

Resource Constrained Task Scheduling

In this tutorial, we are going to schedule a set of tasks which require use of a number of limited resources. Once a task is finished, the associated resources can be released for other tasks to use. To load this project, search for sched_rcpsp in the project loader in ILOG CPLEX Studio.

In this document, we will explain the mod file in detail.

define the number of tasks and resources.

```
int NbTasks = ...;  
int NbRsrcs = ...;
```

define the range of resource ids, indexing from 0.

```
range RsrcIds = 0..NbRsrcs-1;
```

given the range, load resource capacity parameter.

```
int Capacity[r in RsrcIds] = ...;
```

define the tuple to specify parameters for each task, including task id (unique id), processing time, resource demand, successors

```
tuple Task {  
    key int id;  
    int pt;  
    int dmds[RsrcIds];  
    {int} succs;  
}  
{Task} Tasks = ...;
```

define decision variables - interval variable for the tasks with size equalling to the processing time

```
dvar interval itvs[t in Tasks] size t.pt;
```

define the cumulative function for each resource; function pulse means the resource is used throughout the whole interval for each task.

```
cumulFunction rsrcUsage[r in RsrcIds] =  
    sum (t in Tasks: t.dmds[r]>0) pulse(itvs[t], t.dmds[r]);
```

limit CP search time

```
execute {
    cp.param.FailLimit = 10000;
}
```

objective is to minimise max completion time of all tasks.

```
minimize max(t in Tasks) endOf(itvs[t]);
```

define constraints that each resource is limited by its capacity

```
forall (r in RsrcIds)
    rsrcUsage[r] <= Capacity[r];
```

define constraints that a task must be finished before its successors. Note that <t2id> is used to index a successor. Because t2id is the unique id of the task, it can be used as a partial reference for the actual task. Normally we need to include all elements in a tuple, but if we use unique id, we can do this.

```
forall (t1 in Tasks, t2id in t1.succs)
    endBeforeStart(itvs[t1], itvs[<t2id>]);
```

To view the solution, we can click Problem Browser and click the decision variables. The gantt chart is very useful to view intervals, it will show the intervals in a graph for easy understanding.

