A less ~~powerful~~ complicated (apparently less powerful) TM:

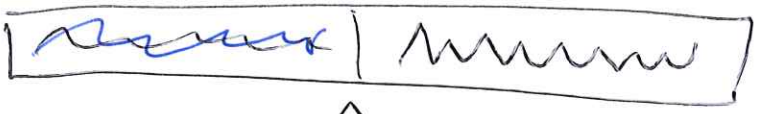
My tape is finite at one end - I'm not allowed to move off tape to the left. (TMs don't crash - all exceptions are handled by halting and saying "No")

Why are these one-ended-TMs equally powerful as standard-TMs?

(1) How do we make a one-ended TM do the same thing as a standard TM?

a) ~~we can~~ if we make a 2-tape-one-ended-TM, we can make a one-ended-TM from that.

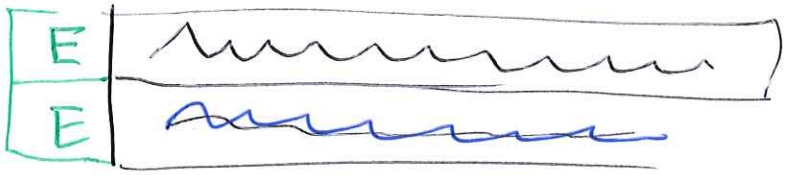
b) Now we think of "bending the tape around the end" - i.e. if we had



-- on our standard-TM

orig beginning

We make our 2-tape one-ended-TM have



~~Las a~~

We have our state keep track of whether we're on top tape or bottom tape.

special E markers to handle transition from working on top to working on bottom.

We make lots of new transitions to essentially mimic the old transitions in this new setting.

---

(2) How do we make a standard-TM that does the same thing as a given one-ended-TM?

- Start by marking the supposed left end with a special symbol that triggers a halt on any state.

Moral: Think about how to store the data - if you can, then you can probably make the transitions work.

Next example:

Multi-tape, multi-head machine.

- I have 5 tapes, and one head for each tape. Depending on what I read under each head, ~~each he~~ I make a new mark on each tape, then the 5 heads can move left or right indep. from each other.

i.e. - my transition function looks like

$$\delta : Q \times \Gamma \times \Gamma \times \Gamma \times \Gamma \times \Gamma$$

$$\longrightarrow Q \times \Gamma \times \Gamma \times \Gamma \times \Gamma \times \Gamma$$

$$\times \{L,R\} \times \{L,R\} \times \cdots \times \{L,R\}$$

I can make a standard-TM emulate a 5-tape, 5-head TM?

I can just reduce to a 10-tape, 1-head TM. I know how to make a standard-TM emulate that.

~~tape 1: tape 1 of 5~~

10-tape 1-head	5-tape 5-head
1	what's on tape 1
2	where the head is on tape 1 (a bunch of 0's, except for a 1 showing where the head is)
3	what's on tape 2
4	where head 2 is
⋮	⋮

Now I have to think about whether I can indeed make a standard-TM maintain this data

- Yes - ~~I~~ every time I need to ~~do some~~ simulate a single move;

1) search for the 1 on tape 2 (marking the space I've searched w/ x's or something)

I am searching by going back and forth - the marks tell me when to turn around.

2) when I find the 1, remember the symbol on tape 1, then erase all the marks on tape 2.

3) do the same w/ tapes 3 + 4, etc.

4) when I've read what's on each tape - decide what to do based on emulated-TM's transitions remember all that

5) search for the 1 on tape 2 again.

6) modify tape 1, move the marker on tape 2, erase search marks on 2.

7) do this to all the other tapes.

Careful check - I can only have finitely many states, so the pieces of info I'm remembering as states can only have finitely many possibilities.

- so I ~~could~~ cannot make step 5 unneccessary by remembering how far away the mark on tape 2 was.