Multi-Tape Turing Machines, Nondeterministic Turing Machines, and the Church-Turing Thesis

Supplement for Preparatory Viewing

Example of a Multi-Tape Turing Machine

Let $\Sigma = \{\texttt{a}, \texttt{b}\}$ and let

$$L = \{\texttt{a}^n\texttt{b}^n \mid n \in \mathbb{N}\}.$$

One Way To Decide Whether $\omega \in L$ — Using Two Tapes:

1. ***Accept*** if $\omega = \lambda$. Otherwise sweep right, copying initial copies of "a" onto the second type (marking the first so it can be found later) — ***rejecting*** if ω starts with "b".
2. ***Reject*** if the next symbol seen, on the first tape, is "$\sqcup$". Otherwise sweep ***right*** over copies of "b" on the first tape while matching with copies of "a" seen while seeping back to the ***left*** on the second tape.
3. If another "a" is seen on the first tape then ***reject***. Otherwise, ***accept*** if "$\sqcup$" is seen on the first tape at the same time as the leftmost call is detected on the second tape, and ***reject*** otherwise.

This can be implemented using a 2-tape Turing machine

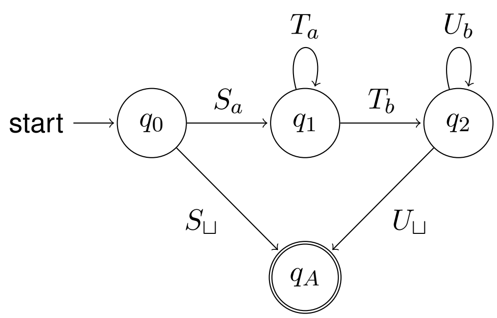
$$M = (Q, \Sigma, \Gamma, \delta, q_0, q_{\text{accept}}, q_{\text{reject}})$$

where

$$Q = \{q_0, q_1, q_2, q_{\text{accept}}, q_{\text{reject}}\},$$

$$\Gamma = \{\texttt{a}, \texttt{b}, \dot{\texttt{a}}, \sqcup\},$$

and transitions are as shown in Figure 1 on page 2. In this figure, the accepting state is shown as "q_A" instead of "q_{accept}", and transitions to the rejecting state (and the rejecting state, itself), are not shown — in order to keep the picture as simple as possible. Each missing transition should move to the rejecting state without changing the symbols visible on tapes or moving tape heads.



Transitions:

S_a: $\texttt{a}, \sqcup / \dot{\texttt{a}}, \texttt{R}, \dot{\texttt{a}}, \texttt{R}$
$S_\sqcup$: $\sqcup, \sqcup / \sqcup, \texttt{L}, \sqcup, \texttt{L}$

T_a: $\texttt{a}, \sqcup / \texttt{a}, \texttt{R}, \texttt{a}, \texttt{R}$
T_b: $\texttt{b}, \sqcup / \texttt{b}, \texttt{R}, \sqcup, \texttt{L}$

U_b: $\texttt{b}, \texttt{a} / \texttt{b}, \texttt{R}, \texttt{a}, \texttt{L}$
$U_\sqcup$: $\sqcup, \dot{\texttt{a}} / \sqcup, \texttt{L}, \dot{\texttt{a}}, \texttt{L}$

Figure 1: Multi-Tape Turing Machine Deciding the Language L

Simulation of a Multi-Tape Turing Machine

The simulation of a k-tape Turing machine, by a standard (single-tape) Turing machine, uses a greatly expanded tape alphabet — which includes symbols allowing a part of the tape to be thought of as including $2k$ ***tracks***, allowing the positions of k tape heads, and contents of cells in k tapes, to be represented with one tape instead. For example, a configuration of a 3-tape Turing machine, whose tapes look like this,

			↓			
?	a	b	a	⊔	⊔	...

↓						
c	d	a	⊔	⊔	⊔	...

				↓		
?	a	⊔	a	⊔	⊔	...

would be representing with a single tape that looks like this:

<table>
<tr><td>1.</td><td rowspan="6">$</td><td>⊔</td><td>⊔</td><td>⊔</td><td>↓</td><td>⊔</td><td rowspan="6">⊔</td><td rowspan="6">...</td></tr>
<tr><td>2.</td><td>?</td><td>a</td><td>b</td><td>a</td><td>⊔</td></tr>
<tr><td>3.</td><td>↓</td><td>⊔</td><td>⊔</td><td>⊔</td><td>⊔</td></tr>
<tr><td>4.</td><td>c</td><td>d</td><td>a</td><td>⊔</td><td>⊔</td></tr>
<tr><td>5.</td><td>⊔</td><td>⊔</td><td>⊔</td><td>⊔</td><td>↓</td></tr>
<tr><td>6.</td><td>?</td><td>a</td><td>⊔</td><td>a</td><td>⊔</td></tr>
</table>