Griffith University

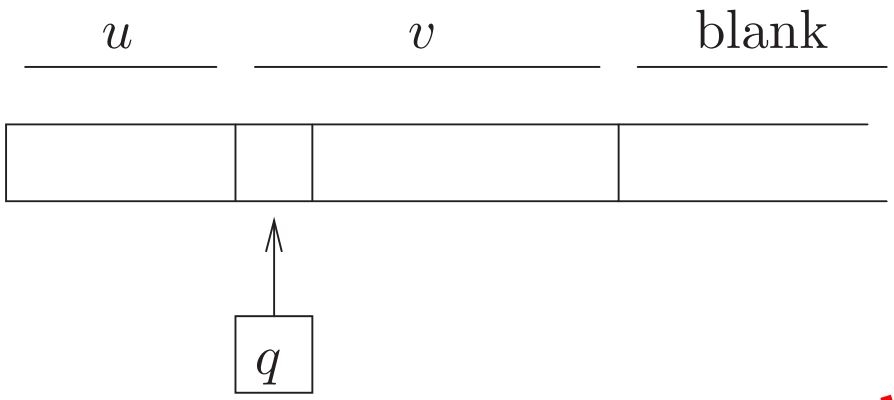
3515ICT Theory of Computation

Turing Machines

(Based loosely on slides by Harald Søndergaard of The University of Melbourne)

Turing Machine Configurations

We write uqv for this configuration:



On input aba, the example machine goes through these configurations:

$$
\begin{array}{ll}
 & \varepsilon q_0 aba \qquad \text{(or just } q_0 aba\text{)} \\
\vdash & a q_1 ba \\
\vdash & ab q_0 a \\
\vdash & aba q_1 \sqcup \\
\vdash & aba \sqcup q_r
\end{array}
$$

We read the relation '$\vdash$' as 'yields'.

Turing Machines and Languages

The set of strings accepted by a Turing machine M is the language *recognised by* M, $L(M)$.

A language A is *Turing-recognisable* or *computably enumerable (c.e.)* or *recursively enumerable (r.e.)* (or *semi-decidable*) iff $A = L(M)$ for some Turing machine M.

Three behaviours are possible for M on input w: M may accept w, reject w, or fail to halt.

If language A is recognised by a Turing machine M that *halts on all inputs*, we say that M *decides* A.

A language A is *Turing-decidable*, or just *decidable*, or *computable*, or (even) *recursive* iff some Turing machine decides A.

(The difference between Turing-recognisable and (Turing-)decidable is *very* important.)

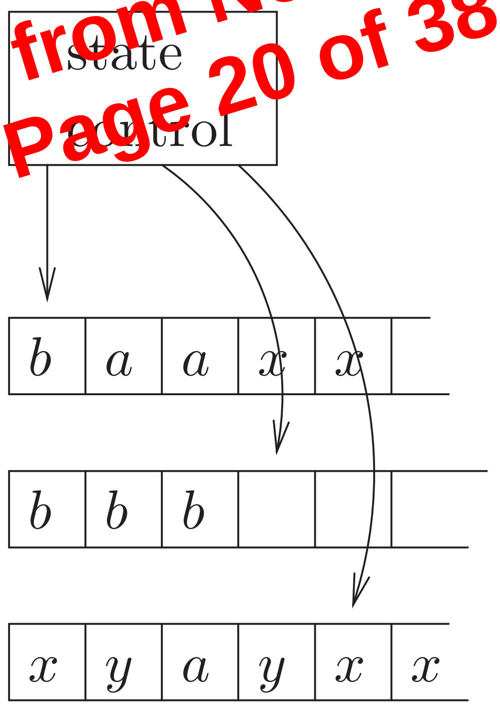
Multitape Machines

A *multitape Turing machine* has k tapes. Input is on tape 1, the other tapes are initially blank. The transition function now has the form

$$\delta : Q \times \Gamma^k \to Q \times \Gamma^k \times \{L, R\}^k$$

It specifies how the k tape heads behave when the machine is in state q_i, reading $a_1, \ldots, a_k$:

$$\delta(q_i, a_1, \ldots, a_k) = (q_j, (b_1, \ldots, b_k), (d_1, \ldots, d_k))$$



Nondet. Turing Machines (cont.)

I.e., adding nondeterminism to Turing machines does not allow them to recognise any more languages.

A nondeterministic Turing machine that halts on every branch of its computation on all inputs is called a *decider*.

Corollary. A language is decidable iff some nondeterministic Turing machine decides it.

Enumerators (cont.)

Corollary. A language L is Turing-decidable iff some enumerator generates each string of L exactly once, in order of nondecreasing length.

Proof. Exercise for the reader. (Actually, a simplification of the previous proof.)