Noname manuscript No.
(will be inserted by the editor)

Intelligent Task Scheduling - Algorithm

Lukáš Korous

the date of receipt and acceptance should be inserted later

1 Application - Intelligent task scheduling

The user inserts a set of tasks, with dependencies among them, with deadlines, with conditions limiting when these tasks can be carried out and in how long intervals of work has the total amount of work for the task be done.

The application returns a set of time intervals ("slots"), each associated with a certain task, for which all the conditions, dependencies, and deadlines are fulfilled.

2 Mathematical description

2.1 Data

All values are integers.
We are given a set of N entities ("tasks")

$$T = \{t_i\}_{i=1}^{N}$$

with *len* attribute representing length:

$$\{len\,(t_i)\}_{i=1}^{N}, len\,(t_i) > 0, i = 1, ..., N,$$

with *dl* attribute representing deadline:

$$\{dl\,(t_i)\}_{i=1}^{N}, dl\,(t_i) > 0, i = 1, ..., N,$$

L. Korous
Softica Solutions
E-mail: lukas@softica.cz

attributes *minSlotLen*, *maxSlotLen* representing minimum and maximum length of intervals associated with the task:

$$minSlotLen > 0,\ maxSlotLen > 0,\ maxSlotLen >= minSlotLen$$

$$\{minSlotLen\,(t_i)\}_{i=1}^{N}, \{maxSlotLen\,(t_i)\}_{i=1}^{N},$$

set of intervals ("allowed slots") where the intervals associated with the task can be scheduled (N_{a_i} is known):

$$\{[a_{ijL}, a_{ijR}]\}_{j=1}^{N_{a_i}}, i = 1, ..., N,$$

and *dep* predicates representing dependencies:

$$dep_{ij},$$

where dep_{ij} has the following meaning: dep_{ij} iff t_j depends on t_i.

2.2 Conditions

We are looking for a set of integer intervals ("slots"):

$$\{[s_{ijL}, s_{ijR}]\}_{j=1}^{N_i},\ i = 1, ...N,$$

where N_i is the (unknown) number of intervals associated with the task t_i with the following conditions:

1. sum of intervals = length of task:

$$\sum_{j=1}^{N_i} s_{ijR} - s_{ijL} = len\,(t_i),\ i = 1, ..., N$$

2. intervals do not overlap:

$$\forall i, j, k, l : s_{ijL} >= s_{klR} \cup s_{klL} >= s_{ijR}$$

3. dependencies are satisfied:

$$dep_{ik} \rightarrow \forall i, j, k, l : s_{ijR} <= s_{klL}$$

4. minSlotLen, maxSlotLen are respected:

$$\forall i, j : s_{ijR} - s_{ijL} \in [minSlotLen\,(t_i), maxSlotLen\,(t_i)]$$

5. Allowed slots $[a_{ijL}, a_{ijR}]$ are respected:

$$\forall i, j : \exists k \in [1, ..., N_{a_i}] : s_{ijR} <= a_{ikR} \cap s_{ijL} >= a_{ikL}.$$

6. Deadlines are satisfied

$$\forall i, j : s_{ijR} <= dl\,(t_i)$$