Gurobi Optimization for Task Scheduling

1 Overview

This document describes the process of optimizing task scheduling across multiple papers and jobs using Gurobi. The primary objective is to minimize the makespan, the total time required to complete all tasks.

2 Problem Definition

The problem involves scheduling tasks, each associated with a paper and a job. Each task has a specific duration and may have a precedence constraint, meaning it must start only after another specified task is completed.

2.1 Tasks

Tasks are defined with the following attributes:

- **Duration** ($d_{i,j}$): The time required to complete the task for paper i and job j.
- **Precedence** ($p_{i,j}$): Another task that must be completed before this task can start. Denoted as (k, l), indicating that task (k, l) must finish before (i, j) can start.

3 Model Setup

3.1 Initialization

A Gurobi model is created to handle the optimization process.

3.2 Variables

- **Makespan** (M): A continuous variable representing the total time required to complete all tasks.
- **Start Times** ($s_{i,j}$): Continuous variables representing the start times for each task (i, j).

- **Order Variables** (o_{i,j_1,j_2}): Binary variables to determine the order of tasks within the same job but different papers.

3.3 Adding Variables

Start time variables are added for each task, and order variables are created to enforce the precedence of tasks within the same job but different papers.

3.4 Constraints

1. **Makespan Constraint**: Ensures each task finishes before the makespan.

$$s_{i,j} + d_{i,j} \leq M \quad \forall (i,j) \tag{1}$$

2. **Precedence Constraints**: Ensures that tasks with precedence constraints start only after their predecessor tasks are completed.

$$s_{k,l} + d_{k,l} \leq s_{i,j} \quad \forall (i,j) \text{ with } p_{i,j} = (k,l) \tag{2}$$

3. **Order Constraints**: Enforces the order between tasks within the same job.

$$s_{j_1,i} + d_{j_1,i} \leq s_{j_2,i} + (1 - o_{i,j_1,j_2}) \cdot M \quad \forall (i, j_1, j_2) \text{ where } j_1 \neq j_2 \tag{3}$$

$$s_{j_2,i} + d_{j_2,i} \leq s_{j_1,i} + o_{i,j_1,j_2} \cdot M \quad \forall (i, j_1, j_2) \text{ where } j_1 \neq j_2 \tag{4}$$

3.5 Objective

The objective is to minimize the makespan, thereby ensuring that all tasks are completed in the shortest possible time.

$$\text{Minimize } M \tag{5}$$

4 Optimization and Results

4.1 Optimization

The model is optimized using Gurobi's solver. If an optimal solution is found, the schedule is displayed.

4.2 Displaying Results

The solution is presented by listing the start times, durations, and finish times for each task in a tabulated format, organized by job.

5 Conclusion

This document outlines the approach to setting up and solving a task scheduling problem using Gurobi. The tasks are scheduled efficiently, respecting their durations and precedence constraints, with the goal of minimizing the overall makespan.