Solver for vehicle routing problems

AWS BATCH SERVICE

INFO@SHOBB.SITE

Contents

AWS Batch Service Description

- To run a batch service in the Amazon cloud, you need the appropriate attributes:
    - User credentials:
        - AWS_DEFAULT_REGION
        - AWS_ACCESS_KEY_ID
        - AWS_SECRET_ACCESS_KEY
    - Batch job parameters:
        - Name of job queue
        - Name of job definition
        - Name of S3 bucket which will be used uploading task and downloading result files for serving.
    - All these parameters are customizable for a specific user.
- You upload the task file into a given S3 Aws bucket, run a batch job, and wait until the solution file is ready for download.
- After executing the batch job, if the solver successfully finds a solution, a file will be generated with the same name as the task file and with a .sol extension. (See file example.sol), otherwise a text file with the .err extension is generated with a description of the errors.
- If the solver cannot read the file and all the information it needs, it does nothing.

Task file description:

- Description of each task must be in a zip archive (with the zip extension). *See file example.zip*.
- This file must contain the following items:
    1. **Points.json** – list of Points
    2. **Vehicles.json** – list of vehicles
    3. **Orders.json** – list of orders
    4. **Matrix.json** – Time/distance matrix. If you need more matrices (up to 4), you will need to add Matrix1.json … Matrix3.json.
    5. Optionally: *options.json* – with options to solver:
        - MinInuse – if 1 then solver will try to minimize number of vehicles in solution even if this solution would have higher cost, otherwise 0. The default value is 0.
        - MinReturns – if 1 then solver will try to minimize number of visits to places, which are loading points of several transfer orders, otherwise 0. The default is 0.
        - NumSolvers – Number of solvers, working in parallel. Must be greater than zero. The default is 1. This is sufficient for most tasks.
        - NumIterations – Number of iterations to solve the task. The default value is 100000. It is also sufficient for most tasks.
    6. Optionally: *sdata.txt* – The initial solution from previous run.
- For description of Points, Vehicles, Orders and Matrix – see *description.docx*.

Solution file description:

- The solution file will have the same name as the problem file and the extension .sol. Inside it is a zip archive with the following elements:
  1. **result.json** – information about the solution in json format. *See description.docx*.
  2. **result.txt** – a solution in human-readable text form.
  1. **sdata.txt** – a file that, if necessary, can be used as an initial solution to the task.

An example of executing a batch job with Aws cli

- Upload task file:
  - ➢ aws s3 cp example.zip s3://<bucket_name>
- Submit job:
  - ➢ aws batch submit-job --job-name <your_job_name> --job-queue <your_job_queue> --job-definition <your_job_definition> --parameters inputfile=example.zip
- List the contents of the bucket:
  - ➢ aws s3 ls s3://<bucket_name>
- Download the solution file from the bucket:
  - ➢ aws s3 cp s3://<bucket_name>/example.sol example.sol