

1. Any language in $SPACE(f(n))$ as defined using the two-tape read-only model can be simulated with a single tape model using at most $O(n)$ space. Similarly, any language in $SPACE(f(n))$ as defined using a single tape model can be simulated with a two-tape read-only model with an improvement of at most $O(n)$ space. Thus, the complexity classes are equivalent where $f(n) \geq n$.
2. The winning strategy for X is to move to the top-right position. O can then move to block only either the top-centre or centre-right position. If O moves to the top-centre, X moves to the centre-right. If O move to the centre-right, X moves to the top-centre.
3. Player I has a winning strategy as follows:
 - Player I begins at node 1.
 - Player 2 chooses node 2.
 - Player I chooses node 4. Node 3 has only one outgoing edge which connects to node 6. As node 6 has no outgoing edges, this path would guarantee a win for Player II.
 - Player II chooses node 5.
 - Player I chooses node 6. As no unchosen nodes remain, Player I wins.
4. Let L_i be a language decided by PSPACE turing machine M_i . We define languages $L_{\cup} = L_i \cup L_j$, $\bar{L}_i = \{w \mid w \notin L_i\}$, $L_i^* = \{x_1x_2 \dots x_k \mid k \geq 0 \text{ and each } x_i \in L_i\}$.

We define M_{\cup} , \bar{M}_i and M_i^* as follows:

$M_{\cup} =$ "On input w

1. Run $M_i\langle w \rangle$. If M_i accepts, *accept*.
2. Run $M_j\langle w \rangle$. If M_j accepts, *accept*.
3. If neither $M_i\langle w \rangle$, $M_j\langle w \rangle$ accepted, *reject*."

$\bar{M}_i =$ "On input w

1. Run $M_i\langle w \rangle$. If M_i accepts, *reject*, else *accept*."

$M_i^* =$ "On input w

1. If $w = \epsilon$, *accept*.
2. For each m , where $1 \leq m \leq n$, $n = |w|$.
 3. Split w into m pieces, such that $w = w_1w_2 \dots w_k$.
 4. For all i , $1 \leq i \leq m$, run $M_i\langle w_i \rangle$. If M_i rejects, go to step 2.

5. M_i has accepted for all i , *accept*.
 6. M_i has rejected for all m , *reject*."
-
5. Construct a TM M to decide A_{DFA} When M receives input $\langle A, w \rangle$, a DFA and a string, M simulates A on w by keeping track of A 's current state and its current head locations, and updating them appropriately. The space required to carry out this simulation is $O(\log n)$ because M can record each of these values by storing a pointer into its input.