Project Task Scheduling and Budget Optimization using Genetic Algorithm

This project aims to optimize the scheduling of project tasks and budget allocation using a genetic algorithm (GA). Given a set of project tasks, a team with different skillsets, and budget constraints, the GA will generate the best schedule that satisfies the dependencies and skill requirements of each task while minimizing the project duration and budget deviation.

Dependencies

This project requires the following libraries to be installed:

- numpy
- matplotlib
- tkinter

Project Parameters

Team Members

- **team_members**: An integer representing the number of team members available for the project.

Project Tasks

- **project_tasks**: A list of dictionaries representing the tasks required to complete the project. Each dictionary contains the following keys:
    - **id**: An integer representing the task's unique identifier.
    - **hours**: An integer representing the number of hours required to complete the task.
    - **required_project_skills**: A list of integers representing the skills required to complete the task. Each integer corresponds to the index of the skill in the team_member_skills list.
    - **dependencies**: A list of integers representing the tasks that must be completed before this task can begin.

Team Member Skills

- **team_member_skills**: A list of dictionaries representing the skills of each team member. Each dictionary contains the following keys:
    - **id**: An integer representing the team member's unique identifier.
    - **skills**: A list of integers representing the skills possessed by the team member. Each integer corresponds to the index of the skill in the project_skills list.

Budget Parameters

- **budget**: An integer representing the budget available for the project.
- **normal_rate**: An integer representing the hourly rate for normal work hours.
- **overtime_rate**: An integer representing the hourly rate for overtime work hours.
- **overtime_threshold**: An integer representing the number of hours per day after which overtime rates apply.

GA Parameters

- **population_size**: An integer representing the size of the GA population.
- **crossover_prob**: A float representing the probability of performing crossover during the GA.
- **generations**: An integer representing the number of GA generations.
- **max_generations_without_improvement**: An integer representing the maximum number of GA generations without improvement before stopping.

Risk Management Parameters

- **sick_probability**: A float representing the probability of a team member being sick.
- **sick_days**: An integer representing the number of days a team member is sick.

Functions

The program consists of several functions that perform different tasks:

create_individual()

This function creates an individual in the genetic algorithm population, which is a list of task assignments for each team member.

evaluate(individual)

This function evaluates an individual by calculating the project duration and the additional budget required to complete the project. It also accounts for the risk of team members being sick and not available for work.

ga(num_restarts, acceptable_fitness, max_restarts_without_improvement)

This function performs the genetic algorithm optimization by creating an initial population, evaluating the fitness of each individual, and applying genetic operators such as crossover and mutation. It returns the best individual found and its fitness.

show_schedule(best_individual)

This function creates a GUI window that displays the schedule of task assignments for each team member in a tabular format.

calculate_task_times(best_individual)

This function calculates the start and end times of each task based on the assigned hours and task dependencies.

plot_gantt_chart(task_times, project_duration)

This function creates a Gantt chart that visualizes the project schedule and duration.

Results

The project scheduler will display the best project schedule and a Gantt chart showing the task timeline. Additionally, the project scheduler will output the best individual (i.e., the optimal task scheduling), the project duration, and the additional budget required to complete the project within the given constraints.