Algorithm 5.13: ORACLE RSA DECRYPTION(n, b, y)

external HALF
$k \leftarrow \lfloor \log_2 n \rfloor$
for $i \leftarrow 0$ **to** k
do $\begin{cases} h_i \leftarrow \text{HALF}(n, b, y) \\ y \leftarrow (y \times 2^b) \bmod n \end{cases}$
$lo \leftarrow 0$
$hi \leftarrow n$
for $i \leftarrow 0$ **to** k
do $\begin{cases} mid \leftarrow (hi + lo)/2 \\ \textbf{if } h_i = 1 \\ \quad \textbf{then } lo \leftarrow mid \\ \quad \textbf{else } hi \leftarrow mid \end{cases}$
return $(\lfloor hi \rfloor)$

Bitowe bezpieczeństwo RSA

Klucz publiczny Alicji: $n = p \cdot q$, e

Klucz prywatny:

$d : d \cdot e \equiv NWW(p-1, q-1) = 1 \pmod{}$

Z zał RSA wynika, że jeśli B prześle do A szyfrogram $m^e \bmod n$ to Oskar nie jest w stanie odszyfrować m,

Pytanie: Czy Oskar nie jest w stanie wydobyć cz. informacji o m?

$\left(\frac{m}{n}\right)$ – symbol Jacobiego

Jeżeli $2 \nmid e$, to $\left(\frac{m^e}{n}\right) = \left(\frac{m}{n}\right)$

Inne pytanie: Czy Oskar może wydobyć z $c = m^e$ ostatni bit m?

$LAST(m^e) = m \bmod 2$

Można też spytać o pierwszy bit m

$$HALF(m^e) = \begin{cases} 1 & m > n/2 \\ 0 & m < n/2 \end{cases}$$

Tw: Dysponując f-cją HALF wyliczającą 1-szy bit można wyliczyć m

Odszyfrowywanie m $(c = m^e)$

$k \leftarrow \lceil \log_2 n \rceil$

for $i \leftarrow 0$ to k do

$h_i \leftarrow HALF(c \cdot 2^{ei})$

$lo \leftarrow 0$; $hi \leftarrow n$

for $i \leftarrow 0$ to k do

$mid \leftarrow (hi + lo)/2$

if $h_i = 0$

then $hi \leftarrow mid$

else $lo \leftarrow mid$

na końcu w lo jest jedyna wartość $m < n$

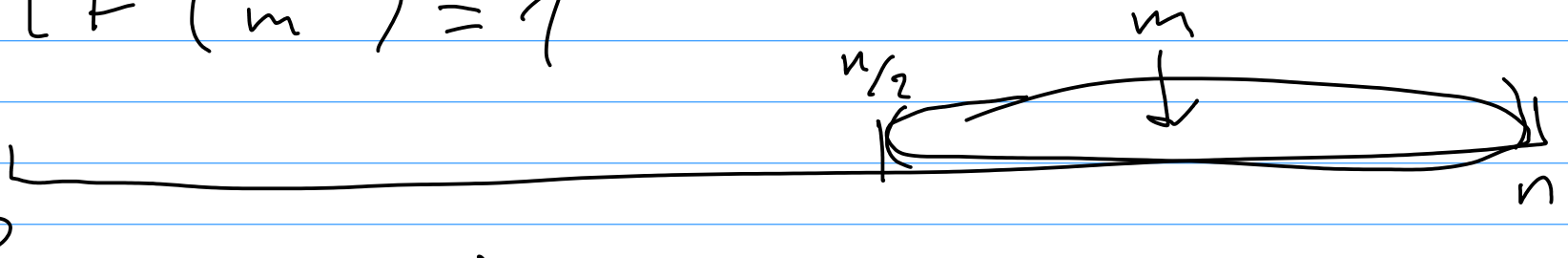
Dlaczego to chciała?

$h_0 = 1 \iff m \in (n/2, n)$

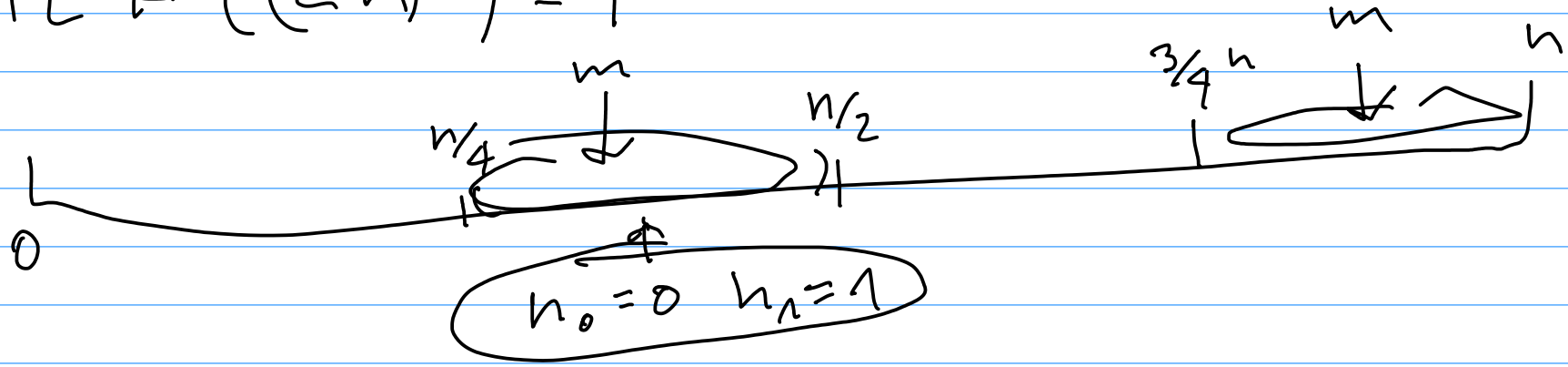
$h_1 = 1 \iff m \in (n/4, n/2) \cup (\frac{3}{4}n, n)$

$h_2 = 1 \iff m \in (\frac{n}{8}, \frac{n}{4}) \cup (\frac{3n}{8}, \frac{n}{2}) \cup (\frac{5n}{8}, \frac{3n}{4}) \cup (\frac{7n}{8}, n)$

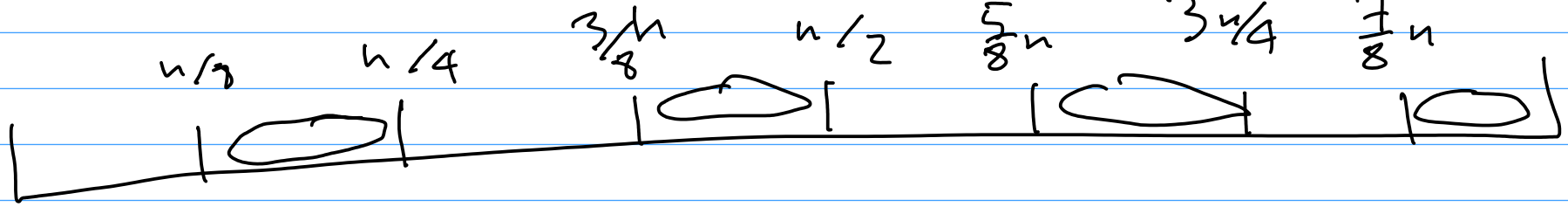
$h_0 = HALF(m^e) = 1$



$h_1 = HALF((2m)^e) = 1$



$$h_2 = HALF((2^2 m)^e) = 1$$



Tu: Jeśli dysponujemy funkcją HALF efektywnie wyliczającą pierwszy bit m, to potrafimy odszyfrować m.

Semantyczne bezpieczeństwo algorytmu Szyfrującego Elgamala

Klucz publiczny Alicji $p = 2q+1$,
$\alpha \in Z_p^*$, $ord(\alpha) = q$, $\beta = \alpha^x$

Klucz prywatny Alicji: x

Bob szyfruje m: $c = (\alpha^r, \beta^r m)$
r - losowe w Z_q

Szyfr jest bezpieczny gdy atakujący nie jest w stanie wygrać następującej gry

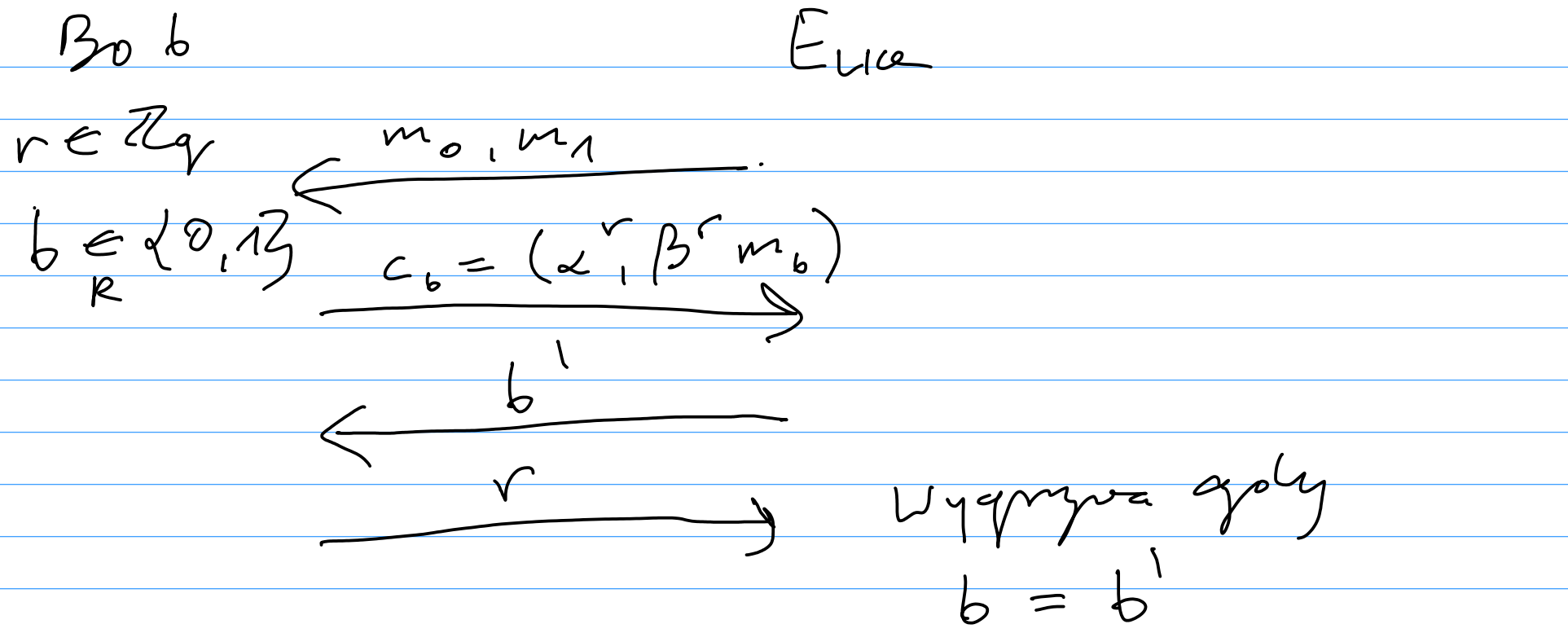
Atakujący wybiera dwie pary Alicji i generuje wiadomości $[m_0, m_1]$

rozkłady modułowe

Atakujący wysyła te wiadomości do Boba

Bob wybiera losową wiadomość m_b i oblicza szyfrogram $c_b = (\alpha^r, \beta^r m_b)^n$

Atakujący wygrywa gdy prawidłowo odgadnie b, czyli która wiadomość została zaszyfrowana



Załóżmy, że Eva potrafi wygrać

$2 \quad pr > \frac{1}{2} + \varepsilon$

Decyzyjny problem Diffie - Hellmana

Ewa dostaje

$$p, g, g^a, g^b, g^{ab} \vee g^r \text{ gdzie } r \in_R \mathbb{Z}_q$$

Ewa wygrywa gdy odgadnie, czy ostatnią wartością jest g^{ab} czy g^r

DDH — decyzyjny problem D-H

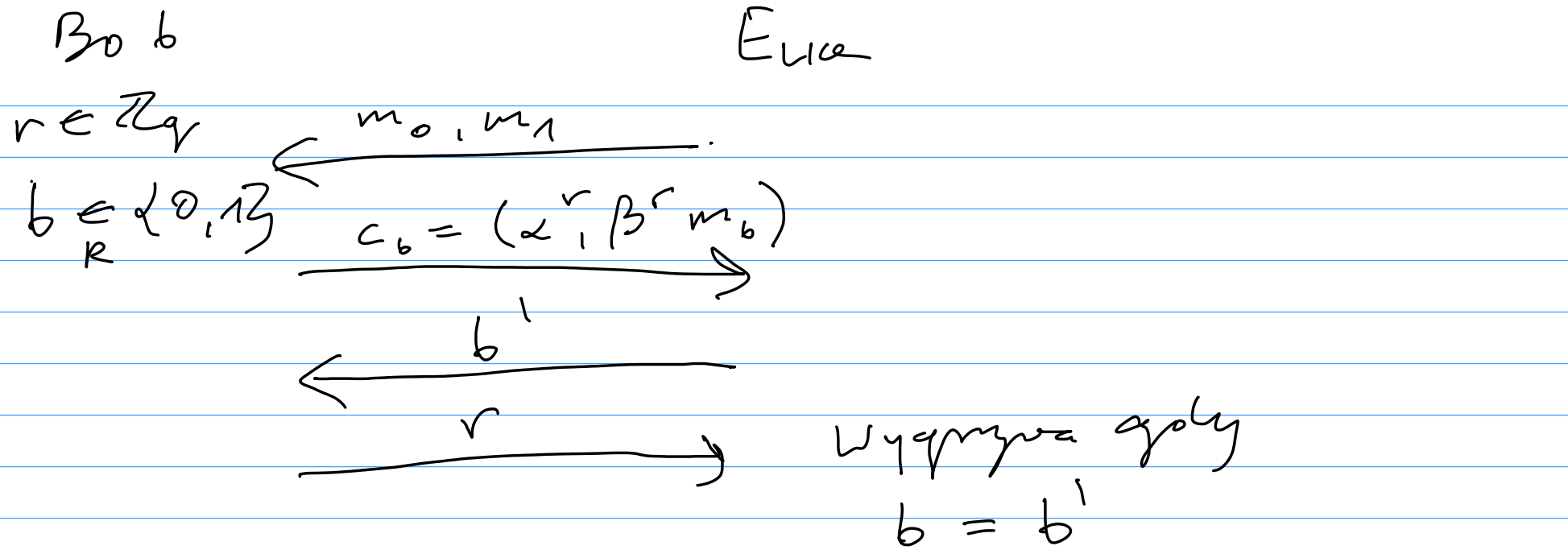
CDH — obliczeniowy pr D-H
(oblicz g^{ab})

Tw Tezeli Eva potrafi wygrać
grę w odróżnianie szyfrogramów
m_0 i m_1 z $pr > \frac{1}{2} + \varepsilon$, to
potrafi też wygrać grę w
odróżnianie g^{ab} i g^r z
$pr > \frac{1}{2} + \varepsilon$ czyli rozwiązać problem
DDH

Dowód

Zn T, że Eva dostała do rozw
problem DDH czyli $p, \alpha^x, \alpha^r, \alpha^{xr}$ v α^R
R losowe
z Z_q

$S = \alpha^{xr}$ v α^R



Ewa losuje $b \in \{0,1\}$ i oblicza

$$c = (\alpha^r, s m_b)$$

Ewa zgaduje b'

Jeżeli $b = b'$ to Ewa stwierdza, że $s = \alpha^{xr}$ vpp $s = \alpha^k$

Możliwość 1 : $s = \alpha^{xr}$

Ewa wygrywa gdy w sem bop

Algorytm A poprawia $> \frac{1}{2} + \varepsilon$

$Pr(b = b') > \frac{1}{2} + \varepsilon$

Ewa stwierdzi że $s = \alpha^{xr}$

A poprawia $> \frac{1}{2} + \varepsilon$

Możliwość 2 : $s = \alpha^{R}$

Każda wartość $c = (\alpha^{r}, \alpha^{R'})$ jest

równie prawdopodobna jako

szyfrogram $(\alpha^{r}, m_b \alpha^{R})$

$\Rightarrow Pr(b = b') = 1/2$

Eva stvi cvdki , ze $S = \alpha^{xr}$

$2pr = \frac{1}{2}$