Proof of $T_p \leq T_1/p + T_\infty$ based on a thread counting argument

Alexei Finski

\# complete steps $\leq T_1/p$

Suppose # complete steps $> T_1/p$. The size of a complete step is p. The work performed is $> T_1$. Contradiction.

\# incomplete steps $\leq T_\infty$

Wlog, let the execution time for each thread be unit time. Every path in G starts from a single source thread and its length is shorter or equal to T_∞. For every thread t_i in a longest path of G there exists a set of threads s_i that can be executed in parallel.

If s_i is executed, then every thread in s_{i+1} is executable or executed. By induction, at any time before program completion there exists s_i^*, a set of executed and executable threads with at least one thread that is executable.

An incomplete step of a greedy scheduler must execute the last executable thread of s_i^*. Otherwise the step is complete. Thus # incomplete steps $\leq T_\infty$.