• Since we have N with us,
we can think of this boolean formula as
$\phi_{N,w}$ that is satisfiable if and only if
any one branch of N goes to q_{accept}
within n^k (or n^k-3) moves
if w is given as input to N.

This we split in 4 parts:

(1) N's branch of computation is represented using an $n^k \times n^k$ matrix. Each cell contains exactly one symbol from the possible set of symbols.

Symbol set $C = \Gamma \cup Q \cup \{\#\}$

- ϕ_{cell}, the first part of the formula.

(2) The branch starts with ω as input.

ϕ_{start}, the next part

(3) ϕ_{move} : From every row in the matrix to the next row, it is a legal move on N.

(4) It goes to q_{accept} within $n^k - 3$ moves.

ϕ_{accept}'

- Once we define all four of them, we can see $\phi_{N,w}$ as an 'AND' of all four.

$$\phi_{N,w} = \phi_{cell} \wedge \phi_{move} \wedge \phi_{start} \wedge \phi_{accept}$$

(order does not matter).

- How to define each of them?

Idea:

ϕ_{cell}: Exactly one symbol in each cell.

or, $x_{i,j,s}$ = true (cell (i,j) contains s)

then $x_{i,j,t}$ = false (for a symbol $t \neq s$).

(and exactly for one s it is true)

- eg: $x_{1,1,\#}$ is true, $x_{1,1,t}$ is false for $t \neq \#$.

$$\phi_{cell} = \bigwedge_{i,j}\left[\left(\bigvee_{s\in C} x_{i,j,s}\right) \wedge \left(\bigwedge_{\substack{s,t\in C\\ s\neq t}} \overline{x_{i,j,s}} \vee \overline{x_{i,j,t}}\right)\right]$$

- In a similar way, how can we write other 3 parts of the formula?

ϕ_{start} : The branch starts with 'w' as input.

- If input $w = w_1 w_2 \ldots w_n$ (n symbols), initial configuration: $q_0 w_1 w_2 \ldots w_n$; the row in the tableau is

$$\underset{(1)}{\#}\ q_0\ w_1\ w_2\ \ldots\ w_n\ \sqcup\ \sqcup\ \ldots\ \ldots\ \underset{(n^k)}{\#}$$

$$\phi_{start} = x_{1,1,\#} \wedge x_{1,2,q_0} \wedge x_{1,3,w_1} \cdots \wedge x_{1,n+2,w_n} \wedge \cdots x_{1,n^k,\#}$$

(ensures these are the contents in row 1)

Similarly, ϕ_{accept} would ensure that the branch goes to q_{accept} (within n^k moves).

$$\phi_{accept} = \bigvee_{\substack{i=1 \text{ to } n^k \\ j=1 \text{ to } n^k}} x_{i,j,q_{accept}}$$

[The precise no. of moves, worst case, captured by this $n^k \times n^k$ tableau is $n^k - 3$]

- ϕ_{move} : Makes sure every row legally follows as a 'next configuration' for some move defined in the TM.
- Total no. of moves is a fixed number, not dependant on n.
- So this also can be done in a formula of size polynomial in n.

- ϕ_{move} can be split into ensuring that every 3x2 block in the tableau is "legal" as per the moves of N.

b	q_1	a
b	c	q_2

e.g. This block is 'legal' if there is a move

$$\delta(q_1, a) = (q_2, c, R).$$

also,

b	q_1	a
b	a	a

is not legal.

ϕ_{move} = $Block_{1,1}$ is legal $\wedge$ ..... $Block_{n^{k-1}, n^{k-2}}$ is legal.

- Note that each "legality check" takes only constant time.

[Because max. no. of moves is a fixed constant.]

Note that

$$\phi_{N,w} = \phi_{cell} \wedge \phi_{start} \wedge \phi_{move} \wedge \phi_{accept}$$

is satisfiable if and only if N, if it runs with w as input, goes to q_{accept} within n^k moves on at least one brach of computation.

$\square$

(SUDEEP)