- **MP-SCHED is NP.**

  A certificate of instance $(X,\ m,\ d)$ is m sequences of jobs. We can check in polynomial time O($\mid X \mid$) whether the sum of job lengths in each sequence is not greater than d.

- **MP-SCHED is NP-hard.**

  We find 2-PARTITION known to be NP-Complete.

  - **Reduction**

    We only consider 2-PARTITION instances where non-negative integers are involved. If N contains some negative integers, we can transform it to N' by finding the smallest integer n_{min} and adding the absolute value of it to every number in N so that the smallest integer in N' is now 0 with $t' = t + n_{min} \mid N \mid$.

    We construct a MP-SCHED instance with two processors with a deadline $d = \frac{L}{2}$ where L denotes the total length of all jobs. For each $n \in N$, we can add the same x to X. The reduction function runs in polynomial time O($\mid N \mid$).

  - **Claim 1: If $N \in$ 2-PARTITION, then $(X,\ m,\ d) \in$ MP-SCHED.**

    If $N \in$ 2-PARTITION, then d must be an integer and that there exists two partitions of N where each partition sums up to $\frac{L}{2}$. We can arbitrarily permute jobs in each partition since the order does not change their sum. By construction $d = \frac{L}{2}$, we conclude that all sequences of jobs are finished exactly at d in the corresponding MP-SCHED instance.

  - **Claim 2: If $(X,\ m,\ d) \in$ MP-SCHED, then $N \in$ 2-PARTITION.**

    If the reduced instance $(X,\ m,\ d) \in$ MP-SCHED, then there exists two sequences in X, each sequence containing numbers that sum up to no larger than d. We also know that the total length of jobs is L, so it must be the case that both sequences have a duration of exactly $\frac{L}{2}$. For each job in a sequence we have the same number to the corresponding subset. Then there exists a way of partitioning N.

  Thus MP-SCHED is NP-hard.

- **MP-SCHED is NP-Complete.**

  Because MP-SCHED is both NP and NP-hard, it is also NP-Complete.