Greedy

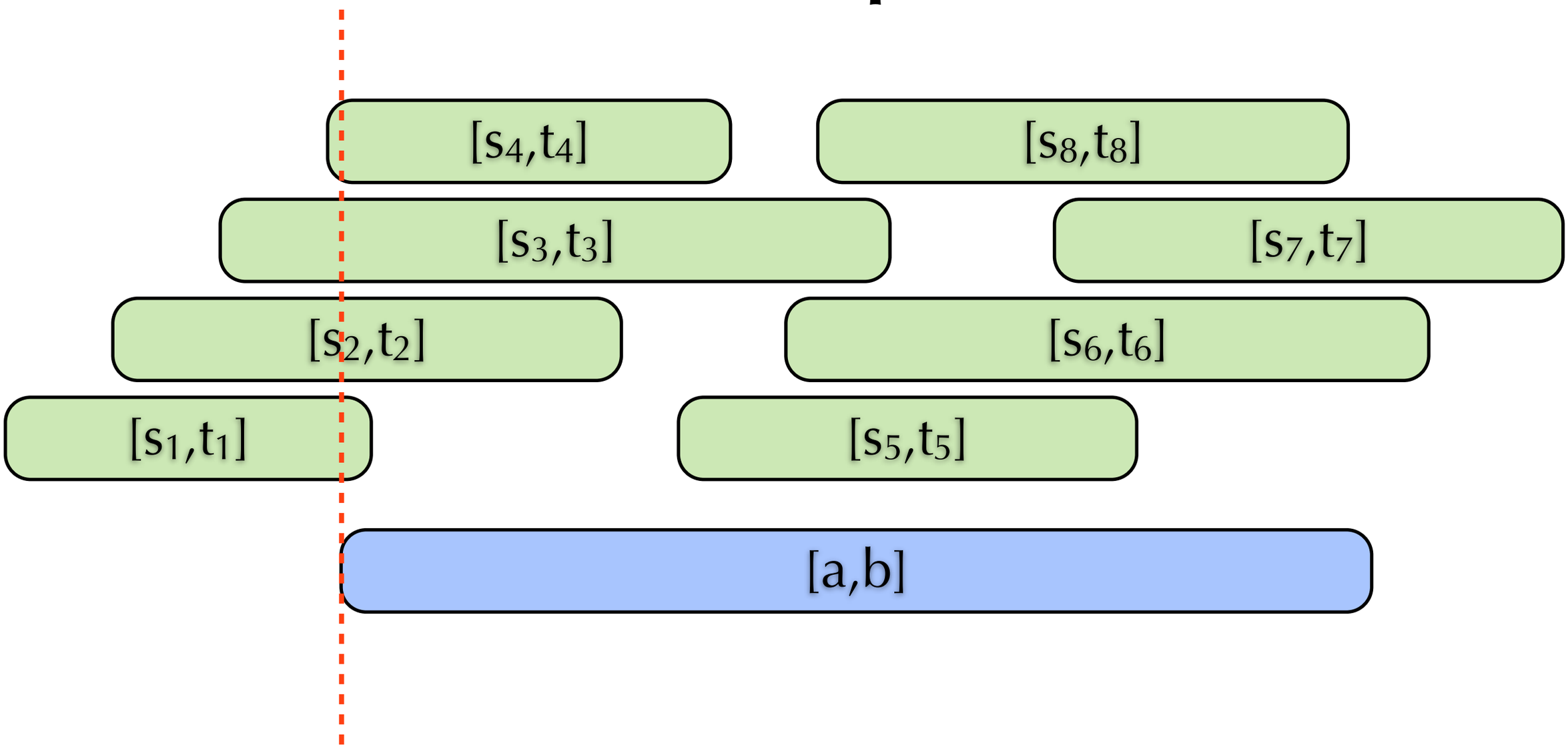
Greedy Algorithms

- Idea: Construct a global optimal solution by making locally greedy choices.
- Works when the problem has
  - Optimal substructures
  - Greedy choice property
- Implementation
  - Priority queues
  - Sorting

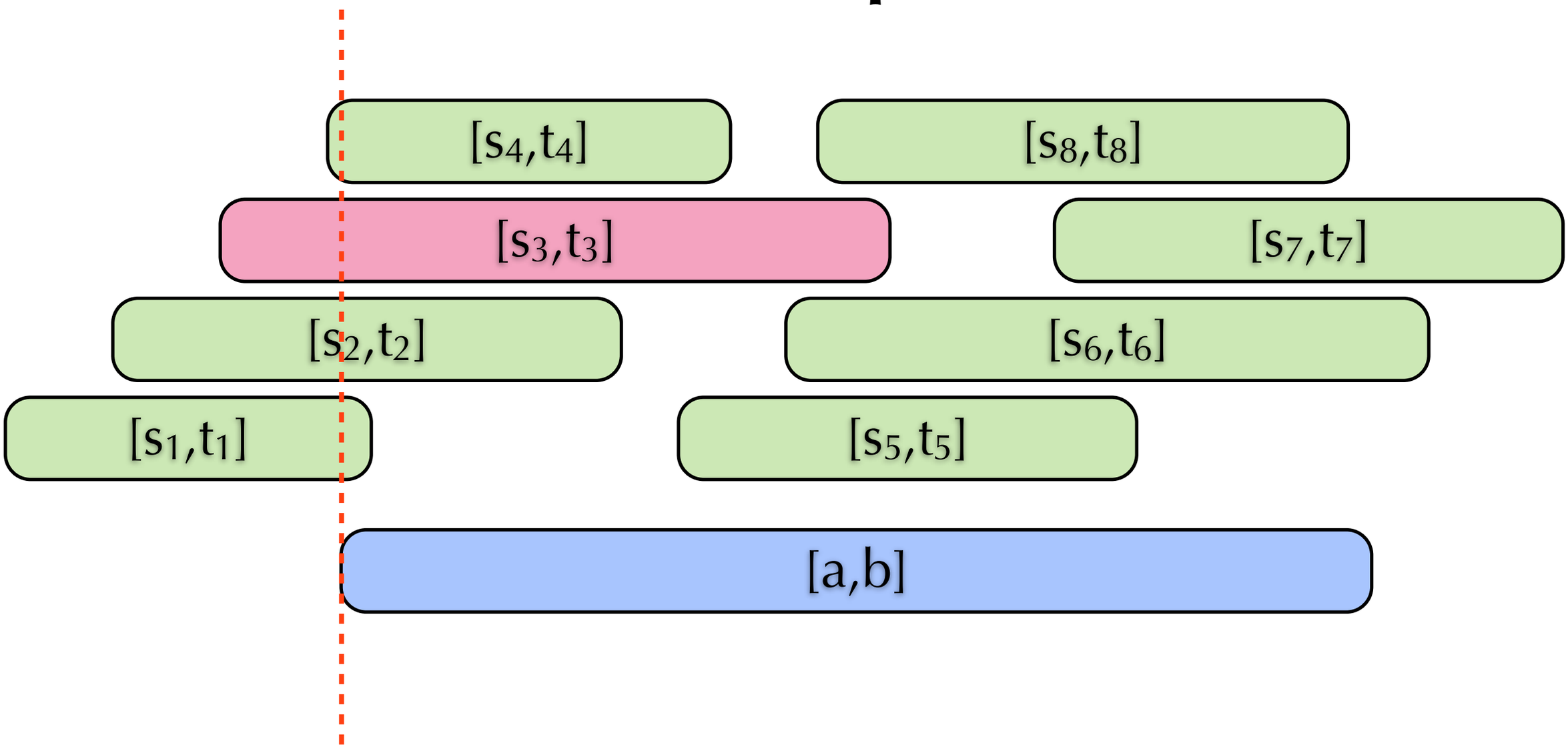
Interval Covering

- Input: A interval set $S=\{[s_1,t_1],\ldots,[s_n,t_n]\}$
- Goal: Find the minimum subset of S which covers all points in [a,b]
- Sketch:
  - Find the interval $[s^*,t^*]$ such that
    - $[s^*,t^*]\in\{[s,t]: s\le a\le t\}$
    - $t^*\ge t'$ if $[s',t'] \in\{[s,t]: s\le a\le t\}$
  - Put $[s^*,t^*]$ into the optimal solution and deal with subproblem $[t^*,b]$.

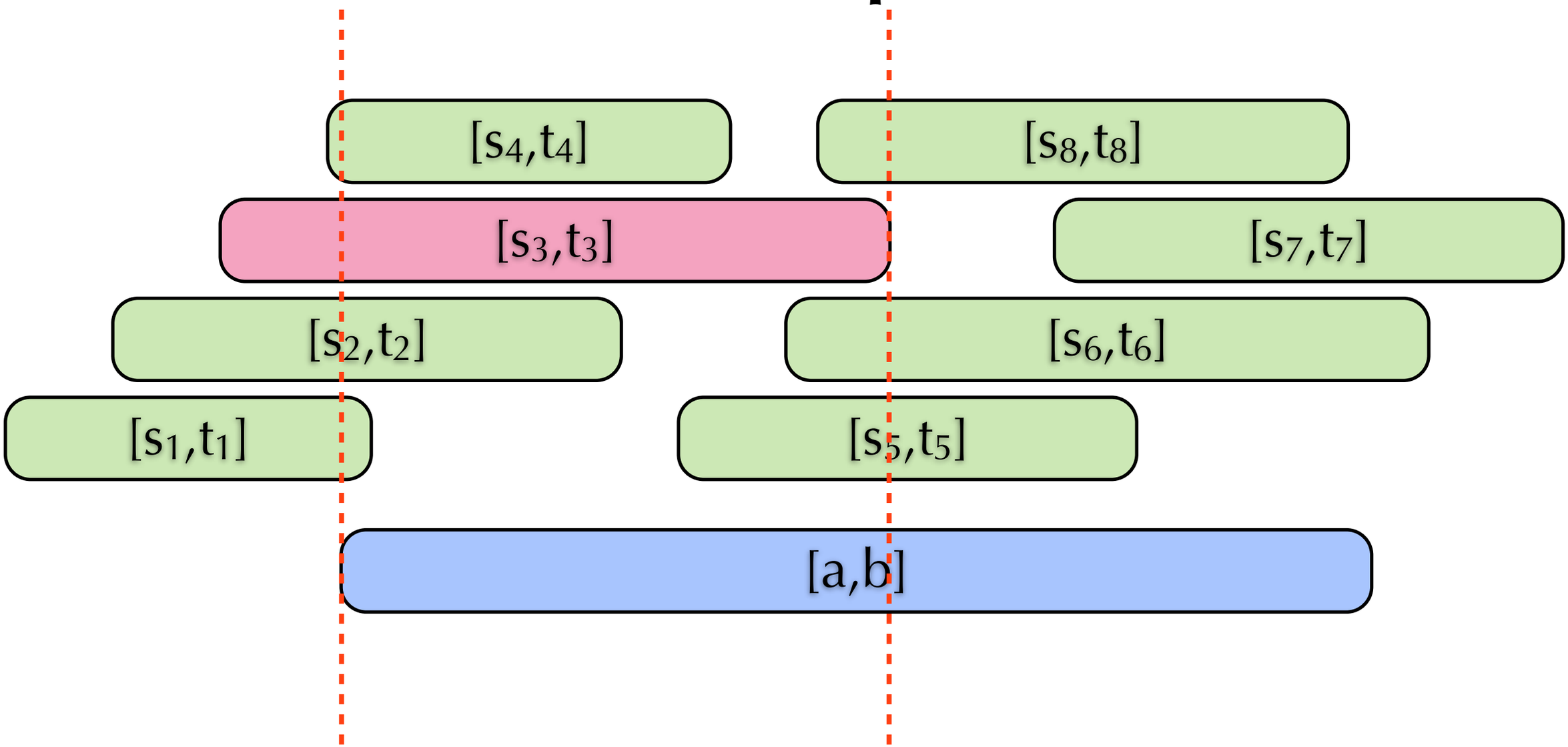
Example



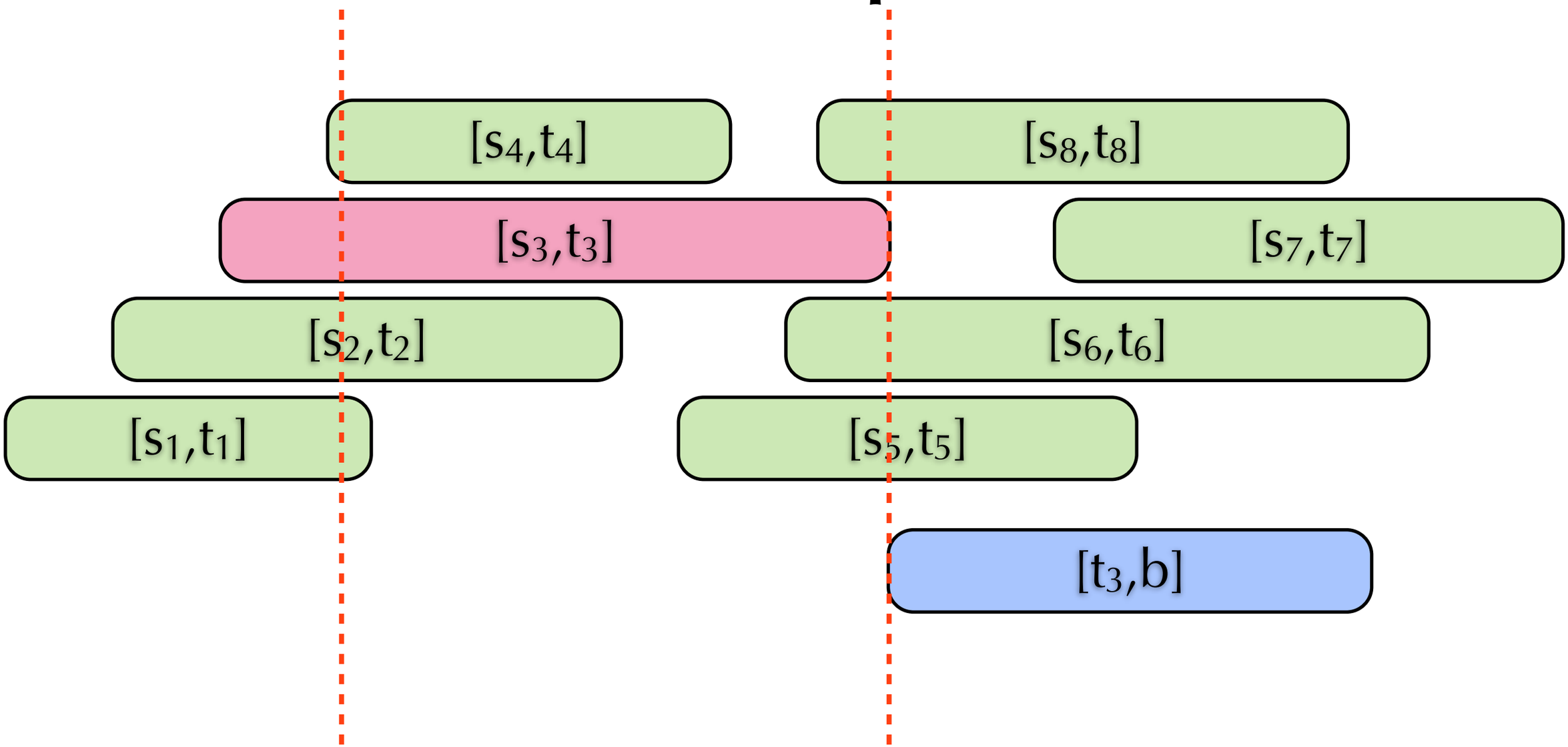
Example



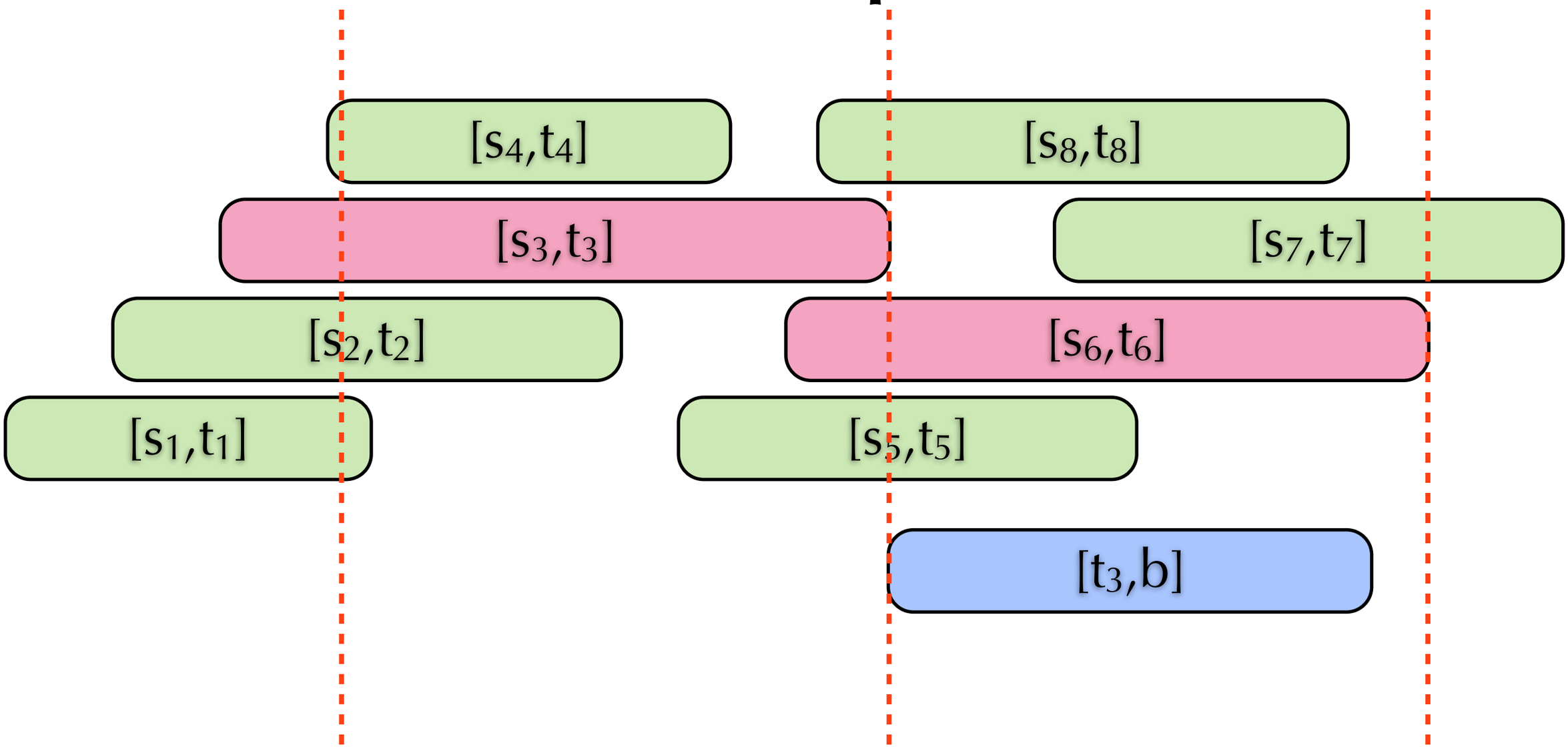
Example



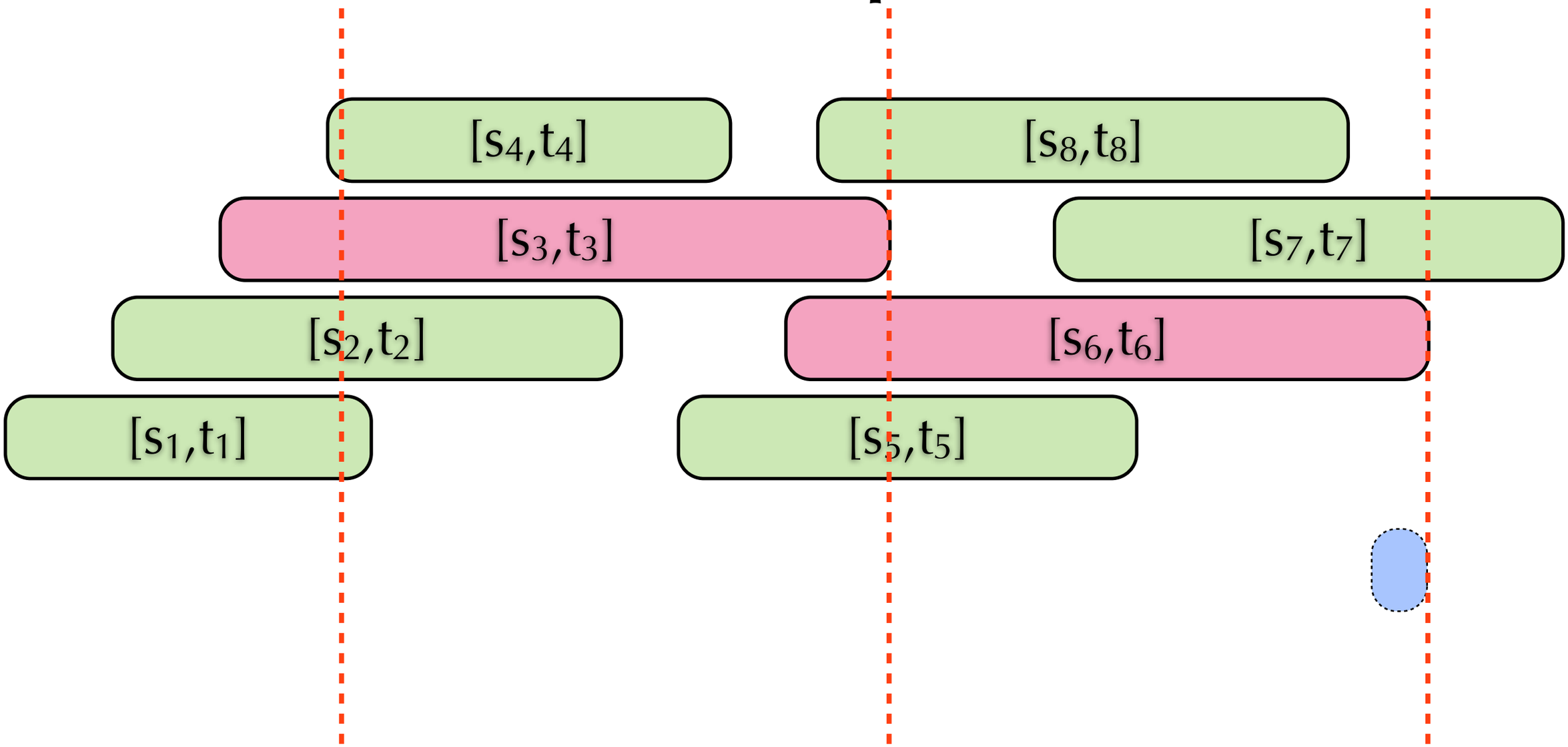
Example



Example



Example



Done!

1D Point Covering

- Input: A set $P=\{x_1,\ldots,x_n\}$ of points on the real line.
- Covering set of P: $S=\{I_1,\ldots,I_k\}$
  - I_j's are unit-length intervals
  - For every $x \in P$, there exists $I \in S$ such that $x \in S$.
- Goal: what is the minimum cardinality of a covering set of P?

Example

Example

Example

Example

Example

Example

Example

Done!