Universal Turing Machines

KR Chowdhary
Professor & Head
Email: kr.chowdhary@acm.org

Department of Computer Science and Engineering
MBM Engineering College, Jodhpur

October 8, 2010

TM Simulations

- A 3-tape TM can be simulated by a standard Turing machine

TM Simulations

- A 3-tape TM can be simulated by a standard Turing machine
- a 2D-Tm can be simutaed by a standard TM

TM Simulations

- A 3-tape TM can be simulated by a standard Turing machine
- a 2D-Tm can be simutaed by a standard TM
- A NDTM can be simulated by a standatd TM

TM Simulations

- A 3-tape TM can be simulated by a standard Turing machine
- a 2D-Tm can be simutaed by a standard TM
- A NDTM can be simulated by a standatd TM
- Can a TM be simulated by a TM?- YES

TM Simulations

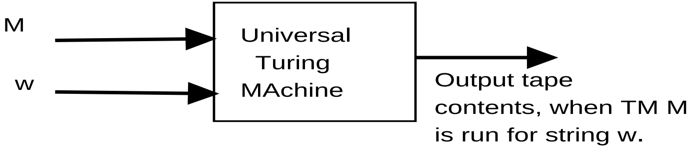
- A 3-tape TM can be simulated by a standard Turing machine
- a 2D-Tm can be simutaed by a standard TM
- A NDTM can be simulated by a standatd TM
- Can a TM be simulated by a TM?- YES
- Can one Turing machine simulate every TM?

Universal TM

Input $= M, w$, Output $=$ the output produced when M runs when input is w. This machine is Universal Turing machine (UTM) U.

Input $= M, w$, Output = the output produced when M runs when input is w. This machine is Universal Turing machine (UTM) U.

- A UTM can be designed to simulate the computations of an arbitrary TM M. To do so, input to UTM must contain representation of both - machine M and string w processed by M.



TM Simulations..

Let M that accepts by halting.

TM Simulations..

Let M that accepts by halting.

We can present a UTM U for this as follows:

TM Simulations..

Let M that accepts by halting.

We can present a UTM U for this as follows:

- Input $= R(M)w$, where $R(M)$ is representation of M

TM Simulations..

Let M that accepts by halting.

We can present a UTM U for this as follows:

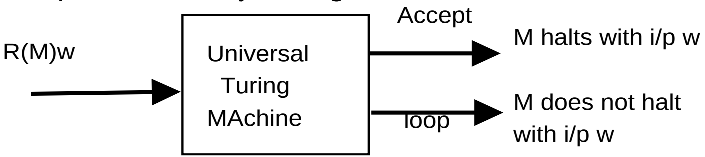
- Input $= R(M)w$, where $R(M)$ is representation of M
- Output1: Accept (indicates that M halts with input w), output2: loops, i.e., M does not halt with input w, i.e. computation of does not terminate

TM Simulations..

Let M that accepts by halting.

We can present a UTM U for this as follows:

- Input $= R(M)w$, where $R(M)$ is representation of M
- Output1: Accept (indicates that M halts with input w), output2: loops, i.e., M does not halt with input w, i.e. computation of does not terminate
- The machine U is called universal TM, as computation of any Turing machine can be simulated by U.



Step1: design a string representation of a TM M

Because of the ability to encode arbitrary symbols as strings over $\{0, 1\}$, we consider Turing machine with inputs $\{0, 1\}$ and tape symbols $\Gamma = \{0, 1, \#\}$

Step1: design a string representation of a TM M

Because of the ability to encode arbitrary symbols as strings over $\{0, 1\}$, we consider Turing machine with inputs $\{0, 1\}$ and tape symbols $\Gamma = \{0, 1, \#\}$

- The states of M are assumed to be $\{q_0, q_1, \ldots, q_n\}$

Step1: design a string representation of a TM M

Because of the ability to encode arbitrary symbols as strings over $\{0, 1\}$, we consider Turing machine with inputs $\{0, 1\}$ and tape symbols $\Gamma = \{0, 1, \#\}$

- The states of M are assumed to be $\{q_0, q_1, \ldots, q_n\}$
- TM M is defined by its transition function:

Step1: design a string representation of a TM M

Because of the ability to encode arbitrary symbols as strings over $\{0,1\}$, we consider Turing machine with inputs $\{0,1\}$ and tape symbols $\Gamma = \{0,1,\#\}$

- The states of M are assumed to be $\{q_0, q_1, \ldots, q_n\}$
- TM M is defined by its transition function:

  $\delta(q_i, a) = (q_j, b, d); \;\; q_i, q_j \in Q; \;\; a, b \in \Gamma; \;\; d \in \{L, R\}$

Encoding of elements of M

Symbol	Encoding
0	1
1	11
#	111
q_0	1
q_1	11
...	...
q_n	1^{n+1}
L	1
R	11

Encoding of elements of M

Symbol	Encoding
0	1
1	11
#	111
q_0	1
q_1	11
...	...
q_n	1^{n+1}
L	1
R	11

Encoding of elements of M

Symbol	Encoding
0	1
1	11
#	111
q_0	1
q_1	11
...	...
q_n	1^{n+1}
L	1
R	11

- let $en(z)$ denote the encoding of z. Hence, Transition $\delta(q_i, a) = (q_j, b, d)$ is encoded by string $en(q_i)0en(a)0en(q_j)0en(b)0en(d)$. *0* separates the components of δ.

Encoding of elements of M..

- A representation of machine M is constructed from encoded transitions. Two consecutive 0s separate transitions. Beginning and end of representation are defined by three 0s.

Encoding of elements of M..

- A representation of machine M is constructed from encoded transitions. Two consecutive 0s separate transitions. Beginning and end of representation are defined by three 0s.

Encoding of elements of M..

- A representation of machine M is constructed from encoded transitions. Two consecutive 0s separate transitions. Beginning and end of representation are defined by three 0s.

Consider the Transitions:

Transition	Encoding
$\delta(q_0, \#) = (q_1, \#, R)$	101110110111011
$\delta(q_1, 0) = (q_0, 0, L)$	1101010101
$\delta(q_1, 1) = (q_2, 1, R)$	110110111011011
$\delta(q_2, 1) = (q_0, 1, L)$	1110110101101

- The machine M is represented by string: 000101110110111011

Encoding of elements of M..

- A representation of machine M is constructed from encoded transitions. Two consecutive 0s separate transitions. Beginning and end of representation are defined by three 0s.

  Consider the Transitions:

Transition	Encoding
$\delta(q_0, \#) = (q_1, \#, R)$	101110110111011
$\delta(q_1, 0) = (q_0, 0, L)$	1101010101
$\delta(q_1, 1) = (q_2, 1, R)$	110110111011011
$\delta(q_2, 1) = (q_0, 1, L)$	1110110101101

- The machine M is represented by string:
  000101110110111011
  00110101010100110110111011011001110110101101000

Simulation of M on Universal TM U

Verification of representation of M:

Simulation of M on Universal TM U

Verification of representation of M:

- TM can be constructed to check whether an arbitrary string $u \in \{0,1\}^*$ is encoding of deterministic TM M. Computations examines whether 000 is prefix, followed by finite sequences of encoded transitions are separated by 00s, then finally 000. (This is not the w to be recognized !!).

Simulation of M on Universal TM U

Verification of representation of M:

- TM can be constructed to check whether an arbitrary string $u \in \{0,1\}^*$ is encoding of deterministic TM M. Computations examines whether 000 is prefix, followed by finite sequences of encoded transitions are separated by 00s, then finally 000. (This is not the w to be recognized !!).
- M is deterministic if $Q \times \Gamma$ in every encoded transition is unique.

Simulation of M on Universal TM U

Verification of representation of M:

- TM can be constructed to check whether an arbitrary string $u \in \{0,1\}^*$ is encoding of deterministic TM M. Computations examines whether 000 is prefix, followed by finite sequences of encoded transitions are separated by 00s, then finally 000. (This is not the w to be recognized !!).
- M is deterministic if $Q \times \Gamma$ in every encoded transition is unique.

3-tape Deterministic TM U:

Simulation of M on Universal TM U

Verification of representation of M:

- TM can be constructed to check whether an arbitrary string $u \in \{0,1\}^*$ is encoding of deterministic TM M. Computations examines whether 000 is prefix, followed by finite sequences of encoded transitions are separated by 00s, then finally 000. (This is not the w to be recognized !!).
- M is deterministic if $Q \times \Gamma$ in every encoded transition is unique.

3-tape Deterministic TM U:

- Tape-1 holds $R(M)w$. Tape-3 simulates computations of of M for input w. Tape-2 as working tape.

Simulation of TM M on U

1. If input u is not of the form $R(M)w$ for deterministic TM M and string w on tape-1, the U moves to right forever.

Simulation of TM M on U

1. If input u is not of the form $R(M)w$ for deterministic TM M and string w on tape-1, the U moves to right forever.
2. w is written on tape-3 at left most square, with tape head at position 1. $\Rightarrow$ tape-3 is initial configuration of M with input w.

Simulation of TM M on U

1. If input u is not of the form $R(M)w$ for deterministic TM M and string w on tape-1, the U moves to right forever.
2. w is written on tape-3 at left most square, with tape head at position 1. $\Rightarrow$ tape-3 is initial configuration of M with input w.
3. Encoding of q_0,i.e., 1 is written tape-2.

Simulation of TM M on U

1. If input u is not of the form $R(M)w$ for deterministic TM M and string w on tape-1, the U moves to right forever.
2. w is written on tape-3 at left most square, with tape head at position 1. $\Rightarrow$ tape-3 is initial configuration of M with input w.
3. Encoding of q_0,i.e., 1 is written tape-2.
4. Transition of M is simulated on tape-3. The transition is determined by symbol scanned on tape-3 and state encoded on tape-2. Let these are a and q_i.

Simulation of TM M on U

1. If input u is not of the form $R(M)w$ for deterministic TM M and string w on tape-1, the U moves to right forever.
2. w is written on tape-3 at left most square, with tape head at position 1. $\Rightarrow$ tape-3 is initial configuration of M with input w.
3. Encoding of q_0,i.e., 1 is written tape-2.
4. Transition of M is simulated on tape-3. The transition is determined by symbol scanned on tape-3 and state encoded on tape-2. Let these are a and q_i.

   a. Tape-1 is scanned for a and q_i as first two components of a transition. If not found, U halts by accepting input.

Simulation of TM M on U

1. If input u is not of the form $R(M)w$ for deterministic TM M and string w on tape-1, the U moves to right forever.
2. w is written on tape-3 at left most square, with tape head at position 1. $\Rightarrow$ tape-3 is initial configuration of M with input w.
3. Encoding of q_0,i.e., 1 is written tape-2.
4. Transition of M is simulated on tape-3. The transition is determined by symbol scanned on tape-3 and state encoded on tape-2. Let these are a and q_i.

a Tape-1 is scanned for a and q_i as first two components of a transition. If not found, U halts by accepting input.

b If tape-1 has encoded information for above, then

Simulation of TM M on U

1. If input u is not of the form $R(M)w$ for deterministic TM M and string w on tape-1, the U moves to right forever.
2. w is written on tape-3 at left most square, with tape head at position 1. $\Rightarrow$ tape-3 is initial configuration of M with input w.
3. Encoding of q_0,i.e., 1 is written tape-2.
4. Transition of M is simulated on tape-3. The transition is determined by symbol scanned on tape-3 and state encoded on tape-2. Let these are a and q_i.

   a Tape-1 is scanned for a and q_i as first two components of a transition. If not found, U halts by accepting input.

   b If tape-1 has encoded information for above, then

   i q_i replaced by q_j on tape-2.

Simulation of TM M on U

1. If input u is not of the form $R(M)w$ for deterministic TM M and string w on tape-1, the U moves to right forever.
2. w is written on tape-3 at left most square, with tape head at position 1. $\Rightarrow$ tape-3 is initial configuration of M with input w.
3. Encoding of q_0,i.e., 1 is written tape-2.
4. Transition of M is simulated on tape-3. The transition is determined by symbol scanned on tape-3 and state encoded on tape-2. Let these are a and q_i.

a. Tape-1 is scanned for a and q_i as first two components of a transition. If not found, U halts by accepting input.

b. If tape-1 has encoded information for above, then

i. q_i replaced by q_j on tape-2.

ii. b is written on tape 3, and tape head on tape-3 is moved for direction given in d.

Simulation of TM M on U

1. If input u is not of the form $R(M)w$ for deterministic TM M and string w on tape-1, the U moves to right forever.
2. w is written on tape-3 at left most square, with tape head at position 1. $\Rightarrow$ tape-3 is initial configuration of M with input w.
3. Encoding of q_0,i.e., 1 is written tape-2.
4. Transition of M is simulated on tape-3. The transition is determined by symbol scanned on tape-3 and state encoded on tape-2. Let these are a and q_i.
   - a Tape-1 is scanned for a and q_i as first two components of a transition. If not found, U halts by accepting input.
   - b If tape-1 has encoded information for above, then
     - i q_i replaced by q_j on tape-2.
     - ii b is written on tape 3, and tape head on tape-3 is moved for direction given in d.
5. Go back to step 4, and carry on computation by