- A small exercise, to understand what is the M_w (and x) in the proof.
  - Consider M to be the TM that accepts all strings of the form 0^n1^n.
  - Take $w_1 = 0011$, $w_2 = 001$.
  - Write/understand what would be
    (i) M_w_1 (ii) M_w_2.

What is $L(M\_w_1)$? What is $L(M\_w_2)$?

Another language:

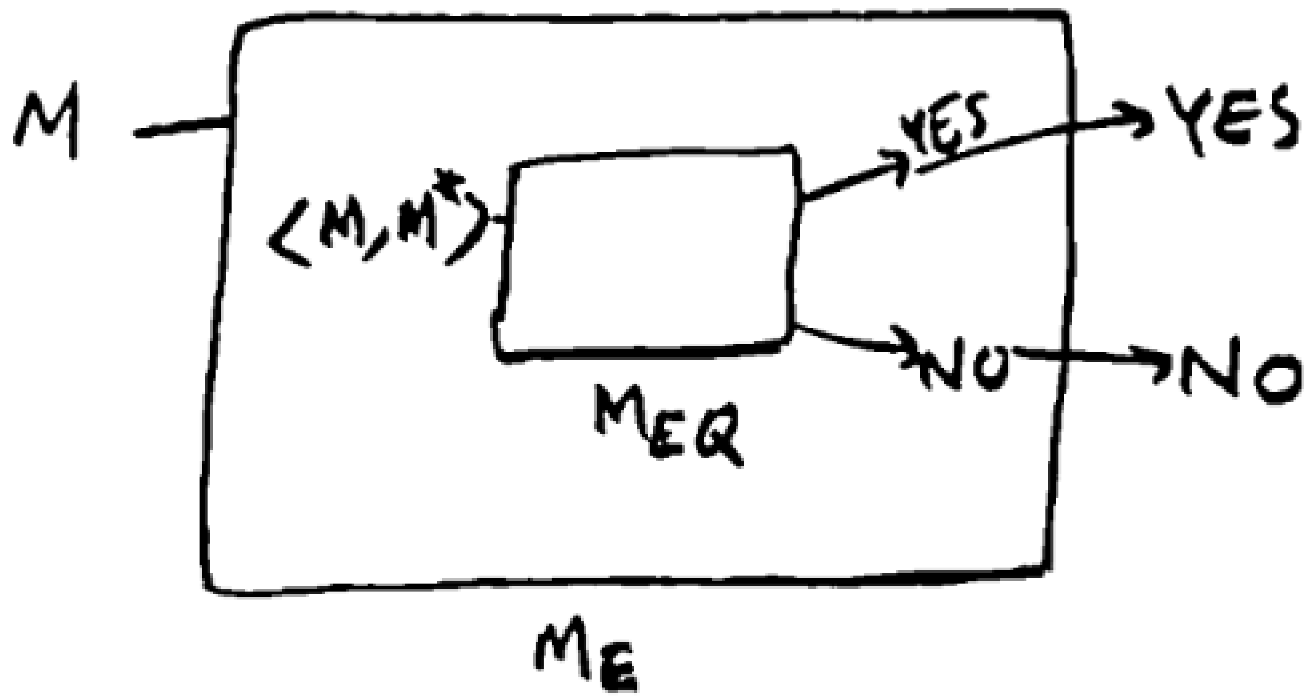
$EQ_{TM} : \{ \langle M_1, M_2 \rangle \mid L(M_1) = L(M_2),$

M_1 and M_2 are TMs$\}$.

- Claim: EQ_{TM} is undecidable.

Proof idea: E_{TM} is decidable

if EQ_{TM} is decidable.



- Define M^* as a TM that rejects all strings, and call M_{EQ} with $\langle M, M^* \rangle$.

- $REGULAR_{TM}$: Given a turing machine M, is $L(M)$ a regular language?

  - This is also undecidable.

- How to prove? Slightly tricky.
- We can design M_{TM} (for L_{TM}) using a machine $M_{REGULAR}$ (if it existed).

Algorithm for M_{TM} :

On input <M, w>:

1. Construct M_w as follows:

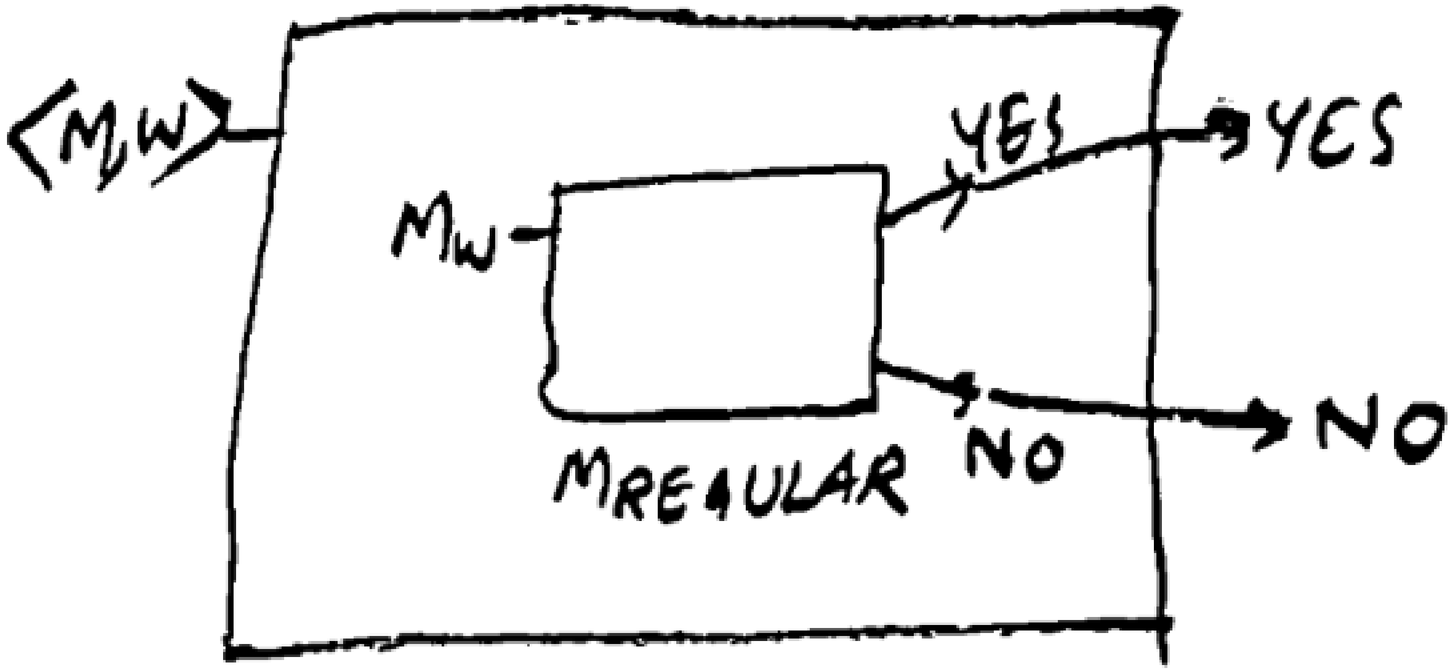
   M_w's algorithm: On input x,

   (i) If x is of the form 0^n1^n, accept.

   (ii) Else run M on input w,
   accept if M accepts w.

2. Run $M_{REGULAR}$ with M_w as input.
   YES → YES , NO → NO.

(SUDEEP)



M_{TM}

- Think: why this works?