About Time and Space for Non-deterministic Turing Machines

We first recall the definitions. Let $\gamma = q_0 w \to C_0 \to C_1 \to \ldots \to C_m$ be a computation of a Turing machine $\mathcal{M}$ on an input word w.

1. the time $t_{\mathcal{M}}(\gamma)$ of this computation is m
2. the space $s_{\mathcal{M}}(\gamma)$ is the number of cells visited by the head during the computation

Complexity for an input w:

$$t_{\mathcal{M}}(w) = \max_{\gamma} t_{\mathcal{M}}(\gamma) \quad \text{et} \quad s_{\mathcal{M}}(w) = \max_{\gamma} s_{\mathcal{M}}(\gamma)$$

Complexity (at worst case):

$$t_{\mathcal{M}}(n) = \max_{|w|=n} t_{\mathcal{M}}(w) \quad \text{et} \quad s_{\mathcal{M}}(n) = \max_{|w|=n} s_{\mathcal{M}}(w)$$

I mentionned during the last lecture that for a non-deterministic Turing machine, the space complexity $s_{\mathcal{M}}(w)$ for an input word w could come from one branch of the computation tree, while the time complexity $t_{\mathcal{M}}(w)$ could come from another branch. Let us assume that for the input word w we have only two finite branches, corresponding to two computations, respectively γ_1 and $\gamma 2$. We claim that the inequality $s_{\mathcal{M}}(w) \leq t_{\mathcal{M}}(w)$ always holds and thus $s_{\mathcal{M}}(n) \leq t_{\mathcal{M}}(n)$ for all $n \geq 0$.

Assume that computation γ_1 defines the space complexity, that is, $s_{\mathcal{M}}(w) = \max_{\gamma} s_{\mathcal{M}}(\gamma) = s_{\mathcal{M}}(\gamma_1)$. By the argument that in one step, the machine can not read more than one cell, we have $s_{\mathcal{M}}(\gamma_1) \leq t_{\mathcal{M}}(\gamma_1)$. In other words, on a deterministic computation, which is the case of the computation γ_1 taken alone, this inequality is easily verified. Now assume that time complexity is defined by the other computation γ_2, that is, $t_{\mathcal{M}}(w) = \max_{\gamma} t_{\mathcal{M}}(\gamma) = t_{\mathcal{M}}(\gamma_2)$. Notice that the last equality implies $t_{\mathcal{M}}(\gamma_1) \leq t_{\mathcal{M}}(\gamma_2)$. Then we have:

$$s_{\mathcal{M}}(w) = s_{\mathcal{M}}(\gamma_1) \leq t_{\mathcal{M}}(\gamma_1) \leq t_{\mathcal{M}}(\gamma_2) = t_{\mathcal{M}}(w)$$

In other words, if we start from a computation γ_1 which defines the space complexity for input word w, then $s_{\mathcal{M}}(w) = s_{\mathcal{M}}(\gamma_1) \leq t_{\mathcal{M}}(\gamma_1)$, and we have the alternative:

— if for all other computation γ, $t_{\mathcal{M}}(\gamma) < t_{\mathcal{M}}(\gamma_1)$, then $t_{\mathcal{M}}(w) = t_{\mathcal{M}}(\gamma_1)$ and thus $s_{\mathcal{M}}(w) = s_{\mathcal{M}}(\gamma_1) \leq t_{\mathcal{M}}(\gamma_1) = t_{\mathcal{M}}(w)$;

— if not, then there is at least a computation γ such that $t_{\mathcal{M}}(\gamma_1) \leq t_{\mathcal{M}}(\gamma)$ and $s_{\mathcal{M}}(w) = s_{\mathcal{M}}(\gamma_1) \leq t_{\mathcal{M}}(\gamma_1) \leq t_{\mathcal{M}}(\gamma) \leq t_{\mathcal{M}}(w)$ as $t_{\mathcal{M}}(w)$ is the maximal value over all computations.