



## Ejercicio 5 - Lista de Ejercicios 2

Demuestre que si  $\text{SAT} \in \text{BPP}$ , entonces  $\text{SAT} \in \text{RP}$ . Nótese que de esto se concluye que si  $\text{NP} \subseteq \text{BPP}$ , entonces  $\text{RP} = \text{NP}$ .

**Solución:** Supongamos que  $\text{SAT} \in \text{BPP}$ . Tomamos como instancia una formula  $\varphi$  con variables  $x_1, \dots, x_n$ .

Como asumimos  $\text{SAT} \in \text{BPP}$ , entonces existe un algoritmo en BPP para SAT  $\mathcal{A}$  con probabilidad de error  $\leq 1/4$ . Por el lema de amplificación, podemos tener un algoritmo  $\mathcal{A}' \in \text{BPP}$  con error  $\leq \left(\frac{3}{4}\right)^k$ , para un  $k \in \mathbb{N}$  conveniente.

Ahora veremos que a partir de este algoritmo  $\mathcal{A}'$  podemos crear un algoritmo  $\mathcal{B}$  para SAT en RP, es decir, un algoritmo que nunca acepta  $\varphi$  cuando  $\varphi \notin \text{SAT}$  y que cuando  $\varphi \in \text{SAT}$ ,  $\mathbb{P}(\mathcal{B} \text{ acepta } \varphi) \geq 3/4$ . Para un input  $\varphi$ ,  $\mathcal{B}$  funciona de la siguiente manera:

- Ejecuta  $\mathcal{A}'$  con input  $\varphi$ . Si  $\mathcal{A}'$  rechaza, entonces rechazo.<sup>1</sup>
- Si  $\mathcal{A}'$  acepta, entonces tengo que asegurarme que no voy a cometer un error, para eso hago lo siguiente:
  1. seteo  $x_1$  a falso y ejecuto  $\mathcal{A}'$  sobre  $\varphi_{(x_1=0)}$ . Si  $\mathcal{A}'$  acepta, continuo seteando  $x_2$  a falso y ejecutando  $\mathcal{A}'$  sobre  $\varphi_{(x_1=0, x_2=0)}$
  2. si  $\mathcal{A}'$  rechaza a  $\varphi_{(x_1=0)}$ , cambio la valuación y dejo seteada  $x_1 = 1$ .
  3. se continúan los dos pasos anteriores con los demás  $x_i$  hasta que tenemos una asignación para cada uno de ellos.
  4. finalmente (al llegar a  $x_n$ ) tenemos la asignación para todos los  $x_i$ , entonces revisamos que esa valuación satisfaga  $\varphi$ . Si la satisface, acepto, en otro caso, rechazo.

Ahora revisamos lo siguiente:

- $\mathcal{B}$  corre en tiempo polinomial.
- $\mathcal{B}$  llama al algoritmo  $\mathcal{A}'$  (que corre en tiempo polinomial), una cantidad lineal de veces (exactamente  $n + 1$  veces <sup>2</sup>). Además el input para  $\mathcal{A}'$  en cada llamada es de largo a

<sup>1</sup>En la sesión de ejercicios convenimos en que este paso no es necesario

<sup>2</sup> $n$  si no consideramos la primera llamada

lo más  $|\varphi|$  (*setear* variables solo puede disminuir el largo de la formula  $\varphi$ ).

■  $\mathcal{B} \in \text{RP}$ .

Primero notamos que  $\mathcal{B}$  no puede aceptar una formula  $\varphi$  que no sea satisfacible (para aceptar tiene que tener una valuación que satisfaga  $\varphi$ ).

Ahora supongamos  $\varphi$  satisfacible y calculemos la probabilidad de aceptar.

- Si  $\mathcal{A}'$  da el output correcto cada vez, entonces terminamos con una valuación que satisface  $\varphi$  y aceptamos.
- la probabilidad de que  $\mathcal{A}'$  se equivoque es  $\left(\frac{3}{4}\right)^k$ , entonces por cota de la unión la probabilidad de que se equivoque en al menos una ejecución es menor o igual a  $(n+1) \cdot \left(\frac{3}{4}\right)^k$ <sup>3</sup>. Escogemos un  $k$  suficientemente grande para que:

$$\begin{aligned} (n+1) \cdot \left(\frac{3}{4}\right)^k &\leq k \left(\frac{3}{4}\right)^k \\ &\leq \frac{1}{4} \end{aligned}$$

(esto funciona para  $k \geq \max\{14, n+1\}$ )<sup>4</sup>.

Por lo tanto la probabilidad de que lo anterior no pase es  $\geq \frac{3}{4}$  (esta es la probabilidad de que siendo  $\varphi$  satisfacible,  $\mathcal{B}$  acepte. )

Finalmente analizamos el ultimo comentario del enunciado. Ya sabemos que  $\text{RP} \subseteq \text{NP}$ , de modo que solo debemos mostrar que  $\text{NP} \subseteq \text{RP}$ .

Asumimos que  $\text{NP} \subseteq \text{BPP}$  (esto implica que  $\text{SAT} \in \text{BPP}$ ). Por el argumento anterior, entonces  $\text{SAT} \in \text{RP}$ . Finalmente como  $\text{RP}$  es cerrado bajo reducciones y  $\text{SAT}$  es  $\text{NP}$ -completo (todo lenguaje en  $\text{NP}$  es reducible a  $\text{SAT}$ ), entonces necesariamente  $\text{NP} \subseteq \text{RP}$ , de lo que se concluye que  $\text{RP} = \text{NP}$ .

---

<sup>3</sup> $(n) \cdot \left(\frac{3}{4}\right)^k$   
<sup>4</sup> $k \geq \max\{14, n\}$