Linear Programming

Guidelines

- Answers to go in **answers.docx**. The document should be submitted in a pdf format. Clearly and succinctly answer the questions.
- Python implementation to go in **answers.py**. In order to get full marks for the implementation parts, your code should be clear, well commented, sensibly structured and correct. As indicated per the question, marks may be deducted if the linear problems are not sensibly formulated.
- The results reported in **answers.docx**. must be verifiable in the code to get full marks.

Electricity market dispatch problem

Your task is to use linear programming to dispatch an electricity market. In this assignment, you will formulate and implement two electricity market designs: a one-sided market (e.g., Australian market); and a two-sided market (e.g., European and U.S. markets).

One-sided market:

- The objective of the one-side market is to accept the “cheapest” generation bids so that generation equals load. In this market design, only generators bid in the market;
- Generation bids consist of price and quantity pairs, as illustrated in Table 1. For instance, the bid of generator 0 indicates that it is willing to supply up to 60 MW as long it gets paid at least 39 $/MW for doing so;
- The total system load to be supplied by the generators is equal to 150 MW;
- The market may partially accept a bid, i.e. select a quantity between zero and the quantity being offered at a price;
- The cost of a generation bid is the bid price times the accepted quantity of that bid. The overall market objective is to minimise the sum of generation bid costs;

Two-sided market:

- The objective of the two-side market is to accept the “cheapest” generation and load bids so that generation equals load. In this market design, generators and loads bid in the market;
- Generation and load bids consist of price and quantity pairs, as illustrated in Table 1 and Table 2;

- The market may partially accept a bid, i.e. select a quantity between zero and the quantity being offered at a price;
- The cost of a generation bid is the bid price times the accepted quantity of that bid. The revenue of a load bid is the bid price times the accepted quantity of that bid. The overall market objective is to minimise the sum of bid net costs;

Table 1. Generation bids.

Generator	0	1	2	3	4	5
Price ($/MW)	39	35	29	58	9	9
Quantity (MW)	60	50	20	80	10	10

Table 2. Load bids.

Load	0	1	2	3	4	5
Price ($/MW)	40	45	58	55	48	35
Quantity (MW)	27	30	24	45	30	24

Questions [100 marks]

1. Formulate and implement the linear problem for the one-sided market. The linear problem should be formulated in the **answers.docx** and implemented in the **answers.py** (in **function_1**). **[15 marks]** for problem formulation and **[15 marks]** for problem implementation;
2. Formulate and implement the linear problem for the two-sided market. The linear problem should be formulated in the **answers.docx** and implemented in the **answers.py** (in **function_2**). **[15 marks]** for problem formulation and **[15 marks]** for problem implementation;
3. Report in **answers.docx** the following information: 3.1) accepted bid quantiles in the one-sided and two-sided markets **[5 marks]**; 3.2) objective value of one-sided and two-sided markets **[5 marks]**; 3.3) marginal prices[1] of one-sided and two-sided markets **[5 marks]**; 3.4) Discuss which is the most cost-effective market design based on the results (only in **answers.docx**) **[5 marks]**;
4. The dispatch of the electricity market is constrained by the electricity network illustrated in Figure 1. The electricity network limits the power flows between generators and loads through constraints (1)-(2). Constraint (1) defines the branch power flows, where G_n is the total generation in bus n (in MW), L_n is the total load in bus n (in MW), θ_n is the voltage phase angle (free variable), $x_{n,m}$ is the reactance, $n \in N$ is the set of buses, and $n \rightarrow m$ defines all the branches that leave from bus n to bus m. It can be considered that the unit of the right side of equation 1 is MW. Constraint (2) defines the branch power flow limits, where $\overline{B_{n,m}}$ is the branch limit (in MW) and $(n, m) \in M$ is the branches.

$$G_n - L_n = \sum_{n \rightarrow m} \frac{\theta_n - \theta_m}{x_{n,m}}, \quad \forall\ n \in N \tag{1}$$

[1] After the market dispatch, generators and loads are paid and charged for their accepted quantity at the marginal price, respectively. This marginal price is a dual variable of your linear problem, associated with the constraint that enforces generation equal to load;

$$-\overline{B_{n,m}} \leq \frac{\theta_n - \theta_m}{x_{n,m}} \leq \overline{B_{n,m}}, \ \forall \ (n, m) \in M \tag{2}$$

Implement the linear network constraints (1)-(2) in the **answers.py** (in **function_2**) using the data from Table 3 **[10 marks].** The network constraints should be implemented in their generalised form to obtain 10 marks. Report in **answers.docx** the following information: 4.1) the new accepted bid quantiles and the branch power flows **[5 marks];** 4.2.) Explain the impact of network constraints on the marginal price and accepted bids **[5 marks].**

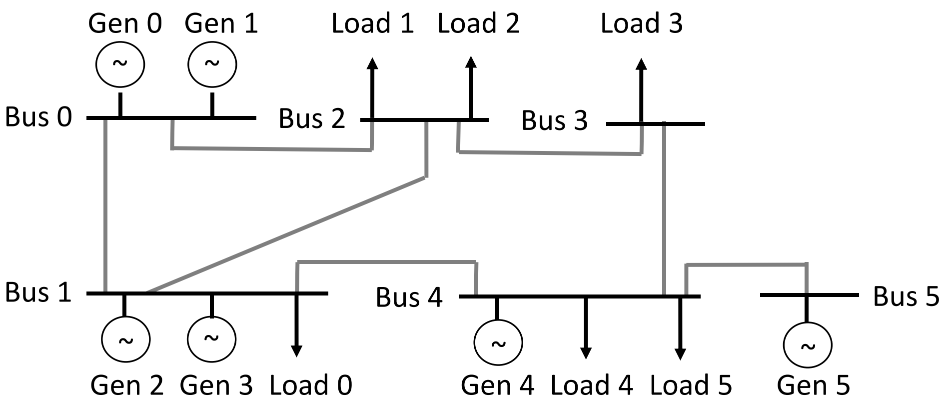

Figure 1. Electricity network (Gen – generator, grey lines – branches).

Table 3. Branch network data.

From bus	To bus	Reactance	Branch limit (MW)
0	1	0.06	40
0	2	0.24	40
1	2	0.18	40
1	4	0.12	40
2	3	0.03	40
3	4	0.24	40
4	5	0.03	40

Further notes and tips:

- Follow the order of the questions to implement and run the linear problems;
- The **answers.py** provides commented code to help the student structure the code. The areas to be coded by the student are **<to be completed by the student>** and **<to be replaced by the student>**. In the main function, the student only needs to activate function_1 or function_2 to run the experiments. The student does not need to code in the main function;
- The results should not have more than two decimal places;
- Dual variables depend on how constraints are written, and depend on the convention used by the solver. What this means is that you could get a result that is the negative of the quantity you actually are interested in.