Optimisation & Decision: list of projects

Course faculty: Duarte Valério, Filipe Santos

February 16, 2025

Contents

General rules

For delivery deadlines and for the rules that apply to project delivery, see the announcements in the course webpage, and, in particular, the first announcement:

- `https://fenix.tecnico.ulisboa.pt/disciplinas/OD36/2024-2025/2-semestre`

The two parts that make up the project are the same for all groups:

Part 1. Present the mathematical formulation of the problem. Simplify the problem as needed so as to formulate it as a LP problem; then solve it using the simplex method. In your report, when documenting the simplifications of the model, comment also on their impact on the results and on the solution of the real problem.

Part 2. Use at least two methods to solve the problem without the simplifications employed for Part 1. One of the two methods has to be a metaheuristic. You may, of course, use three methods or more, in which case it is recommended that two methods be metaheuristics. Compare the results obtained with the different methods, including the simplex method from Part 1. Conclude your report with a critical analysis of the methods and their results. Your report should also have all the relevant bibliographical references.

Students may, if interested, use datasets different from those appointed. In such case, please talk to the faculty to validate the dataset beforehand.

Acknowledgement

Most of the projects below were compiled by Susana Vieira, João Sousa, Filipe Santos, and others.

Chapter 1

Open Shop Scheduling (Taillard datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Open Shop Scheduling datasets from

- `http://mistic.heig-vd.ch/taillard/problemes.dir/ordonnancement.dir/ordonnancement.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 2

Flow Shop Scheduling (Taillard datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Flow Shop Scheduling datasets from

- `http://mistic.heig-vd.ch/taillard/problemes.dir/ordonnancement.dir/ordonnancement.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 3

Job Shop Scheduling (Taillard datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Job Shop Scheduling datasets from

- `http://mistic.heig-vd.ch/taillard/problemes.dir/ordonnancement.dir/ordonnancement.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 4

Aircraft Landing Scheduling (Beasley datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Aircraft Landing datasets from

- `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 5

Common Due Date Scheduling (Beasley datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Common Due Date datasets from

- `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 6

Flow Shop Scheduling (Beasley datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Flow Shop datasets from

- `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 7

Hybrid Reentrant Shop Scheduling (Beasley datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Hybrid Reentrant Shop datasets from

- `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 8

Job Shop Scheduling (Beasley datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Job Shop datasets from

- `http://people.brunel.ac.uk/∼mastjjb/jeb/info.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 9

Lot Streaming Scheduling (Beasley datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Lot Streaming datasets from

- `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 10

Open Shop Scheduling (Beasley datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Open Shop datasets from

- `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 11

Shift Minimisation Personnel Task Scheduling (Beasley datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Shift Minimisation Personnel Task datasets from

- `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 12

Weighted Tardiness Scheduling (Beasley datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Weighted Tardiness datasets from

- `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 13

Flexible Job Shop Scheduling (Brandimarte datasets)

There are different types of scheduling problems in production planning and operations management. Each type of problem has a different set of constraints and requirements.

In **Open Shop Scheduling**, a set of jobs are processed through a set of machines, but the order in which the jobs are processed on each machine is not fixed. Each job can be processed on any machine, subject to availability. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The open shop is a less structured environment than the flow shop, as there is more flexibility in the order and scheduling of the jobs.

In **Flow Shop Scheduling**, a set of identical jobs are processed through a sequence of machines in a fixed order. Each job must pass through all machines in the same sequence. The objective is to minimize the total completion time, which is the sum of the processing times for each job on each machine. The flow shop is a highly structured environment, where each job must follow a fixed path, and there is no flexibility in the order or scheduling of the jobs.

In **Job Shop Scheduling**, a set of jobs are processed through a set of machines, but each job requires a different sequence of operations on the machines. The job shop is the most flexible environment, as each job can have a different sequence of operations, and there is more flexibility in the scheduling of the jobs.

Objective function

Minimize the makespan (total time to complete all jobs) or minimize the total completion time (sum of completion time of all jobs).

Project material

Choose at least two Flexible Jobshop datasets from

- `https://www.ime.usp.br/~cris/fjs/benchmark/brandimarte/`

Reference

- Pinedo, M. L. (2022). Scheduling: Theory, algorithms, and systems. `https://doi.org/10.1007/978-3-031-05921-6`

Chapter 14

Timetabling (Unitime datasets)

The timetabling problem consists in scheduling a sequence of lectures between teachers and students in a prefixed period of time (typically a week), satisfying a set of constraints of various types. The manual solution of the timetabling problem usually requires many person-days of work. In addition, the solution obtained may be unsatisfactory in some respect; for example, a student may not be able to take desired courses, because they are scheduled at the same time.

A large number of variants of the timetabling problem have been proposed in the literature, which differ from each other based on the type of institution involved (university or school) and the type of constraints. There are three main types.

School Timetabling addresses the weekly scheduling for all the classes of a school, with restrictions such as not assigning two classes at the same time to the same teacher, and so on.

Course Timetabling addresses the weekly scheduling for all the lectures of a set of university courses, minimizing the overlaps of lectures of courses having common students.

Examination Timetabling addresses the scheduling of the exams of a set of university courses, avoiding overlap of exams of courses having common students, and spreading the exams for the students as much as possible.

Objective function

Find feasible solutions for different instances of a selected dataset.

Project material

Choose at least two datasets from

- `https://www.unitime.org/exam_datasets.php`

References

- Schaerf, A. (1999). Survey of automated timetabling. Artificial Intelligence Review, 13(2), 87–127. `https://doi.org/10.1023/A:1006576209967`
- Almeida, J., Santos, D. R. dos, Figueira, J. R. (2022). A Multi-Objective Model for Thesis Defence Scheduling. `https://doi.org/10.48550/arxiv.2205.07727`

Chapter 15

Timetabling (Nottingham datasets)

The timetabling problem consists in scheduling a sequence of lectures between teachers and students in a prefixed period of time (typically a week), satisfying a set of constraints of various types. The manual solution of the timetabling problem usually requires many person-days of work. In addition, the solution obtained may be unsatisfactory in some respect; for example, a student may not be able to take desired courses, because they are scheduled at the same time.

A large number of variants of the timetabling problem have been proposed in the literature, which differ from each other based on the type of institution involved (university or school) and the type of constraints. There are three main types.

School Timetabling addresses the weekly scheduling for all the classes of a school, with restrictions such as not assigning two classes at the same time to the same teacher, and so on.

Course Timetabling addresses the weekly scheduling for all the lectures of a set of university courses, minimizing the overlaps of lectures of courses having common students.

Examination Timetabling addresses the scheduling of the exams of a set of university courses, avoiding overlap of exams of courses having common students, and spreading the exams for the students as much as possible.

Objective function

Find feasible solutions for different instances of a selected dataset.

Project material

Choose at least two datasets from

- `http://www.cs.nott.ac.uk/~pszrq/data.htm`

References

- Schaerf, A. (1999). Survey of automated timetabling. Artificial Intelligence Review, 13(2), 87–127. `https://doi.org/10.1023/A:1006576209967`
- Almeida, J., Santos, D. R. dos, Figueira, J. R. (2022). A Multi-Objective Model for Thesis Defence Scheduling. `https://doi.org/10.48550/arxiv.2205.07727`

Chapter 16

Vehicle Routing (sets with number 1)

One of the big challenges companies face today is improving efficiency. Costs should be reduced while the quality is maintained. Making planning decisions is often computationally hard. Therefore, advanced planning methods are needed to determine the best strategy for the near and distant future. One of those hard planning problems is known as the Vehicle Routing Problem with Time Windows (VRPTW).

A fleet of identical vehicles with a fixed capacity is located at a central depot. Furthermore, there is a set of customers. Every customer has to be visited exactly once within a predefined time window by one of the vehicles. A time window consists of a release date and a due date. In transportation problems, two types of time window handling can occur.

- Time windows are called **soft** when service can start before the release date, or after the due date. However, this induces a violation cost, which penalizes early or late service.
- When service must start within the time windows, time windows are called **hard**. When a vehicle arrives at a customer before its release date, it is required to wait until the release date before service can start. A vehicle is not allowed to arrive after the due date.

Each vehicle starts and ends at the depot.

Objective function

This project deals with soft time windows, and thus has a bigger set of feasible solutions. If a vehicle arrives to a customer outside the time window, it will be penalised with a weight in the objective function according to the temporal distance from the time window. The travel time between two customers is assumed to be a stochastic variable with a given probabilistic density function around the deterministic travel time. Since no information is available, in the dataset, about travel times, the deterministic travel time has been calculated assuming an average speed.

Project material

Use the randomly generated datasets with number 1 from

- `http://w.cba.neu.edu/∼msolomon`

Due to the complexity of the problem, you may choose a subset of ten to twenty customers when there are more.

References

- Martins, M. S. E., Coito, T., Firme, B., Viegas, J., Sousa, J. M. C., Figueiredo, J., Vieira, S. M. (2020). Solving Dynamic Delivery Services Using Ant Colony Optimization. Information Processing and Management of Uncertainty in Knowledge-Based Systems, 1237, 327. `https://doi.org/10.1007/978-3-030-50146-4_25`

Chapter 17

Vehicle Routing (sets with number 2)

One of the big challenges companies face today is improving efficiency. Costs should be reduced while the quality is maintained. Making planning decisions is often computationally hard. Therefore, advanced planning methods are needed to determine the best strategy for the near and distant future. One of those hard planning problems is known as the Vehicle Routing Problem with Time Windows (VRPTW).

A fleet of identical vehicles with a fixed capacity is located at a central depot. Furthermore, there is a set of customers. Every customer has to be visited exactly once within a predefined time window by one of the vehicles. A time window consists of a release date and a due date. In transportation problems, two types of time window handling can occur.

- Time windows are called **soft** when service can start before the release date, or after the due date. However, this induces a violation cost, which penalizes early or late service.
- When service must start within the time windows, time windows are called **hard**. When a vehicle arrives at a customer before its release date, it is required to wait until the release date before service can start. A vehicle is not allowed to arrive after the due date.

Each vehicle starts and ends at the depot.

Objective function

This project deals with soft time windows, and thus has a bigger set of feasible solutions. If a vehicle arrives to a customer outside the time window, it will be penalised with a weight in the objective function according to the temporal distance from the time window. The travel time between two customers is assumed to be a stochastic variable with a given probabilistic density function around the deterministic travel time. Since no information is available, in the dataset, about travel times, the deterministic travel time has been calculated assuming an average speed.

Project material

Use the randomly generated datasets with number 2 from

- `http://w.cba.neu.edu/~msolomon`

Due to the complexity of the problem, you may choose a subset of ten to twenty customers when there are more.

References

- Martins, M. S. E., Coito, T., Firme, B., Viegas, J., Sousa, J. M. C., Figueiredo, J., Vieira, S. M. (2020). Solving Dynamic Delivery Services Using Ant Colony Optimization. Information Processing and Management of Uncertainty in Knowledge-Based Systems, 1237, 327. `https://doi.org/10.1007/978-3-030-50146-4_25`

Chapter 18

Travelling Salesman (Djibouti, Zimbabwe)

The Travelling Salesman Problem (TSP) is often represented as a graph, where the cities are nodes, and the distances between them are edges. The problem is to find the shortest Hamiltonian cycle, which is a cycle that visits each node exactly once and returns to the starting node.

The TSP is a combinatorial optimization problem, which means that the number of possible solutions grows exponentially with the number of cities. This makes it very difficult to solve for large instances of the problem, and it is therefore considered an NP-hard problem.

Objective function

Minimize the total distance or cost required to visit all the given cities exactly once and return to the starting city.

Project material

Use the Djibouti and Zimbabwe datasets from

- `https://www.math.uwaterloo.ca/tsp/world/countries.html`

You may choose additional countries with more cities if you so desire.

Reference

- Laporte, G. (1992). The traveling salesman problem: An overview of exact and approximate algorithms. European Journal of Operational Research, 59(2), 231–247. `https://doi.org/10.1016/0377-2217(92)90138-Y`

Chapter 19

Travelling Salesman (Qatar, Uruguay)

The Travelling Salesman Problem (TSP) is often represented as a graph, where the cities are nodes, and the distances between them are edges. The problem is to find the shortest Hamiltonian cycle, which is a cycle that visits each node exactly once and returns to the starting node.

The TSP is a combinatorial optimization problem, which means that the number of possible solutions grows exponentially with the number of cities. This makes it very difficult to solve for large instances of the problem, and it is therefore considered an NP-hard problem.

Objective function

Minimize the total distance or cost required to visit all the given cities exactly once and return to the starting city.

Project material

Use the Qatar and Uruguay datasets from

- `https://www.math.uwaterloo.ca/tsp/world/countries.html`

You may choose additional countries with more cities if you so desire.

Reference

- Laporte, G. (1992). The traveling salesman problem: An overview of exact and approximate algorithms. European Journal of Operational Research, 59(2), 231–247. `https://doi.org/10.1016/0377-2217(92)90138-Y`

Chapter 20

Travelling Salesman (Western Sahara, Oman)

The Travelling Salesman Problem (TSP) is often represented as a graph, where the cities are nodes, and the distances between them are edges. The problem is to find the shortest Hamiltonian cycle, which is a cycle that visits each node exactly once and returns to the starting node.

The TSP is a combinatorial optimization problem, which means that the number of possible solutions grows exponentially with the number of cities. This makes it very difficult to solve for large instances of the problem, and it is therefore considered an NP-hard problem.

Objective function

Minimize the total distance or cost required to visit all the given cities exactly once and return to the starting city.

Project material

Use the Western Sahara and Oman datasets from

- `https://www.math.uwaterloo.ca/tsp/world/countries.html`

You may choose additional countries with more cities if you so desire.

Reference

- Laporte, G. (1992). The traveling salesman problem: An overview of exact and approximate algorithms. European Journal of Operational Research, 59(2), 231–247. `https://doi.org/10.1016/0377-2217(92)90138-Y`

Chapter 21

Symmetric Travelling Salesman (Heidelberg Database)

The Travelling Salesman Problem (TSP) is often represented as a graph, where the cities are nodes, and the distances between them are edges. The problem is to find the shortest Hamiltonian cycle, which is a cycle that visits each node exactly once and returns to the starting node.

The TSP is a combinatorial optimization problem, which means that the number of possible solutions grows exponentially with the number of cities. This makes it very difficult to solve for large instances of the problem, and it is therefore considered an NP-hard problem.

Objective function

Minimize the total distance or cost required to visit all the given cities exactly once and return to the starting city.

Project material

Choose at least two symmetric TSP datasets from

- `http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/`

Begin with a dataset of about 100 points, and another with at least 200.

Reference

- Laporte, G. (1992). The traveling salesman problem: An overview of exact and approximate algorithms. European Journal of Operational Research, 59(2), 231–247. `https://doi.org/10.1016/0377-2217(92)90138-Y`

Chapter 22

Asymmetric Travelling Salesman (Heidelberg Database)

The Asymmetric Traveling Salesman Problem (ATSP) is a variant of the classic Traveling Salesman Problem (TSP) where the distance between any two cities is not necessarily the same in both directions. In other words, the ATSP considers a directed graph, where the distance from city A to city B may be different than the distance from city B to city A.

The objective of the ATSP is to find the shortest Hamiltonian cycle that visits each city exactly once and returns to the starting city. This problem is known to be NP-hard, which means that finding an optimal solution for large instances of the problem is computationally very difficult.

Objective function

Minimize the total distance or cost required to visit all the given cities exactly once and return to the starting city.

Project material

Choose at least two ATSP datasets from

- `http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/`

Begin with a dataset of about 50 points, and another with at least 100.

Reference

- Roberti, R., Toth, P. (2012). Models and algorithms for the Asymmetric Traveling Salesman Problem: an experimental comparison. EURO Journal on Transportation and Logistics, 1(1–2), 113–133. `https://doi.org/10.1007/S13676-012-0010-0`

Chapter 23

Sequential Ordering (Heidelberg Database)

The Sequential Ordering Problem (SOP) is an asymmetric traveling salesman problem with additional constraints. Given a set of n nodes and distances for each pair of nodes, find a Hamiltonian path from node 1 to node n of minimal length which takes given precedence constraints into account. Each precedence constraint requires that some node i has to be visited before some other node j.

Objective function

Minimize the total distance or cost required to visit all the given cities exactly once and return to the starting city.

Project material

Choose at least two SOP datasets from

- `http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/`

Begin with a dataset of about 50 points, and another with at least 100.

Reference

- Hernádvölgyi, I. T. (2004). Solving the Sequential Ordering Problem with Automatically Generated Lower Bounds. Operations Research Proceedings, 355–362. https://doi.org/10.1007/978-3-642-17022-5_46

Chapter 24

Capacitated Vehicle Routing (Heidelberg Database)

Given $n - 1$ nodes, one depot and distances from the nodes to the depot, as well as between nodes. All nodes have demands which can be satisfied by the depot. For delivery to the nodes, trucks with identical capacities are available. The problem is to find tours for the trucks of minimal total length that satisfy the node demands without violating truck capacity constraint. The number of trucks is not specified. Each tour visits a subset of the nodes and starts and terminates at the depot. (In some data files a collection of alternate depots is given. A capacitated vehicle routing problem (CVRP) is then given by selecting one of these depots.)

Objective function

Find tours for the trucks of minimal total length that satisfy the node demands without violating truck capacity constraint.

Project material

Choose at least two CVRP datasets from

- `http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/`

Begin with a dataset of about 50 points, and another with at least 100.

Reference

- Feld, S., Roch, C., Gabor, T., Seidel, C., Neukart, F., Galter, I., Mauerer, W., Linnhoff-Popien, C. (2019). A hybrid solution method for the capacitated vehicle routing problem using a quantum annealer. Frontiers in ICT, 6, 13. `https://doi.org/10.3389/fict.2019.00013`

Chapter 25

Knapsack (Burkardt datasets)

The Knapsack Problem (KP) is a combinatorial problem. KP has been studied for more than a century, with early works dating as far back as 1897. The decision problem form of the knapsack problem (can a value of at least V be achieved without exceeding the weight W?) is NP-complete, thus there is no known algorithm both correct and fast (polynomial-time) in all cases. There are many variations of the knapsack problem that have arisen from the vast number of applications of the basic problem. The main variations occur by changing the number of some problem parameter such as the number of items, number of objectives, or even the number of knapsacks. Knapsack problems appear in real-world decision-making processes in a wide variety of fields, such as finding the least wasteful way to cut raw materials, selection of investments and portfolios, selection of assets for asset-backed securitization, etc..

Common to all versions are a set of n items, with each item $1 \leq j \leq n$ having an associated profit p_j, and weight w_j. The binary decision variable x_j is used to select the item.

Objective function

The objective is to pick some of the items, with maximal total profit, while obeying that the maximum total weight of the chosen items must not exceed total capacity W. Generally, these coefficients are scaled to become integers, and they are almost always assumed to be positive.

Project material

Use all the datasets from

- `https://people.sc.fsu.edu/~jburkardt/datasets/knapsack_01/knapsack_01.html`

Reference

- Dudziński, K., Walukiewicz, S. (1987). Exact methods for the knapsack problem and its generalizations. European Journal of Operational Research, 28(1), 3–21. `https://doi.org/10.1016/0377-2217(87)90165-2`

Chapter 26

Multiple Knapsack (Burkardt datasets)

A Multiple Knapsack Problem (MKP) differs from the original KP (described in project proposal 25) as there are a number of available knapsacks with different capacities.

Project material

Choose at least two datasets (a low-dimensional one and large-scale one) from

- https://people.sc.fsu.edu/~jburkardt/datasets/knapsack_multiple/knapsack_multiple.html

Reference

- Dudziński, K., Walukiewicz, S. (1987). Exact methods for the knapsack problem and its generalizations. European Journal of Operational Research, 28(1), 3–21. https://doi.org/10.1016/0377-2217(87)90165-2

Chapter 27

Max Cut (Adriaensen datasets)

The Maximum Cut Problem (MCP) is a classic combinatorial optimization problem that involves partitioning the vertices of a graph into two disjoint sets such that the number of edges that cross the partition is maximized. In other words, the objective is to find a cut that separates the vertices into two groups such that the number of edges that connect vertices in different groups is as large as possible.

MCP has many real-world applications in various fields such as social network analysis, image segmentation, and network design. For example, in social network analysis, the problem can be used to identify communities of individuals with similar interests or behaviours.

Objective function

The objective is to find a cut that separates the vertices into two groups such that the number of edges that connect vertices in different groups is as large as possible.

Project material

Use the MCP datasets from

- `https://github.com/Steven-Adriaensen/hyflext`

Reference

- Adriaensen, S., Ochoa, G., Nowe, A. (2015). A benchmark set extension and comparative study for the HyFlex framework. 2015 IEEE Congress on Evolutionary Computation, CEC 2015 - Proceedings, 784–791. `https://doi.org/10.1109/CEC.2015.7256971`, `http://www.cs.stir.ac.uk/~goc/papers/HyFlexExt_CEC2015.pd`

Chapter 28

Max Cut (Optsicom datasets)

The Maximum Cut Problem (MCP) is a classic combinatorial optimization problem that involves partitioning the vertices of a graph into two disjoint sets such that the number of edges that cross the partition is maximized. In other words, the objective is to find a cut that separates the vertices into two groups such that the number of edges that connect vertices in different groups is as large as possible.

MCP has many real-world applications in various fields such as social network analysis, image segmentation, and network design. For example, in social network analysis, the problem can be used to identify communities of individuals with similar interests or behaviours.

Objective function

The objective is to find a cut that separates the vertices into two groups such that the number of edges that connect vertices in different groups is as large as possible.

Project material

Choose at least two datasets from

- `https://grafo.etsii.urjc.es/optsicom/maxcut.html`

Reference

- Adriaensen, S., Ochoa, G., Nowe, A. (2015). A benchmark set extension and comparative study for the HyFlex framework. 2015 IEEE Congress on Evolutionary Computation, CEC 2015 - Proceedings, 784–791. `https://doi.org/10.1109/CEC.2015.7256971`, `http://www.cs.stir.ac.uk/~goc/papers/HyFlexExt_CEC2015.pd`

Chapter 29

Quadratic Assignment (Taillard datasets)

The Quadratic Assignment Problem (QAP) is a classic combinatorial optimization problem that involves assigning a set of facilities to a set of locations, subject to various constraints. The objective is to minimize the total cost of the assignment, which is calculated based on the distances between the facilities and the distances between the locations.

The QAP has many real-world applications in various fields such as operations research, facility layout design, and manufacturing. For example, in facility layout design, the problem can be used to determine the optimal arrangement of machines on a factory floor to minimize the total cost of moving materials and products between machines.

Objective function

Minimize the total cost of the assignment.

Project material

Choose at least two datasets from

- `http://mistic.heig-vd.ch/taillard/problemes.dir/qap.dir/qap.html`

Reference

- Adriaensen, S., Ochoa, G., Nowe, A. (2015). A benchmark set extension and comparative study for the HyFlex framework. 2015 IEEE Congress on Evolutionary Computation, CEC 2015 - Proceedings, 784–791. `https://doi.org/10.1109/CEC.2015.7256971`, `http://www.cs.stir.ac.uk/~goc/papers/HyFlexExt_CEC2015.pd`

Chapter 30

Bin Packing (Beasley datasets)

Packing of items into boxes or bins is a recurring task in production and distribution. A large variety of different packing problems can be distinguished, depending on the size and shape of items, as well as the form and capacity of bins. This type of problems can be found in the many applications, such as in the steel cutting process or the scheduling of parallel processors.

The bin packing problem (BPP) can be informally defined in a very simple way. We are given n items, each having an integer weight $w_j,\ j = 1, \ldots, n$, and an unlimited number of identical bins of integer capacity c.

Objective function

The objective is to pack all the items into the minimum number of bins so that the total weight packed in any bin does not exceed the capacity.

A BPP can be easily transformed into an equivalent Cutting Stock Problem (described in project proposal 31) and vice-versa.

Project material

Choose at least two datasets, one with 10 and another with at least 100 bins, from

- `https://people.brunel.ac.uk/~mastjjb/jeb/orlib/binpacktwoinfo.html`

Reference

- Delorme, M., Iori, M., Martello, S. (2016). Bin packing and cutting stock problems: Mathematical models and exact algorithms. European Journal of Operational Research, 255(1), 1–20. `https://doi.org/10.1016/J.EJOR.20`

Chapter 31

Cutting Stock (Beasley datasets)

Packing of items into boxes or bins is a recurring task in production and distribution. A large variety of different packing problems can be distinguished, depending on the size and shape of items, as well as the form and capacity of bins. This type of problems can be found in the many applications, such as in the steel cutting process or the scheduling of parallel processors.

The cutting stock problem (CSP) is a similar problem in which we are given m item types, each having an integer weight w_j and an integer demand d_j, $j = 1, \ldots, m$, and an unlimited number of identical bins (frequently called rolls in the literature) of integer capacity c.

Objective function

The objective is to produce d_j copies of each item type j using the minimum number of bins so that the total weight packed in any bin does not exceed its capacity.

A CSP can be easily transformed into an equivalent Bin Packing Problem (described in project proposal 30) and vice-versa.

Project material

Choose at least two datasets, one with 10 and another with at least 100 bins, from

- `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

Reference

- Delorme, M., Iori, M., Martello, S. (2016). Bin packing and cutting stock problems: Mathematical models and exact algorithms. European Journal of Operational Research, 255(1), 1–20. `https://doi.org/10.1016/J.EJOR.2`