# P-completeness

Simon Samuel

National Institute of Technology Calicut

March 5, 2014

# Overview

1. CIRCUIT VALUE PROBLEM
2. ODD MAX FLOW

# CIRCUIT VALUE IS P-COMPLETE

To prove that CIRCUIT VALUE is P-Complete,

- CIRCUIT VALUE should be in P.
- For any language

$$L \in P$$

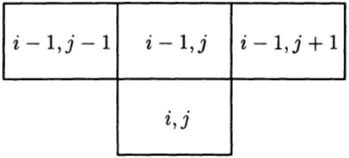
there is a reduction R from L to CIRCUIT VALUE.

Let M be the Turing machine that decides L in time $n^k$ , and consider the computation table of M on x, call it T.

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\triangleright$ | $0_s$ | 1 | 1 | 0 | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | $1_{q_0}$ | 1 | 0 | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | 1 | $1_{q_0}$ | 0 | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | 1 | 1 | $0_{q_0}$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | 1 | 1 | 0 | $\sqcup_{q_0}$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | 1 | 1 | $0_{q_0'}$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | 1 | $1_q$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | $1_q$ | 1 | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright_q$ | 1 | 1 | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | $1_s$ | 1 | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | $\triangleright$ | $1_{q_1}$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | $\triangleright$ | 1 | $\sqcup_{q_1}$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | $\triangleright$ | $1_{q_1'}$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | $\triangleright_q$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | $\triangleright$ | $\sqcup_s$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |
| $\triangleright$ | $\triangleright$ | $\triangleright$ | "yes" | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ | $\sqcup$ |

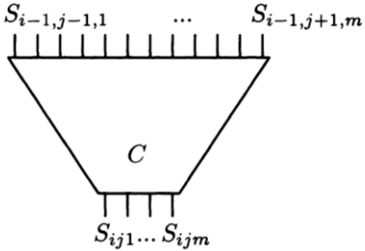
Figure : Computation Table

- When i = 0, or j = 0, or j = $|x|^k$ - 1, then the value of Tij is a priori known
- The value of $T_{ij}$ reflects the contents of position j of the string at time i, which depends only on the contents of the same position or adjacent positions at time i-1.
- That is, $T_{ij}$ depends only on the entries $T_{i-1,j-1}$, $T_{i-1,j}$, and $T_{i-1,j+1}$



- Let $\gamma$ denote the set of all symbols that can appear on the table (symbols of the alphabet of M, or symbol-state combinations).
- Encode next each symbol a $\sigma \in \gamma$ a vector $(s_1, .., s_m)$, where $s_1, ..., s_m \in \{0, 1\}$, and m = $[\log |\gamma|]$.
- The computation table can now be thought of as a table of binary entries $S_{ijl}$ with $0 < i < |x|^k - 1, 0 < j < |x|^k - 1, and 1 < l < m.$
- Each binary entry $S_{ijl}$ only depends on the 3m entries $S_{i-1,j-1,l'}$, $S_{i-1,j,l'}$, and $S_{i-1,j+1,l'}$, where $l'$ ranges over 1,..., m.

- That is, there are m Boolean functions $F_1, ..., F_m$ with 3m inputs each such that, for all $i, j > 0$
  $S_{ijl} = F_l(S_{i-l,j-l,l}, ..., S_{i-l,j-l,m}, S_{i-1,j,l}, ..., S_{i-l,j+l,m})$



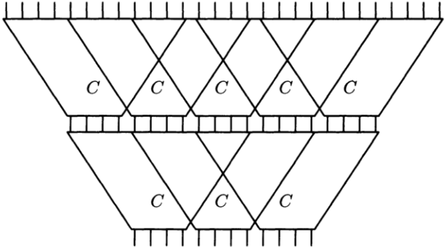



Figure : The construction of the circuit.

- It follows that there is a Boolean circuit C with 3m inputs and m outputs that computes the binary encoding of $T_{ij}$ given the binary encodings of $T_{i-1,j-1}$, $T_{i-1,j}$, and $T_{i-1,j+1}$ for all $i = 1, ..., |x|^k$ and $j = 1, ..., |x|^k - 1$.
- Circuit C depends only on M, and has a fixed, constant size, independent of the length of x.

- In our reduction R from L, for each input x, R(x) will basically consist of $(|x|^k - 1).(|x|^k - 2)$ copies of the circuit C.
- If $C_{ij}$ is the $(i, j)th$ copy of C, then for $i > 1$, the input gates of $C_{ij}$ will be identified with the output gates of $C_{i-1,j-1}$, $C_{i-1,j}$, and $C_{i-1,j+1}$.
- The input gates of the overall circuit are the gates corresponding to the first row, and the first and last column.
- Finally, the output gate of the R(x) is the first output of circuit $C_{|x|^k-1,1}$ (assuming that M always ends with "yes" or "no")

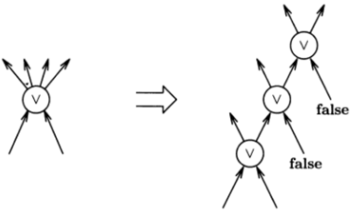
- Circuit C is fixed, depending only on M. The computation of R entails constructing the input gates (easy to do by inspecting x and counting up to $|x|^k$ ), and generating many indexed copies of the fixed circuit C and identifying appropriate input and output gates of these copies-tasks involving straightforward manipulations of indices, and thus easy to perform in O(log $|x|$) space.
- As every language L ( in P) can be reduced to CIRCUIT VALUE, it is P-Complete.

# ODD MAX-FLOW IS P-COMPLETE

ODD MAX-FLOW is P-Complete if,

- ODD MAX FLOW is in P.
- there is a reduction from CIRCUIT VALUE PROBLEM to ODD MAX FLOW.

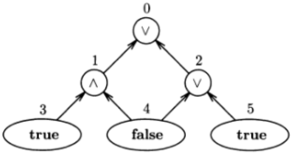
- Given a monotone circuit C, we assume that the output gate of C is an OR gate and no gate of C has out degree more than two.
- The gates of C are given consecutive numbers 0,1, . . . , n, so that each gate has a smaller label than its predecessor.
- Thus the output gate will have label 0, and the larger labels will be assigned to the inputs



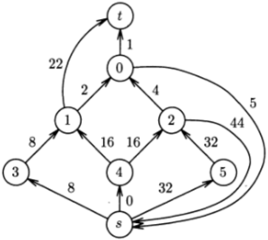
Construction:

- The network N = (V, E, s, t, c) produced from C has as its set of nodes the gates 0, . . . , n, plus two new nodes s and t.
- Edges leaving each node are given capacities= $d2^i$ where d is the outdegree of the gate and i is the label of the gate.
- Since AND or OR gate has at most two outgoing edges of capacity $2^i$ , and the capacities of each of the two incoming edges is at least twice (the labels of- its predecessors are strictly larger than i), there is a surplus of incoming capacity denoted as S(i).
- If i is an AND gate, there is an edge (i, t) of capacity S(i); if it is an OR gate, then there is an edge (i, s) of capacity S(i).

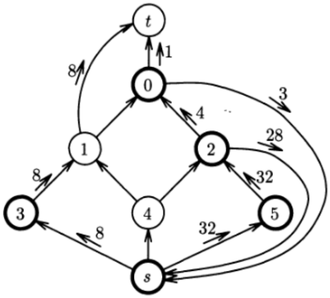
- A Gate is called full with respect to this flow if all of its outgoing edges to its successors gates are filled to capacity.
- It is called empty if all of these edges have zero flow.
- Flow f is called standard if all gates that have value true are full, and all gates that have value false are empty




(a)



- All true input gates have enough flow to become full and all false input gates must be empty (no incoming flow).
- All OR gates with true value have at least one incoming edge filled to capacity.
- All OR gates that have value false have no incoming flow, because their predecessors are empty.
- All AND gates with value true have both incoming edges filled.
- Finally all AND gates that have value false have at most one incoming edge filled with flow, which they can direct to the surplus edge .



- As CIRCUIT VALUE (which is P-Complete) can be reduced to ODD MAX FLOW, it is P-Complete.