\in X} \left| L(p^i) \setminus \bigcup_{q\in X\,:\,q<p} L(q^i) \right| \\ &= \sum_{p\in X} \left| L(p^i) \right| \frac{\left| L(p^i) \setminus \bigcup_{q\in X\,:\,q<p} L(q^i) \right|}{\left| L(p^i) \right|} \end{aligned}$$

Entonces usamos la siguiente estimación:

$$N(X^i) = \sum_{p\in X} N(p^i) \frac{\left| S(p^i) \setminus \bigcup_{q\in X\,:\,q<p} L(q^i) \right|}{\left| S(p^i) \right|}$$

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

$N(X^i)$ puede ser calculado en tiempo polinomial

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

$N(X^i)$ puede ser calculado en tiempo polinomial

- $S(p^i) \setminus \bigcup_{q \in X \,:\, q < p} L(q^i)$ es contruido verificando para cada $w \in S(p^i)$ si w no está en $L(q^i)$ para cada $q \in X$ tal que $q < p$

Calculando una estimación $N(X^i)$ de $|L(X^i)|$

$N(X^i)$ puede ser calculado en tiempo polinomial

- $S(p^i) \smallsetminus \bigcup_{q \in X \,:\, q < p} L(q^i)$ es contruido verificando para cada $w \in S(p^i)$ si w no está en $L(q^i)$ para cada $q \in X$ tal que $q < p$

¿Qué garantiza que $N(X^i)$ es una buena estimación de $|L(X^i)|$?

El invariante del algoritmo

$\mathcal{E}(i)$ es cierto si para cada $p \in Q$ y $X \subseteq Q$:

$$\left| \frac{\left|L(p^i) \setminus \bigcup_{q \in X} L(q^i)\right|}{\left|L(p^i)\right|} - \frac{\left|S(p^i) \setminus \bigcup_{q \in X} L(q^i)\right|}{\left|S(p^i)\right|} \right| < \frac{1}{\kappa^3}$$

Una primera consecuencia del invariante

...

3. Para $i = 0, \ldots, n-1$ y estado $q \in Q$:
   (a) Calcule $N(q^{i+1})$ dado sketch$[i]$
   (b) Genere de manera uniforme $2\kappa^7$ palabras en $L(q^{i+1})$ utilizando $N(q^{i+1})$ y sketch$[i]$, y defina $S(q^{i+1})$ como el muticonjunto de las muestras obtenidas

...

Una primera consecuencia del invariante

...

3. Para $i = 0, \ldots, n-1$ y estado $q \in Q$:
   (a) Calcule $N(q^{i+1})$ dado sketch$[i]$
   (b) Genere de manera uniforme $2\kappa^7$ palabras en $L(q^{i+1})$ utilizando $N(q^{i+1})$ y sketch$[i]$, y defina $S(q^{i+1})$ como el muticonjunto de las muestras obtenidas

...

Proposición

Si $\mathcal{E}(i)$ es cierto y $N(p^i)$ es una $(1 \pm \kappa^{-2})^i$-aproximación de $|L(p^i)|$ para cada $p \in Q$, entonces $N(X^i)$ es una $(1 \pm \kappa^{-2})^{i+1}$-aproximación de $|L(X^i)|$ para cada $X \subseteq Q$

La demostración de la proposición

Dado que $\mathcal{E}(i)$ es cierto, para cada $P \subseteq Q$ y $p \in P$:

$$\frac{|L(p^i) \setminus \bigcup_{q \in P\,:\,q<p} L(q^i)|}{|L(p^i)|} - \kappa^{-3} < \frac{|S(p^i) \setminus \bigcup_{q \in P\,:\,q<p} L(q^i)|}{|S(p^i)|} < \frac{|L(p^i) \setminus \bigcup_{q \in P\,:\,q<p} L(q^i)|}{|L(p^i)|} + \kappa^{-3}$$

La demostración de la proposición

Dado que $\mathcal{E}(i)$ es cierto, para cada $P \subseteq Q$ y $p \in P$:

$$\frac{|L(p^i) \setminus \bigcup_{q \in P\,:\,q<p} L(q^i)|}{|L(p^i)|} - \kappa^{-3} \;<\; \frac{|S(p^i) \setminus \bigcup_{q \in P\,:\,q<p} L(q^i)|}{|S(p^i)|} \;<\; \frac{|L(p^i) \setminus \bigcup_{q \in P\,:\,q<p} L(q^i)|}{|L(p^i)|} + \kappa^{-3}$$

Además, dado que $N(p^i)$ es una $(1 \pm \kappa^{-2})^i$-aproximación de $|L(p^i)|$:

$$(1-\kappa^{-2})^i |L(p^i)| \;\leq\; N(p^i) \;\leq\; (1+\kappa^{-2})^i |L(p^i)|.$$

La demostración de la proposición

Recuerde que:

$$N(P^i) \quad = \quad \sum_{p \in P} N(p^i) \frac{\left| S(p^i) \setminus \bigcup_{q \in P \,:\, q < p} L(q^i) \right|}{\left| S(p^i) \right|}$$

La demostración de la proposición

Recuerde que:

$$N(P^i) \;=\; \sum_{p\in P} N(p^i)\frac{\left|S(p^i)\setminus\bigcup_{q\in P\,:\,q<p}L(q^i)\right|}{\left|S(p^i)\right|}$$

Entonces concluimos que:

$$(1-\kappa^{-2})^i\sum_{p\in P}\left(\left|L(p^i)\setminus\bigcup_{q\in P\,:\,q<p}L(q^i)\right|-\kappa^{-3}\left|L(p^i)\right|\right) \;<\; N(P^i) \;<\; (1+\kappa^{-2})^i\sum_{p\in P}\left(\left|L(p^i)\setminus\bigcup_{q\in P\,:\,q<p}L(q^i)\right|+\kappa^{-3}\left|L(p^i)\right|\right).$$

La demostración de la proposición

Dado que $L(p^i) \subseteq L(P^i)|$ y $|P| \leq m \leq \kappa$:

$$\sum_{p \in P} |L(p^i)| \;\leq\; \sum_{p \in P} |L(P^i)| \;=\; |P| \cdot |L(P^i)| \;\leq\; \kappa \cdot |L(P^i)|$$

La demostración de la proposición

Dado que $L(p^i) \subseteq L(P^i)|$ y $|P| \leq m \leq \kappa$:

$$\sum_{p \in P} |L(p^i)| \ \leq \ \sum_{p \in P} |L(P^i)| \ = \ |P| \cdot |L(P^i)| \ \leq \ \kappa \cdot |L(P^i)|$$

Por lo tanto, considerando que $|L(P^i)| = \sum_{p \in P} |L(p^i) \setminus \bigcup_{q \in P \,:\, q < p} L(q^i)|$:

$$(1 - \kappa^{-2})^i(|L(P^i)| - \kappa^{-3} \cdot \kappa |L(P^i)|) \ < \ N(P^i) \ < \\ (1 + \kappa^{-2})^i(|L(P^i)| + \kappa^{-3} \cdot \kappa |L(P^i)|)$$

La demostración de la proposición

Dado que $L(p^i) \subseteq L(P^i)|$ y $|P| \leq m \leq \kappa$:

$$\sum_{p \in P} |L(p^i)| \;\leq\; \sum_{p \in P} |L(P^i)| \;=\; |P| \cdot |L(P^i)| \;\leq\; \kappa \cdot |L(P^i)|$$

Por lo tanto, considerando que $|L(P^i)| = \sum_{p \in P} |L(p^i) \setminus \bigcup_{q \in P \,:\, q < p} L(q^i)|$:

$$(1 - \kappa^{-2})^i(|L(P^i)| - \kappa^{-3} \cdot \kappa |L(P^i)|) \;<\; N(P^i) \;<\; (1 + \kappa^{-2})^i(|L(P^i)| + \kappa^{-3} \cdot \kappa |L(P^i)|)$$

Lo cual es equivalente a:

$$(1 - \kappa^{-2})^{i+1}|L(P^i)| \;<\; N(P^i) \;<\; (1 + \kappa^{-2})^{i+1}|L(P^i)|$$

Usando la proposición

$\mathcal{E}(0)$ es cierto y $N(p^0)$ es una $(1 \pm \kappa^{-2})^0$-aproximación de $|L(p^0)|$ para cada $p \in Q$

- ¿Por qué?

Usando la proposición

$\mathcal{E}(0)$ es cierto y $N(p^0)$ es una $(1 \pm \kappa^{-2})^0$-aproximación de $|L(p^0)|$ para cada $p \in Q$

- ¿Por qué?

Entonces $N(X^0)$ es una $(1 \pm \kappa^{-2})$-aproximación de $|L(X^0)|$ para cada $X \subseteq Q$

Usando la proposición

$\mathcal{E}(0)$ es cierto y $N(p^0)$ es una $(1 \pm \kappa^{-2})^0$-aproximación de $|L(p^0)|$ para cada $p \in Q$

- ¿Por qué?

Entonces $N(X^0)$ es una $(1 \pm \kappa^{-2})$-aproximación de $|L(X^0)|$ para cada $X \subseteq Q$

- Vamos a usar los valores $N(X^0)$ para estimar los valores $N(p^1)$

Usando la proposición

Para cada $p \in Q$:

$$
\begin{aligned}
Y &= \{q^0 \mid (q^0, 0, p^1) \text{ es una transición en } \mathcal{A}_{unroll}\} \\
Z &= \{q^0 \mid (q^0, 1, p^1) \text{ es una transición en } \mathcal{A}_{unroll}\}
\end{aligned}
$$

Usando la proposición

Para cada $p \in Q$:

$$
\begin{array}{rcl}
Y & = & \{q^0 \mid (q^0, 0, p^1) \text{ es una transición en } \mathcal{A}_{unroll}\} \\
Z & = & \{q^0 \mid (q^0, 1, p^1) \text{ es una transición en } \mathcal{A}_{unroll}\}
\end{array}
$$

Tenemos que $L(p^1) \;=\; L(Y) \cdot \{0\} \uplus L(Z) \cdot \{1\}$

Usando la proposición

Para cada $p \in Q$:

$$\begin{array}{rcl} Y & = & \{q^0 \mid (q^0, 0, p^1) \text{ es una transición en } \mathcal{A}_{unroll}\} \\ Z & = & \{q^0 \mid (q^0, 1, p^1) \text{ es una transición en } \mathcal{A}_{unroll}\} \end{array}$$

Tenemos que $L(p^1) \; = \; L(Y) \cdot \{0\} \uplus L(Z) \cdot \{1\}$

- Por lo tanto: $|L(p^1)| = |L(Y)| + |L(Z)|$

Usando la proposición

Para cada $p \in Q$:

$$\begin{array}{rcl} Y & = & \{q^0 \mid (q^0, 0, p^1) \text{ es una transición en } \mathcal{A}_{unroll}\} \\ Z & = & \{q^0 \mid (q^0, 1, p^1) \text{ es una transición en } \mathcal{A}_{unroll}\} \end{array}$$

Tenemos que $L(p^1) \ = \ L(Y) \cdot \{0\} \uplus L(Z) \cdot \{1\}$

- Por lo tanto: $|L(p^1)| = |L(Y)| + |L(Z)|$

Así, dado que $N(Y)$ es una $(1 \pm \kappa^{-2})$-aproximación de $|L(Y)|$ y $N(Z)$ es una $(1 \pm \kappa^{-2})$-aproximación de $|L(Z)|$:

$$N(Y) + N(Z) \text{ es una } (1 \pm \kappa^{-2})\text{-aproximación de } N(p^1)$$

Usando la proposición

$\mathcal{E}(0)$ es cierto y $N(p^0)$ es una $(1 \pm \kappa^{-2})^0$-aproximación de $|L(p^0)|$ para cada $p \in Q$

Usando la proposición

$\mathcal{E}(0)$ es cierto y $N(p^0)$ es una $(1 \pm \kappa^{-2})^0$-aproximación de $|L(p^0)|$ para cada $p \in Q$

$$\Downarrow$$

$N(X^0)$ es una $(1 \pm \kappa^{-2})^1$-aproximación de $|L(X^0)|$ para cada $X \subseteq Q$

Usando la proposición

$\mathcal{E}(0)$ es cierto y $N(p^0)$ es una $(1 \pm \kappa^{-2})^0$-aproximación de $|L(p^0)|$ para cada $p \in Q$

$$\Downarrow$$

$N(X^0)$ es una $(1 \pm \kappa^{-2})^1$-aproximación de $|L(X^0)|$ para cada $X \subseteq Q$

$$\Downarrow$$

$N(p^1) = N(R_0(p^1)) + N(R_1(p^1))$ es una $(1 \pm \kappa^{-2})^1$-aproximación de $N(p^1)$ para cada $p \in Q$

donde $R_b(p^1) = \{q^0 \mid (q^0, b, p^1) \text{ es una transición en } \mathcal{A}_{unroll}\}$

Usando la proposición

$N(p^1)$ es una $(1 \pm \kappa^{-2})^1$-aproximación de $|L(p^1)|$ para cada $p \in Q$

Usando la proposición

$\mathcal{E}(1)$ es cierto y $N(p^1)$ es una $(1 \pm \kappa^{-2})^1$-aproximación de $|L(p^1)|$ para cada $p \in Q$

Usando la proposición

$\mathcal{E}(1)$ es cierto y $N(p^1)$ es una $(1 \pm \kappa^{-2})^1$-aproximación de $|L(p^1)|$ para cada $p \in Q$

$$\Downarrow$$

$N(X^1)$ es una $(1 \pm \kappa^{-2})^2$-aproximación de $|L(X^1)|$ para cada $X \subseteq Q$

Usando la proposición

$\mathcal{E}(1)$ es cierto y $N(p^1)$ es una $(1 \pm \kappa^{-2})^1$-aproximación de $|L(p^1)|$ para cada $p \in Q$

$$\Downarrow$$

$N(X^1)$ es una $(1 \pm \kappa^{-2})^2$-aproximación de $|L(X^1)|$ para cada $X \subseteq Q$

$$\Downarrow$$

$N(p^2) = N(R_0(p^2)) + N(R_1(p^2))$ es una $(1 \pm \kappa^{-2})^2$-aproximación de $N(p^2)$ para cada $p \in Q$

donde $R_b(p^2) = \{q^1 \mid (q^1, b, p^2) \text{ es una transición en } \mathcal{A}_{unroll}\}$

Usando la proposición: resumen

Proposición

Si $\mathcal{E}(j)$ *es cierto para cada* $j \in \{0, 1, \ldots, i\}$*, entonces* $N(p^{i+1})$ *es una* $(1 \pm \kappa^{-2})^{i+1}$*-aproximación de* $|L(p^{i+1})|$ *para cada* $p \in Q$

El resultado final

Proposición

Si $\mathcal{E}(i)$ es cierto para cada $i \in \{0, 1, \ldots, n\}$, entonces $N(F^n)$ es una $(1 \pm \varepsilon)$-aproximación de $|L(F^n)|$

Demostración del resultado final

Utilizando las proposiciones anteriores obtenemos que:

$$(1-\kappa^{-2})^{n+1}|L_n(\mathcal{A})| \;\leq\; N(F^n) \;\leq\; (1+\kappa^{-2})^{n+1}|L_n(\mathcal{A})|$$

Demostración del resultado final

Pero tenemos que:

$$
\begin{aligned}
(1+\kappa^{-2})^{n+1} &\leq \left(1+\left(\frac{\varepsilon}{mn}\right)^2\right)^{n+1} \\
&= \left[\left(1+\left(\frac{1}{(\frac{nm}{\varepsilon})^2}\right)\right)^{(\frac{nm}{\varepsilon})^2}\right]^{\frac{(n+1)\varepsilon^2}{n^2m^2}} \\
&\leq e^{\frac{\varepsilon^2}{m^2}} && \text{dado que } n \geq 2 \\
&\leq 1+2\frac{\varepsilon^2}{m^2} && \text{dado que } e^x \leq (1+2x) \text{ para } x \in [0,1] \\
&= 1+\varepsilon \cdot \frac{2\varepsilon}{m^2} \\
&\leq 1+\varepsilon && \text{dado que } m \geq 2 \text{ and } \varepsilon \in (0,1)
\end{aligned}
$$

Demostración del resultado final

De la misma forma concluimos que:

$$(1-\kappa^{-2})^{n+1} \quad \geq \quad 1-\varepsilon$$

Demostración del resultado final

De la misma forma concluimos que:

$$(1-\kappa^{-2})^{n+1} \quad \geq \quad 1-\varepsilon$$

Poniendo todo junto obtenemos que:

$$(1-\kappa)|L_n(\mathcal{A})| \;\leq\; N(F^n) \;\leq\; (1+\kappa)|L_n(\mathcal{A})|$$

Completando la plantilla del algoritmo

1. Construya $\mathcal{A}_{unroll}$ desde $\mathcal{A}$

2. Para cada estado $q^0 \in I^0$, sea $N(q^0) = |L(q^0)| = 1$ y sea $S(q^0)$ un multiconjunto que contiene $2\kappa^7$ veces la palabra vacía λ

3. Para $i = 0, \ldots, n-1$ y estado $q \in Q$:
   (a) Calcule $N(q^{i+1})$ dado sketch[i]
   (b) Genere de manera uniforme $2\kappa^7$ palabras en $L(q^{i+1})$ utilizando $N(q^{i+1})$ y sketch[i], y defina $S(q^{i+1})$ como el muticonjunto de las muestras obtenidas

4. Retorne el valor $|N(F^n)|$ dado sketch[n]

Muestreando desde un estado

1. Construya $\mathcal{A}_{unroll}$ desde $\mathcal{A}$

2. Para cada estado $q^0 \in I^0$, sea $N(q^0) = |L(q^0)| = 1$ y sea $S(q^0)$ un multiconjunto que contiene $2\kappa^7$ veces la palabra vacía λ

3. Para $i = 0, \ldots, n-1$ y estado $q \in Q$:
   (a) Calcule $N(q^{i+1})$ dado sketch$[i]$
   (b) Genere de manera uniforme $2\kappa^7$ palabras en $L(q^{i+1})$ utilizando $N(q^{i+1})$ y sketch$[i]$, y defina $S(q^{i+1})$ como el muticonjunto de las muestras obtenidas

4. Retorne el valor $|N(F^n)|$ dado sketch$[n]$

Muestreando desde un estado

Tenemos que generar de manera uniforme el multiconjunto de muestras $S(q^{i+1})$

Recuerde que:

- $S(q^{i+1})$ contiene $2\kappa^7$ palabras de $L(q^{i+1})$
- $S(q^{i+1})$ es calculado suponiendo que $N(q^{i+1})$ y sketch[i] ya fueron construidos

Muestreando desde q^{i+1}

Para generar una palabra en $L(q^{i+1})$, construimos una secuencia w^{i+1}, w^i, ..., w^1, w^0 tal que:

Muestreando desde q^{i+1}

Para generar una palabra en $L(q^{i+1})$, construimos una secuencia w^{i+1}, w^i, ..., w^1, w^0 tal que:

- $w^{i+1} = \lambda$

Muestreando desde q^{i+1}

Para generar una palabra en $L(q^{i+1})$, construimos una secuencia w^{i+1}, w^i, ..., w^1, w^0 tal que:

- $w^{i+1} = \lambda$
- $w^j = b_j w^{j+1}$ con $b_j \in \{0, 1\}$

Muestreando desde q^{i+1}

Para generar una palabra en $L(q^{i+1})$, construimos una secuencia w^{i+1}, w^i, ..., w^1, w^0 tal que:

- $w^{i+1} = \lambda$
- $w^j = b_j w^{j+1}$ con $b_j \in \{0, 1\}$
- $w^0 \in L(q^{i+1})$

Muestreando desde q^{i+1}

Para generar una palabra en $L(q^{i+1})$, construimos una secuencia w^{i+1}, w^i, ..., w^1, w^0 tal que:

- $w^{i+1} = \lambda$
- $w^j = b_j w^{j+1}$ con $b_j \in \{0, 1\}$
- $w^0 \in L(q^{i+1})$

Para generar $w^i = bw^{i+1}$, construimos para $b = 0, 1$:

$$P_b \quad = \quad \{p^i \mid (p^i, b, q^{i+1}) \text{ es una transición en } \mathcal{A}_{unroll}\}$$

Muestreando desde q^{i+1}

P_0 y P_1 son conjuntos de estados en la capa i

Muestreando desde q^{i+1}

P_0 y P_1 son conjuntos de estados en la capa i

Elegimos $b \in \{0, 1\}$ con probabilidad

$$\frac{N(P_b)}{N(P_0) + N(P_1)}$$

Recuerde que $N(P_0)$ y $N(P_1)$ son definidos de la siguiente forma:

$$N(X^i) \quad = \quad \sum_{p \in X} N(p^i) \frac{\left| S(p^i) \smallsetminus \bigcup_{q \in X \,:\, q < p} L(q^i) \right|}{\left| S(p^i) \right|}$$

Podemos empezar desde un conjunto de estados

El procedimiento anterior se puede extender a un cojunto de estados P^{i+1}:

$$P_b \quad = \quad \{p^i \mid \exists r^{i+1} \in P^{i+1} : (p^i, b, r^{i+1}) \text{ es una transición en } \mathcal{A}_{unroll}\}$$

En la diapositiva anterior aplicamos el procedimiento a $P^{i+1} = \{q^{i+1}\}$

Podemos empezar desde un conjunto de estados

El procedimiento anterior se puede extender a un cojunto de estados P^{i+1}:

$$P_b \quad = \quad \{p^i \mid \exists r^{i+1} \in P^{i+1} : (p^i, b, r^{i+1}) \text{ es una transición en } \mathcal{A}_{unroll}\}$$

En la diapositiva anterior aplicamos el procedimiento a $P^{i+1} = \{q^{i+1}\}$

El siguiente procedimiento **Sample** implementa estas ideas

- Utiliza una probabilidad φ para asegurar que el muestreo se realice con probabilidad uniforme

El algoritmo de muestreo

Sample(j, P, w, φ)

1. Si $j = 0$, entonces con probabilidad φ retorne w, en caso contrario retorne **fail**

2. Construya para $b = 0, 1$:

$$P_b \quad = \quad \{p^{j-1} \mid \exists r^j \in P : (p^{j-1}, b, r^j) \text{ es una transición en } \mathcal{A}_{unroll}\}$$

3. Elija $b \in \{0, 1\}$ con probabilidad $p_b = \dfrac{N(P_b)}{N(P_0) + N(P_1)}$

4. Retorne **Sample**$(j - 1, P_b, bw, \frac{\varphi}{p_b})$

Observaciones importantes

Cuando queremos muestrear desde un estado q^{i+1} realizamos la llamada **Sample**$(i+1, \{q^{i+1}\}, \lambda, \varphi_0)$

- φ_0 es una probabilidad inicial que depende de $N(q^{i+1})$

Observaciones importantes

Cuando queremos muestrear desde un estado q^{i+1} realizamos la llamada **Sample**$(i+1, \{q^{i+1}\}, \lambda, \varphi_0)$

- φ_0 es una probabilidad inicial que depende de $N(q^{i+1})$

En cada llamada **Sample**(j, P, w, φ) sabemos que P es un conjunto de estados en la capa j

- El procedimiento está bien definido

Observaciones importantes

Sea $x = x_1 \cdots x_{i+1}$ una palabra en $L(q^{i+1})$

Observaciones importantes

Sea $x = x_1 \cdots x_{i+1}$ una palabra en $L(q^{i+1})$

- Suponemos que los conjuntos de estados en las llamadas recursivas son $P^{i+1} = \{q^{i+1}\}$, P^i, ..., P^1, P^0

Observaciones importantes

Sea $x = x_1 \cdots x_{i+1}$ una palabra en $L(q^{i+1})$

- Suponemos que los conjuntos de estados en las llamadas recursivas son $P^{i+1} = \{q^{i+1}\}$, P^i, ..., P^1, P^0

Tenemos que:

$$
\begin{aligned}
&\mathbf{Pr}(\text{la salida de } \textbf{Sample} \text{ es } x) \\
&\quad = \mathbf{Pr}(w^0 = x \wedge \text{ la última llamada a } \textbf{Sample} \text{ no falla}) \\
&\quad = \mathbf{Pr}(\text{la última llamada a } \textbf{Sample} \text{ no falla} \mid w^0 = x) \cdot \mathbf{Pr}(w^0 = x) \\
&\quad = \left(\left(\prod_{j=1}^{i+1} \frac{N(P^j_{x_j})}{N(P^j_0) + N(P^j_1)} \right)^{-1} \cdot \varphi_0 \right) \cdot \left(\prod_{j=1}^{i+1} \frac{N(P^j_{x_j})}{N(P^j_0) + N(P^j_1)} \right) \\
&\quad = \varphi_0
\end{aligned}
$$

El valor de la probabilidad inicial φ_0

Proposición

Suponga que $\mathcal{E}(j)$ *es cierto para cada* $j \in \{0, \ldots, i\}$. *Si* w *es la salida de* $\textbf{Sample}(i+1, \{q^{i+1}\}, \lambda, \frac{e^{-5}}{N(q^{i+1})})$, *entonces:*

- $\varphi \in (0,1)$ *en cada llamada recursiva de* **Sample**
- $\mathbf{Pr}(w = \textbf{fail}) \leq 1 - e^{-9}$
- $\mathbf{Pr}(w = x) = \dfrac{e^{-5}}{N(q^{i+1})}$ *para cada* $x \in L(q^{i+1})$

El valor de la probabilidad inicial φ_0

Proposición

Suponga que $\mathcal{E}(j)$ es cierto para cada $j \in \{0, \ldots, i\}$. Si w es la salida de $\textbf{Sample}(i+1, \{q^{i+1}\}, \lambda, \frac{e^{-5}}{N(q^{i+1})})$*, entonces:*

- $\varphi \in (0,1)$ *en cada llamada recursiva de* **Sample**
- $\textbf{Pr}(w = \textbf{fail}) \leq 1 - e^{-9}$
- $\textbf{Pr}(w = x) = \dfrac{e^{-5}}{N(q^{i+1})}$ *para cada* $x \in L(q^{i+1})$

Condicionado en no fallar, $\textbf{Sample}(i+1, \{q^{i+1}\}, \lambda, \frac{e^{-5}}{N(q^{i+1})})$ genera una palabra en $L(q^{i+1})$ con distribución uniforme

El valor de la probabilidad inicial φ_0

Proposición

Suponga que $\mathcal{E}(j)$ es cierto para cada $j \in \{0, \ldots, i\}$. Si w es la salida de **Sample**$(i+1, \{q^{i+1}\}, \lambda, \frac{e^{-5}}{N(q^{i+1})})$, *entonces:*

- $\varphi \in (0,1)$ *en cada llamada recursiva de* **Sample**
- $\mathbf{Pr}(w = \mathbf{fail}) \leq 1 - e^{-9}$
- $\mathbf{Pr}(w = x) = \dfrac{e^{-5}}{N(q^{i+1})}$ *para cada* $x \in L(q^{i+1})$

Condicionado en no fallar, **Sample**$(i+1, \{q^{i+1}\}, \lambda, \frac{e^{-5}}{N(q^{i+1})})$ genera una palabra en $L(q^{i+1})$ con distribución uniforme

Ejercicio

Demuestre la proposición

Acotando la probabilidad de que se cumpla la invariante

Recuerde que $\mathcal{E}(i)$ se cumple si para cada $q \in Q$ y $X \subseteq Q$:

$$\left| \; \frac{\left|L(q^i) \smallsetminus \bigcup_{p \in X} L(p^i)\right|}{\left|L(q^i)\right|} \; - \; \frac{\left|S(q^i) \smallsetminus \bigcup_{p \in X} L(p^i)\right|}{\left|S(q^i)\right|} \; \right| \quad < \quad \frac{1}{\kappa^3}$$

Acotando la probabilidad de que se cumpla la invariante

Recuerde que $\mathcal{E}(i)$ se cumple si para cada $q \in Q$ y $X \subseteq Q$:

$$\left| \frac{\left|L(q^i) \smallsetminus \bigcup_{p \in X} L(p^i)\right|}{\left|L(q^i)\right|} - \frac{\left|S(q^i) \smallsetminus \bigcup_{p \in X} L(p^i)\right|}{\left|S(q^i)\right|} \right| < \frac{1}{\kappa^3}$$

Sabemos que $\mathcal{E}(0)$ se cumple.

Acotando la probabilidad de que se cumpla la invariante

Recuerde que $\mathcal{E}(i)$ se cumple si para cada $q \in Q$ y $X \subseteq Q$:

$$\left| \frac{\left|L(q^i) \smallsetminus \bigcup_{p \in X} L(p^i)\right|}{\left|L(q^i)\right|} - \frac{\left|S(q^i) \smallsetminus \bigcup_{p \in X} L(p^i)\right|}{\left|S(q^i)\right|} \right| \quad < \quad \frac{1}{\kappa^3}$$

Sabemos que $\mathcal{E}(0)$ se cumple. Necesitamos calcular una cota inferior para:

$$\mathbf{Pr}\left(\bigwedge_{j=0}^{n} \mathcal{E}(j) \right)$$

Acotando la probabilidad de que se cumpla la invariante

Recuerde que $\mathcal{E}(i)$ se cumple si para cada $q \in Q$ y $X \subseteq Q$:

$$\left| \frac{\left|L(q^i) \smallsetminus \bigcup_{p \in X} L(p^i)\right|}{\left|L(q^i)\right|} - \frac{\left|S(q^i) \smallsetminus \bigcup_{p \in X} L(p^i)\right|}{\left|S(q^i)\right|} \right| < \frac{1}{\kappa^3}$$

Sabemos que $\mathcal{E}(0)$ se cumple. Necesitamos calcular una cota inferior para:

$$\mathbf{Pr}\Big(\bigwedge_{j=0}^{n} \mathcal{E}(j) \Big)$$

Para calcular una cota inferior vamos a necesitar la desigualdad de Hoeffding

La desigualdad de Hoeffding

Teorema

Sean $X_1, \ldots, X_t$ variables aleatorias independientes tales que $a \leq X_i \leq b$ y $\mathbf{E}[X_i] = \mu$ para cada $i \in \{1, \ldots, t\}$. Entonces para cada $\delta > 0$:

$$\mathbf{Pr}\left(\left| \frac{1}{t} \sum_{i=1}^{t} X_i - \mu \right| \geq \delta \right) \quad \leq \quad 2e^{\frac{-2t\delta^2}{(b-a)^2}}$$

La desigualdad de Hoeffding

Teorema

Sean $X_1, \ldots, X_t$ *variables aleatorias independientes tales que* $a \leq X_i \leq b$ *y* $\mathbf{E}[X_i] = \mu$ *para cada* $i \in \{1, \ldots, t\}$. *Entonces para cada* $\delta > 0$*:*

$$\mathbf{Pr}\left(\left|\frac{1}{t}\sum_{i=1}^{t} X_i - \mu\right| \geq \delta\right) \quad \leq \quad 2e^{\frac{-2t\delta^2}{(b-a)^2}}$$

Vamos a utilizar el siguiente corolario:

Corolario

Sean $X_1, \ldots, X_t$ *variables aleatorias independientes tales que* $0 \leq X_i \leq 1$ *y* $\mathbf{E}[X_i] = \mu$ *para cada* $i \in \{1, \ldots, t\}$. *Entonces para cada* $\delta > 0$*:*

$$\mathbf{Pr}\left(\left|\frac{1}{t}\sum_{i=1}^{t} X_i - \mu\right| \geq \delta\right) \quad \leq \quad 2e^{-2t\delta^2}$$

Lema de Hoeffding

Lema

Sea X una variable aleatoria tal que $a \leq X \leq b$. Entonces para cada $\delta \in \mathbb{R}$:

$$\mathbf{E}[e^{\delta(X-\mathbf{E}[X])}] \quad \leq \quad e^{\frac{\delta^2(b-a)^2}{8}}$$

Lema de Hoeffding

Lema

Sea X una variable aleatoria tal que $a \leq X \leq b$. Entonces para cada $\delta \in \mathbb{R}$:

$$\mathbf{E}[e^{\delta(X - \mathbf{E}[X])}] \quad \leq \quad e^{\frac{\delta^2 (b-a)^2}{8}}$$

Vamos a demostrar una versión más simple del lema:

$$\mathbf{E}[e^{\delta(X - \mathbf{E}[X])}] \quad \leq \quad e^{\frac{\delta^2 (b-a)^2}{2}}$$

Esta versión tienen los ingredientes principales de la demostración

La demostración de lema de Hoeffding

Dado que la función $e^{\delta x}$ es convexa, para $x \in [a-b, b-a]$:

$$e^{\delta x} \quad \leq \quad \frac{(b-a)-x}{2(b-a)} e^{\delta(a-b)} + \frac{x-(a-b)}{2(b-a)} e^{\delta(b-a)}$$

La demostración de lema de Hoeffding

Dado que la función $e^{\delta x}$ es convexa, para $x \in [a-b, b-a]$:

$$e^{\delta x} \quad \leq \quad \frac{(b-a)-x}{2(b-a)} e^{\delta(a-b)} + \frac{x-(a-b)}{2(b-a)} e^{\delta(b-a)}$$

Dado que $a-b \leq X - \mathbf{E}[X] \leq b-a$:

$$e^{\delta(X-\mathbf{E}[X])} \quad \leq \quad \frac{(b-a)-(X-\mathbf{E}[X])}{2(b-a)} e^{\delta(a-b)} + \frac{(X-\mathbf{E}[X])-(a-b)}{2(b-a)} e^{\delta(b-a)}$$

La demostración de lema de Hoeffding

Concluimos que:

$$\begin{aligned}\mathbf{E}[e^{\delta(X-\mathbf{E}[X])}] &\leq \frac{1}{2}e^{\delta(a-b)} + \frac{1}{2}e^{\delta(b-a)} \\ &= e^{\ln(\frac{1}{2}e^{\delta(a-b)} + \frac{1}{2}e^{\delta(b-a)})}\end{aligned}$$

La demostración de lema de Hoeffding

Concluimos que:

$$\begin{aligned} \mathbf{E}[e^{\delta(X-\mathbf{E}[X])}] &\leq \frac{1}{2}e^{\delta(a-b)} + \frac{1}{2}e^{\delta(b-a)} \\ &= e^{\ln(\frac{1}{2}e^{\delta(a-b)}+\frac{1}{2}e^{\delta(b-a)})} \end{aligned}$$

Defina la función $F(x) = \ln(\frac{1}{2}e^{-x} + \frac{1}{2}e^{x})$. Tenemos que:

$$\mathbf{E}[e^{\delta(X-\mathbf{E}[X])}] \leq e^{F(\delta(b-a))}$$

La demostración de lema de Hoeffding

Concluimos que:

$$\begin{aligned} \mathbf{E}[e^{\delta(X-\mathbf{E}[X])}] &\leq \frac{1}{2}e^{\delta(a-b)} + \frac{1}{2}e^{\delta(b-a)} \\ &= e^{\ln(\frac{1}{2}e^{\delta(a-b)}+\frac{1}{2}e^{\delta(b-a)})} \end{aligned}$$

Defina la función $F(x) = \ln(\frac{1}{2}e^{-x} + \frac{1}{2}e^{x})$. Tenemos que:

$$\mathbf{E}[e^{\delta(X-\mathbf{E}[X])}] \leq e^{F(\delta(b-a))}$$

Además, tenemos que $F(0) = 0$

La demostración de lema de Hoeffding

Sabemos que:

$$
\begin{aligned}
F'(x) &= \frac{e^x - e^{-x}}{e^x + e^{-x}} \\
F''(x) &= 1 - \left(\frac{e^x - e^{-x}}{e^x + e^{-x}}\right)^2
\end{aligned}
$$

La demostración de lema de Hoeffding

Sabemos que:

$$
\begin{aligned}
F'(x) &= \frac{e^x - e^{-x}}{e^x + e^{-x}} \\
F''(x) &= 1 - \left(\frac{e^x - e^{-x}}{e^x + e^{-x}}\right)^2
\end{aligned}
$$

Por lo tanto, $F'(0) = 0$ y $F''(x) \leq 1$ para todo $x \in \mathbb{R}$

La demostración de lema de Hoeffding

Sabemos que:

$$\begin{aligned} F'(x) &= \frac{e^x - e^{-x}}{e^x + e^{-x}} \\ F''(x) &= 1 - \left(\frac{e^x - e^{-x}}{e^x + e^{-x}}\right)^2 \end{aligned}$$

Por lo tanto, $F'(0) = 0$ y $F''(x) \leq 1$ para todo $x \in \mathbb{R}$

Para todo $x \geq 0$, existe $\xi \in [0, x]$ tal que:

$$F(x) = F(0) + F'(0)x + F''(\xi)\frac{x^2}{2}$$

La demostración de lema de Hoeffding

Dado que $F(0) = 0$, $F'(0) = 0$ y $F''(\xi) \leq 1$, para todo $x \geq 0$:

$$F(x) \leq \frac{x^2}{2}$$

La demostración de lema de Hoeffding

Dado que $F(0) = 0$, $F'(0) = 0$ y $F''(\xi) \leq 1$, para todo $x \geq 0$:

$$F(x) \quad \leq \quad \frac{x^2}{2}$$

En particular: $F(\delta(b-a)) \leq \frac{\delta^2(b-a)^2}{2}$

La demostración de lema de Hoeffding

Dado que $F(0) = 0$, $F'(0) = 0$ y $F''(\xi) \leq 1$, para todo $x \geq 0$:

$$F(x) \leq \frac{x^2}{2}$$

En particular: $F(\delta(b-a)) \leq \frac{\delta^2(b-a)^2}{2}$

De esto se deduce que:

$$\mathbf{E}[e^{\delta(X-\mathbf{E}[X])}] \leq e^{F(\delta(b-a))} \leq e^{\frac{\delta^2(b-a)^2}{2}}$$

La demostración de la desigualdad de Hoeffding

Sean $X_1, \ldots, X_t$ variables aleatorias independientes tales que $a \leq X_i \leq b$ y $\mathbf{E}[X_i] = \mu$ para cada $i \in \{1, \ldots, t\}$, y sea $\delta > 0$

La demostración de la desigualdad de Hoeffding

Sean $X_1, \ldots, X_t$ variables aleatorias independientes tales que $a \leq X_i \leq b$ y $\mathbf{E}[X_i] = \mu$ para cada $i \in \{1, \ldots, t\}$, y sea $\delta > 0$

Dado $\theta > 0$, tenemos que:

$$\begin{aligned}
\mathbf{Pr}\left(\frac{1}{t}\sum_{i=1}^{t} X_i - \mu \geq \delta\right) &= \mathbf{Pr}\left(\frac{1}{t}\sum_{i=1}^{t}(X_i - \mu) \geq \delta\right) \\
&= \mathbf{Pr}\left(\sum_{i=1}^{t}(X_i - \mu) \geq \delta t\right) \\
&= \mathbf{Pr}\left(\theta\sum_{i=1}^{t}(X_i - \mu) \geq \theta\delta t\right) \\
&= \mathbf{Pr}\left(e^{\theta\sum_{i=1}^{t}(X_i - \mu)} \geq e^{\theta\delta t}\right)
\end{aligned}$$

La demostración de la desigualdad de Hoeffding

Utilizando la desigualdad de Markov y el lema de Hoeffding:

$$
\begin{aligned}
\mathbf{Pr}\left(e^{\theta\sum_{i=1}^{t}(X_i-\mu)} \geq e^{\theta\delta t}\right) &\leq \frac{\mathbf{E}[e^{\theta\sum_{i=1}^{t}(X_i-\mu)}]}{e^{\theta\delta t}} \\
&= \frac{\mathbf{E}[\prod_{i=1}^{t} e^{\theta(X_i-\mu)}]}{e^{\theta\delta t}} \\
&= \frac{\prod_{i=1}^{n}\mathbf{E}[e^{\theta(X_i-\mu)}]}{e^{\theta\delta t}} \\
&\leq \frac{\prod_{i=1}^{t} e^{\frac{\theta^2(b-a)^2}{8}}}{e^{\theta\delta t}} \\
&= e^{\frac{t\theta^2(b-a)^2}{8}-\theta\delta t}
\end{aligned}
$$

La demostración de la desigualdad de Hoeffding

La función $g(\theta) = \frac{t\theta^2(b-a)^2}{8} - \theta\delta t$ alcanza su menor valor en $\theta = \frac{4\delta}{(b-a)^2}$

La demostración de la desigualdad de Hoeffding

La función $g(\theta) = \frac{t\theta^2(b-a)^2}{8} - \theta\delta t$ alcanza su menor valor en $\theta = \frac{4\delta}{(b-a)^2}$

Por lo tanto, dado que la ecuación en la diapositiva enterior es cierta para todo $\theta > 0$:

$$\begin{aligned}
\mathbf{Pr}\left(\frac{1}{t}\sum_{i=1}^{t} X_i - \mu \geq \delta\right) &\leq \mathbf{Pr}\left(e^{\theta\sum_{i=1}^{t}(X_i-\mu)} \geq e^{\theta\delta t}\right) \\
&\leq e^{\frac{t\theta^2(b-a)^2}{8} - \theta\delta t} \\
&\leq e^{\frac{-2t\delta^2}{(b-a)^2}}
\end{aligned}$$

La demostración de la desigualdad de Hoeffding

Finalmente:

$$
\begin{aligned}
&\mathbf{Pr}\left(\left|\frac{1}{t}\sum_{i=1}^{t} X_i - \mu\right| \geq \delta\right) \ = \\
&\qquad \mathbf{Pr}\left(\frac{1}{t}\sum_{i=1}^{t} X_i - \mu \geq \delta \vee \frac{1}{t}\sum_{i=1}^{t} X_i - \mu \leq -\delta\right) \ \leq \\
&\qquad \mathbf{Pr}\left(\frac{1}{t}\sum_{i=1}^{t} X_i - \mu \geq \delta\right) + \mathbf{Pr}\left(\frac{1}{t}\sum_{i=1}^{t} X_i - \mu \leq -\delta\right) \ = \\
&\qquad \mathbf{Pr}\left(\frac{1}{t}\sum_{i=1}^{t} X_i - \mu \geq \delta\right) + \mathbf{Pr}\left(\frac{1}{t}\sum_{i=1}^{t} (-X_i) - (-\mu) \geq \delta\right) \ \leq \\
&\qquad e^{\frac{-2t\delta^2}{(b-a)^2}} + e^{\frac{-2t\delta^2}{(-a-(-b))^2}} \ = \ 2e^{\frac{-2t\delta^2}{(b-a)^2}}
\end{aligned}
$$

Acotando la probabilidad de que se cumpla la invariante

Suponga que $\bigwedge_{j=0}^{i-1} \mathcal{E}(j)$ se cumple

Sea $q \in Q$ y sea $S(q^i)$ el multiconjunto de $2\kappa^7$ muestras de $L(q^i)$ construido llamando a $\textbf{Sample}(i, \{q^i\}, \lambda, \frac{e^{-5}}{N(q^i)})$

- Cada palabra de $S(q^i)$ es obtenida llamando a **Sample** hasta que la salida es distinta de **fail**

Acotando la probabilidad de que se cumpla la invariante

Suponga que $\bigwedge_{j=0}^{i-1} \mathcal{E}(j)$ se cumple

Sea $q \in Q$ y sea $S(q^i)$ el multiconjunto de $2\kappa^7$ muestras de $L(q^i)$ construido llamando a **Sample**$(i, \{q^i\}, \lambda, \frac{e^{-5}}{N(q^i)})$

- Cada palabra de $S(q^i)$ es obtenida llamando a **Sample** hasta que la salida es distinta de **fail**

Suponga que $S(q^i) = \{w_1, \ldots, w_t\}$ con $t = 2\kappa^7$

Acotando la probabilidad de que se cumpla la invariante

Sea $X \subseteq Q$, y para $i \in \{1, \ldots, t\}$ sea Y_i una variable aleatoria:

$$Y_i = 1 \quad \text{si y sólo si} \quad w_i \in \left(L(q^i) \smallsetminus \bigcup_{p \in X} L(p^i) \right)$$

Acotando la probabilidad de que se cumpla la invariante

Sea $X \subseteq Q$, y para $i \in \{1, \ldots, t\}$ sea Y_i una variable aleatoria:

$$Y_i = 1 \quad \text{si y sólo si} \quad w_i \in \left(L(q^i) \smallsetminus \bigcup_{p \in X} L(p^i) \right)$$

Tenemos que:

$$\begin{aligned}
\mathbf{E}[Y_i] &= \frac{|L(q^i) \smallsetminus \bigcup_{p \in X} L(p^i)|}{|L(q^i)|} \\
\sum_{j=1}^{t} Y_i &= |S(q^i) \smallsetminus \bigcup_{p \in X} L(p^i)| \\
t &= |S(q^i)|
\end{aligned}$$

Utilizando la desigualdad de Hoeffding

$$\mathbf{Pr}\Bigg(\Bigg|\frac{|S(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|S(q^i)|} - \frac{|L(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|L(q^i)|}\Bigg| \geq \frac{1}{\kappa^3} \;\Bigg|\; \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\Bigg)$$

Utilizando la desigualdad de Hoeffding

$$\mathbf{Pr}\Bigg(\Bigg|\frac{|S(q^i) \setminus \bigcup_{p\in X} L(p^i)|}{|S(q^i)|} - \frac{|L(q^i) \setminus \bigcup_{p\in X} L(p^i)|}{|L(q^i)|}\Bigg| \geq \frac{1}{\kappa^3} \;\Bigg|\; \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\Bigg) =$$

$$\mathbf{Pr}\Bigg(\Bigg|\frac{1}{t}\sum_{j=1}^{t} Y_i - \mathbf{E}\Bigg[\frac{1}{t}\sum_{j=1}^{t} Y_i\Bigg]\Bigg| \geq \frac{1}{\kappa^3} \;\Bigg|\; \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\Bigg)$$

Utilizando la desigualdad de Hoeffding

$$\mathbf{Pr}\left(\left| \frac{|S(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|S(q^i)|} - \frac{|L(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|L(q^i)|} \right| \geq \frac{1}{\kappa^3} \; \middle| \bigwedge_{j=0}^{i-1} \mathcal{E}(j) \right) =$$

$$\mathbf{Pr}\left(\left| \frac{1}{t} \sum_{j=1}^{t} Y_i - \mathbf{E}\left[\frac{1}{t} \sum_{j=1}^{t} Y_i \right] \right| \geq \frac{1}{\kappa^3} \; \middle| \bigwedge_{j=0}^{i-1} \mathcal{E}(j) \right) \leq 2e^{-2\left(\frac{1}{\kappa^3}\right)^2 t}$$

Utilizando la desigualdad de Hoeffding

$$\mathbf{Pr}\left(\left| \frac{|S(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|S(q^i)|} - \frac{|L(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|L(q^i)|} \right| \geq \frac{1}{\kappa^3} \;\; \middle| \bigwedge_{j=0}^{i-1} \mathcal{E}(j) \right) =$$

$$\mathbf{Pr}\left(\left| \frac{1}{t} \sum_{j=1}^{t} Y_i - \mathbf{E}\left[\frac{1}{t} \sum_{j=1}^{t} Y_i \right] \right| \geq \frac{1}{\kappa^3} \;\; \middle| \bigwedge_{j=0}^{i-1} \mathcal{E}(j) \right) \leq 2e^{-2\left(\frac{1}{\kappa^3}\right)^2 t}$$

$$= 2e^{-2\left(\frac{1}{\kappa^6}\right) 2\kappa^7}$$

Utilizando la desigualdad de Hoeffding

$$\mathbf{Pr}\left(\left|\frac{|S(q^i) \setminus \bigcup_{p\in X} L(p^i)|}{|S(q^i)|} - \frac{|L(q^i) \setminus \bigcup_{p\in X} L(p^i)|}{|L(q^i)|}\right| \geq \frac{1}{\kappa^3} \;\middle|\; \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\right) =$$

$$\mathbf{Pr}\left(\left|\frac{1}{t}\sum_{j=1}^{t} Y_i - \mathbf{E}\left[\frac{1}{t}\sum_{j=1}^{t} Y_i\right]\right| \geq \frac{1}{\kappa^3} \;\middle|\; \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\right) \leq 2e^{-2\left(\frac{1}{\kappa^3}\right)^2 t}$$

$$= 2e^{-2\left(\frac{1}{\kappa^6}\right)2\kappa^7}$$

$$= 2e^{-4\kappa}$$

Utilizando la cota de la unión

$$\mathbf{Pr}\Bigg(\exists q \in Q \, \exists X \subseteq Q \left| \frac{|S(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|S(q^i)|} - \right.$$

$$\left. \frac{|L(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|L(q^i)|} \right| \geq \frac{1}{\kappa^3} \, \Bigg| \bigwedge_{j=0}^{i-1} \mathcal{E}(j) \Bigg) \leq$$

Utilizando la cota de la unión

$$\mathbf{Pr}\Bigg(\exists q \in Q\, \exists X \subseteq Q \left| \frac{|S(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|S(q^i)|} - \right.$$

$$\left. \frac{|L(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|L(q^i)|} \right| \geq \frac{1}{\kappa^3} \ \Bigg| \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\Bigg) \ \leq$$

$$\sum_{q \in Q} \sum_{X \subseteq Q} \mathbf{Pr}\Bigg(\left| \frac{|S(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|S(q^i)|} - \right.$$

$$\left. \frac{|L(q^i) \setminus \bigcup_{p \in X} L(p^i)|}{|L(q^i)|} \right| \geq \frac{1}{\kappa^3} \ \Bigg| \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\Bigg) \ \leq$$

Utilizando la cota de la unión

$$\mathbf{Pr}\Bigg(\exists q \in Q\, \exists X \subseteq Q \left| \frac{|S(q^i) \setminus \bigcup_{p\in X} L(p^i)|}{|S(q^i)|} - \frac{|L(q^i) \setminus \bigcup_{p\in X} L(p^i)|}{|L(q^i)|} \right| \geq \frac{1}{\kappa^3} \ \Bigg| \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\Bigg) \leq$$

$$\sum_{q\in Q}\sum_{X\subseteq Q} \mathbf{Pr}\Bigg(\left| \frac{|S(q^i) \setminus \bigcup_{p\in X} L(p^i)|}{|S(q^i)|} - \frac{|L(q^i) \setminus \bigcup_{p\in X} L(p^i)|}{|L(q^i)|} \right| \geq \frac{1}{\kappa^3} \ \Bigg| \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\Bigg) \leq$$

$$m2^m 2e^{-4\kappa} \leq \kappa 2^\kappa 2e^{-4\kappa} \leq 2e^{-2\kappa}$$

La conclusión

El resultado anterior nos dice que:

$$\mathbf{Pr}\left(\mathcal{E}(i) \;\middle|\; \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\right) \;\; \geq \;\; 1 - e^{-2\kappa}$$

La conclusión

El resultado anterior nos dice que:

$$\mathbf{Pr}\Big(\mathcal{E}(i) \;\Big|\; \bigwedge_{j=0}^{i-1} \mathcal{E}(j)\Big) \;\geq\; 1 - e^{-2\kappa}$$

Por lo tanto:

$$\begin{aligned}\mathbf{Pr}\Big(\bigwedge_{i=0}^{n} \mathcal{E}(j)\Big) \;&=\; \prod_{i=1}^{n} \mathbf{Pr}(\mathcal{E}(i) \mid \bigwedge_{j=0}^{i-1} \mathcal{E}(j)) \\ &\geq\; \prod_{i=1}^{n} (1 - e^{-2\kappa}) \;=\; (1 - e^{-2\kappa})^{n}\end{aligned}$$

La conclusión

Pero tenemos que:

$$\begin{aligned}
(1 - e^{-2\kappa})^n &= 1 + \sum_{j=1}^{n} \binom{n}{j} (-1)^j e^{-2\kappa \cdot j} \\
&\geq 1 - \sum_{j=1}^{n} \binom{n}{j} e^{-2\kappa \cdot j} \\
&\geq 1 - e^{-2\kappa} \cdot \sum_{j=1}^{n} \binom{n}{j} \\
&\geq 1 - e^{-2\kappa} \cdot 2^n \\
&\geq 1 - e^{-2\kappa} \cdot e^{\kappa} = 1 - e^{-\kappa}
\end{aligned}$$

La conclusión

Pero tenemos que:

$$\begin{aligned}
(1-e^{-2\kappa})^n &= 1+\sum_{j=1}^{n}\binom{n}{j}(-1)^j e^{-2\kappa\cdot j} \\
&\geq 1-\sum_{j=1}^{n}\binom{n}{j} e^{-2\kappa\cdot j} \\
&\geq 1-e^{-2\kappa}\cdot\sum_{j=1}^{n}\binom{n}{j} \\
&\geq 1-e^{-2\kappa}\cdot 2^n \\
&\geq 1-e^{-2\kappa}\cdot e^{\kappa} = 1-e^{-\kappa}
\end{aligned}$$

Proposición

La probabilidad de que $\mathcal{E}(i)$ se cumpla para todas las capas $i \in \{0,\dots,n\}$ es al menos $1-e^{-\kappa}$

El algoritmo completo

Entrada: NFA $\mathcal{A} = (Q, \{0,1\}, \Delta, I, F)$ con $m = |Q|$, largo n dado en unario y error $\varepsilon \in (0,1)$

El algoritmo completo

Entrada: NFA $\mathcal{A} = (Q, \{0,1\}, \Delta, I, F)$ con $m = |Q|$, largo n dado en unario y error $\varepsilon \in (0,1)$

Suponemos dados los siguientes procedimientos:

- **CountOneLength**($\mathcal{A}$): cuenta el número de palabras de largo 1 aceptadas por $\mathcal{A}$
- **CountSingleState**($\mathcal{A}, n$): cuenta el número de palabras de largo n aceptadas por $\mathcal{A}$, suponiendo que $\mathcal{A}$ tiene un estado

El algoritmo completo

El algoritmo completo

1. Si $L_n(\mathcal{A}) = \emptyset$, entonces retorne 0

El algoritmo completo

1. Si $L_n(\mathcal{A}) = \emptyset$, entonces retorne 0
2. Si $n = 0$, entonces retorne 1

El algoritmo completo

1. Si $L_n(\mathcal{A}) = \emptyset$, entonces retorne 0
2. Si $n = 0$, entonces retorne 1
3. Si $n = 1$, entonces retorne $\textbf{CountOneLength}(\mathcal{A})$

El algoritmo completo

1. Si $L_n(\mathcal{A}) = \emptyset$, entonces retorne 0
2. Si $n = 0$, entonces retorne 1
3. Si $n = 1$, entonces retorne **CountOneLength**$(\mathcal{A})$
4. Si $m = 1$, entonces retorne **CountSingleState**$(\mathcal{A}, n)$

El algoritmo completo

1. Si $L_n(\mathcal{A}) = \emptyset$, entonces retorne 0
2. Si $n = 0$, entonces retorne 1
3. Si $n = 1$, entonces retorne **CountOneLength**$(\mathcal{A})$
4. Si $m = 1$, entonces retorne **CountSingleState**$(\mathcal{A}, n)$
5. Construya $\mathcal{A}_{unroll}$ y defina $\kappa = \lceil \frac{nm}{\varepsilon} \rceil$

El algoritmo completo

1. Si $L_n(\mathcal{A}) = \emptyset$, entonces retorne 0
2. Si $n = 0$, entonces retorne 1
3. Si $n = 1$, entonces retorne **CountOneLength**$(\mathcal{A})$
4. Si $m = 1$, entonces retorne **CountSingleState**$(\mathcal{A}, n)$
5. Construya $\mathcal{A}_{unroll}$ y defina $\kappa = \lceil \frac{nm}{\varepsilon} \rceil$
6. Elimine desde $\mathcal{A}_{unroll}$ cada nodo q^i que no es parte de un camino desde un estado inicial p^0 $(p \in I)$ a un estado final r^n $(r \in F)$

El algoritmo completo

1. Si $L_n(\mathcal{A}) = \emptyset$, entonces retorne 0
2. Si $n = 0$, entonces retorne 1
3. Si $n = 1$, entonces retorne **CountOneLength**$(\mathcal{A})$
4. Si $m = 1$, entonces retorne **CountSingleState**$(\mathcal{A}, n)$
5. Construya $\mathcal{A}_{unroll}$ y defina $\kappa = \lceil \frac{nm}{\varepsilon} \rceil$
6. Elimine desde $\mathcal{A}_{unroll}$ cada nodo q^i que no es parte de un camino desde un estado inicial p^0 $(p \in I)$ a un estado final r^n $(r \in F)$
7. Para cada estado $q^0 \in I^0$, sea $N(q^0) = |L(q^0)| = 1$ y sea $S(q^0)$ un multiconjunto que contiene $2\kappa^7$ veces la palabra vacía λ

El algoritmo completo

8. Para cada capa $i = 1, \ldots, n$ y estado q^i en $\mathcal{A}_{unroll}$:

El algoritmo completo

8. Para cada capa $i = 1, \ldots, n$ y estado q^i en $\mathcal{A}_{unroll}$:

   8.1 Sea $R_b = \{p^{i-1} \mid (p^{i-1}, b, q^i) \text{ es una transición en } \mathcal{A}_{unroll}\}$ para $b = 0, 1$

El algoritmo completo

8. Para cada capa $i = 1, \ldots, n$ y estado q^i en $\mathcal{A}_{unroll}$:

    8.1 Sea $R_b = \{p^{i-1} \mid (p^{i-1}, b, q^i)$ es una transición en $\mathcal{A}_{unroll}\}$ para $b = 0, 1$

    8.2 Sea $N(q^i) = N(R_0) + N(R_1)$

El algoritmo completo

8. Para cada capa $i = 1, \ldots, n$ y estado q^i en $\mathcal{A}_{unroll}$:

    8.1 Sea $R_b = \{p^{i-1} \mid (p^{i-1}, b, q^i)$ es una transición en $\mathcal{A}_{unroll}\}$ para $b = 0, 1$

    8.2 Sea $N(q^i) = N(R_0) + N(R_1)$

    8.3 Sea $S(q^i) = \emptyset$. Mientras $|S(q^i)| < 2\kappa^7$:

El algoritmo completo

8. Para cada capa $i = 1, \ldots, n$ y estado q^i en $\mathcal{A}_{unroll}$:

   8.1 Sea $R_b = \{p^{i-1} \mid (p^{i-1}, b, q^i)$ es una transición en $\mathcal{A}_{unroll}\}$ para $b = 0, 1$

   8.2 Sea $N(q^i) = N(R_0) + N(R_1)$

   8.3 Sea $S(q^i) = \emptyset$. Mientras $|S(q^i)| < 2\kappa^7$:

      8.3.1 Ejecute **Sample**$(i, \{q^i\}, \lambda, \frac{e^{-5}}{N(q^i)})$ hasta que retorne $w \neq$ **fail**, pero a lo más $c(\kappa)$ veces

El algoritmo completo

8. Para cada capa $i = 1, \ldots, n$ y estado q^i en $\mathcal{A}_{unroll}$:

    8.1 Sea $R_b = \{p^{i-1} \mid (p^{i-1}, b, q^i)$ es una transición en $\mathcal{A}_{unroll}\}$ para $b = 0, 1$

    8.2 Sea $N(q^i) = N(R_0) + N(R_1)$

    8.3 Sea $S(q^i) = \emptyset$. Mientras $|S(q^i)| < 2\kappa^7$:

        8.3.1 Ejecute **Sample**$(i, \{q^i\}, \lambda, \frac{e^{-5}}{N(q^i)})$ hasta que retorne $w \neq$ **fail**, pero a lo más $c(\kappa)$ veces

        8.3.2 Si $w =$ **fail**, entonces retorne 0

El algoritmo completo

8. Para cada capa $i = 1, \ldots, n$ y estado q^i en $\mathcal{A}_{unroll}$:

   8.1 Sea $R_b = \{p^{i-1} \mid (p^{i-1}, b, q^i)$ es una transición en $\mathcal{A}_{unroll}\}$ para $b = 0, 1$

   8.2 Sea $N(q^i) = N(R_0) + N(R_1)$

   8.3 Sea $S(q^i) = \emptyset$. Mientras $|S(q^i)| < 2\kappa^7$:

      8.3.1 Ejecute **Sample**$(i, \{q^i\}, \lambda, \frac{e^{-5}}{N(q^i)})$ hasta que retorne $w \neq$ **fail**, pero a lo más $c(\kappa)$ veces

      8.3.2 Si $w =$ **fail**, entonces retorne 0

      8.3.3 Sea $S(q^i) = S(q^i) \cup \{w\}$

El algoritmo completo

8. Para cada capa $i = 1, \dots, n$ y estado q^i en $\mathcal{A}_{unroll}$:

   8.1 Sea $R_b = \{p^{i-1} \mid (p^{i-1}, b, q^i) \text{ es una transición en } \mathcal{A}_{unroll}\}$ para $b = 0, 1$

   8.2 Sea $N(q^i) = N(R_0) + N(R_1)$

   8.3 Sea $S(q^i) = \emptyset$. Mientras $|S(q^i)| < 2\kappa^7$:

      8.3.1 Ejecute **Sample**$(i, \{q^i\}, \lambda, \frac{e^{-5}}{N(q^i)})$ hasta que retorne $w \neq$ **fail**, pero a lo más $c(\kappa)$ veces

      8.3.2 Si $w =$ **fail**, entonces retorne 0

      8.3.3 Sea $S(q^i) = S(q^i) \cup \{w\}$

9. Retorne $N(F^n)$ como una estimación de $|L_n(\mathcal{A})|$

La complejidad del algoritmo

Definimos $c(\kappa) = \left\lceil \dfrac{2 + \ln(4) + 8\ln(\kappa)}{\ln(1 - e^{-9})^{-1}} \right\rceil$

La complejidad del algoritmo

Definimos $c(\kappa) = \left\lceil \dfrac{2 + \ln(4) + 8\ln(\kappa)}{\ln(1 - e^{-9})^{-1}} \right\rceil$

Dado que $c(\kappa) \in O(\ln(\kappa))$, el algoritmo funciona en tiempo $p(m, n, \frac{1}{\varepsilon})$, para un polinomio fijo p

La correctitud del algoritmo

Nos falta demostrar que la probabilidad de que el algoritmo de un resultado incorrecto es a lo más $\frac{1}{4}$

La correctitud del algoritmo

Nos falta demostrar que la probabilidad de que el algoritmo de un resultado incorrecto es a lo más $\frac{1}{4}$

Para esto necesitamos considerar el siguiente evento:

- $\mathcal{E}_{\textbf{fail}}(i, q^i, j)$: la llamada **Sample**$(i, \{q^i\}, \lambda, \frac{e^{-5}}{N(q^i)})$ falla $c(\kappa)$ veces en la construcción de la j-ésima muestra de $S(q^i)$, donde $j \in [1, 2\kappa^7]$

La correctitud del algoritmo

La probabilidad de que el algoritmo falle es:

$$\mathbf{Pr}\left(\bigvee_{i=1}^{n} \bigvee_{q \in Q^i} \bigvee_{j=1}^{2\kappa^7} \mathcal{E}_{\mathbf{fail}}(i, q^i, j) \vee \bigvee_{k=0}^{n} \neg\mathcal{E}(k)\right)$$

La correctitud del algoritmo

La probabilidad de que el algoritmo falle es:

$$\mathbf{Pr}\Big(\bigvee_{i=1}^{n}\bigvee_{q\in Q^i}\bigvee_{j=1}^{2\kappa^7}\mathcal{E}_{\textbf{fail}}(i,q^i,j)\vee\bigvee_{k=0}^{n}\neg\mathcal{E}(k)\Big) \;\leq$$

$$\sum_{i=1}^{n}\sum_{q\in Q^i}\sum_{j=1}^{2\kappa^7}\mathbf{Pr}(\mathcal{E}_{\textbf{fail}}(i,q^i,j))+1-\mathbf{Pr}\Big(\bigwedge_{k=0}^{n}\mathcal{E}(k)\Big)$$

Sabemos que:

$$\begin{aligned}
\mathbf{Pr}(\mathcal{E}_{\textbf{fail}}(i,q^i,j)) &\leq (1-e^{-9})^{c(\kappa)}\\
\mathbf{Pr}\Big(\bigwedge_{k=0}^{n}\mathcal{E}(k)\Big) &\geq 1-e^{-\kappa}
\end{aligned}$$

La correctitud del algoritmo

Por lo tanto:

$$\begin{aligned}
\mathbf{Pr}\left(\bigvee_{i=1}^{n}\bigvee_{q\in Q^i}\bigvee_{j=1}^{2\kappa^7}\mathcal{E}_{\textbf{fail}}(i,q^i,j)\vee\bigvee_{k=0}^{n}\neg\mathcal{E}(k)\right) &\leq \sum_{i=1}^{n}\sum_{q\in Q^i}\sum_{j=1}^{2\kappa^7}(1-e^{-9})^{c(\kappa)}+e^{-\kappa} \\
&\leq nm2\kappa^7(1-e^{-9})^{c(\kappa)}+e^{-2} \\
&\leq 2\kappa^8(1-e^{-9})^{c(\kappa)}+e^{-2}
\end{aligned}$$

La correctitud del algoritmo

Pero tenemos que:

$$\begin{aligned}
2\kappa^8(1-e^{-9})^{c(\kappa)} &= 2\kappa^8(1-e^{-9})^{\left\lceil \frac{2+\ln(4)+8\ln(\kappa)}{\ln(1-e^{-9})-1} \right\rceil} \\
&\leq 2\kappa^8(1-e^{-9})^{\frac{2+\ln(4)+8\ln(\kappa)}{\ln(1-e^{-9})-1}} \\
&= 2\kappa^8(1-e^{-9})^{\frac{\ln(4e^2\kappa^8)}{\ln(1-e^{-9})-1}} \\
&= 2\kappa^8(1-e^{-9})^{\frac{\ln(\frac{1}{4}e^{-2}\kappa^{-8})}{\ln(1-e^{-9})}} \\
&= 2\kappa^8(1-e^{-9})^{\log_{(1-e^{-9})}(\frac{1}{4}e^{-2}\kappa^{-8})} \\
&= 2\kappa^8\frac{1}{4}e^{-2}\kappa^{-8} \\
&= \frac{1}{2}e^{-2}
\end{aligned}$$

La correctitud del algoritmo

Concluimos que:

$$\mathbf{Pr}\left(\bigvee_{i=1}^{n}\bigvee_{q\in Q^i}\bigvee_{j=1}^{2\kappa^7}\mathcal{E}_{\textbf{fail}}(i,q^i,j)\vee\bigvee_{k=0}^{n}\neg\mathcal{E}(k)\right)\ \leq$$

$$2\kappa^8(1-e^{-9})^{c(\kappa)}+e^{-2}\ \leq\ \frac{1}{2}e^{-2}+e^{-2}\ =\ \frac{3}{2}e^{-1}\ <\ \frac{1}{4}$$

La complejidad exacta el algoritmo y una mejora

El algoritmo funciona en tiempo:

$$\widetilde{O}\left(\frac{m^{17}n^{17}}{\varepsilon^{14}}\right)$$

Un versión más eficiente del algoritmo fue presentado aquí
`https://arxiv.org/pdf/2312.13320`:

$$\widetilde{O}\left(\frac{m^{2}n^{10}+m^{3}n^{6}}{\varepsilon^{4}}\right)$$