Appendix II: ChatGPT Economic Dispatch Prompt & Response

This Appendix includes the prompt supplied to ChatGPT for the economic dispatch problem supplied and the response returned by ChatGPT. This exercise was carried out using ChatGPT Plus in June 2023, with access to the Wolfram plug-in.

Prompt

I am going to ask you to write code which may be used to solve an optimisation problem in GAMS. The optimisation problem involves the economic dispatch of a simple power system consisting of six generators (G1, G2, G3, G4, G5, G6, G7, G8, G9). Each unit may be assumed to be operational and online.

The cost curves for each of the generators may be described as C_i = a_i*P_i^2+ b_i*P_i+c_i, where C_i is the cost associated with running generator i with output P_i MW. The parameters a_i, b_i and c_i vary for each unit i and are described below. Each unit also has different minimum and maximum outputs, P_min,i and P_max,i. These minimum and maximum parameters are also described in the table below.

Unit	P_min,i	P_max,i	a_i	b_i	c_i
G1	2.4	12	0.025	25.5	24.4
G2	4	20	0.012	37.6	117.8
G3	0	50	0	0.5	0
G4	15.2	76	0.009	13.3	81.1
G5	25	100	0.006	8	217.9
G6	54.24	155	0.005	10.7	142.7
G7	68.95	197	0.003	23	259.1
G8	140	350	0.002	10.9	177.1
G9	100	400	0.00211	7.5	311.9

The objective of this optimisation problem is to minimise the cost of running the power system. The constraints acting on the problem are the individual unit minimum and maximum outputs, P_min,i and P_max,i. The output of each unit max may not lie outside the bounds [P_min,i,P_max,i]. The other constraint is that the sum of outputs from all units must equal demand. The demand in the system is 1495.2 MW.

Now please write the GAMS code which may be used to solve this economic dispatch optimisation problem. Note that the I want to report the system marginal price and the output of each generator in GAMS after the problem is solved. After writing the GAMS code, I would like you to formulate the optimisation problem in terms of an objective function and a set of constraints. Finally, I would like you to form a Lagrangian function for the optimisation problem and describe how it may be used to solve the problem by hand. Remember that you must include equality and inequality constraints in the Lagrangian function.

Response

Sure, let's start with the GAMS code for the economic dispatch problem. The code will define the parameters, variables, objective function, and constraints, and then solve the problem.

```
SETS

    i 'generators' / G1*G9 /;

ALIAS (i,j);

PARAMETERS

    P_min(i) 'minimum output' / G1 2.4, G2 4, G3 0, G4
    15.2, G5 25, G6 54.24, G7 68.95, G8 140, G9 100 /
    P_max(i) 'maximum output' / G1 12, G2 20, G3 50,
    G4 76, G5 100, G6 155, G7 197, G8 350, G9 400 /
    a(i) 'cost curve parameter a' / G1 0.025, G2 0.012, G3 0, G4
    0.009, G5 0.006, G6 0.005, G7 0.003, G8 0.002, G9 0.00211 /
    b(i) 'cost curve parameter b' / G1 25.5, G2 37.6, G3
    0.5, G4 13.3, G5 8, G6 10.7, G7 23, G8 10.9, G9 7.5 /
    c(i) 'cost curve parameter c' / G1 24.4, G2 117.8, G3 0, G4
    81.1, G5 217.9, G6 142.7, G7 259.1, G8 177.1, G9 311.9 /
    Demand 'total demand' / 1495.2 /;

VARIABLES

    P(i) 'output of generator i'
    Cost 'total cost';

POSITIVE VARIABLE P;

EQUATIONS

    Obj 'objective function'
    DemandConstraint 'total demand constraint'
    P_min_constraint(i) 'minimum output constraint'
    P_max_constraint(i) 'maximum output constraint';

    Obj..  Cost =e= SUM(i, a(i)*sqr(P(i)) + b(i)*P(i) + c(i));

    DemandConstraint..  SUM(i, P(i)) =e= Demand;

    P_min_constraint(i)..  P(i) =g= P_min(i);

    P_max_constraint(i)..  P(i) =l= P_max(i);

MODEL EconomicDispatch / all /;

SOLVE EconomicDispatch USING NLP MINIMIZING Cost;

DISPLAY P.l, Cost.l;
```

The GAMS code above defines the economic dispatch problem. The objective function is the total cost of running the generators, which is the sum of the costs of each generator. The cost of each generator is calculated using the given cost curve parameters. The constraints are that the total output of the generators must equal the demand, and the output of each generator must be between its minimum and maximum output. Next, let's formulate the optimization problem in terms of an objective function and a set of constraints.

Objective function:

$$\text{minimize} \quad \text{Cost} = \sum_{i=\text{G1}}^{\text{G9}} (a_i P_i^2 + b_i P_i + c_i)$$

Subject to:

1. $\sum_{i=\text{G1}}^{\text{G9}} P_i = \text{Demand}$ (Total demand constraint)
2. $P_{\text{min},i} \le P_i \le P_{\text{max},i}$ for all i

Finally, let's form the Lagrangian function for the optimization problem. The Lagrangian function is used to solve constrained optimization problems and is formed by adding the product of the constraints and their associated Lagrange multipliers to the objective function. The Lagrangian function for this problem is:

$$L(P, \gamma, \mu, \nu) = \sum_{i=\text{G1}}^{\text{G9}} (a_i P_i^2 + b_i P_i + c_i) + \gamma(\sum_{i=\text{G1}}^{\text{G9}} P_i - \text{Demand}) - \sum_{i=\text{G1}}^{\text{G9}} \mu_i (P_i - P_{\text{min},i}) + \sum_{i=\text{G1}}^{\text{G9}} \nu_i (P_{\text{max},i} - P_i)$$

where:

- P is the vector of generator outputs,
- γ is the Lagrange multiplier associated with the total demand constraint,
- μ is the vector of Lagrange multipliers associated with the minimum output constraints, and
- ν is the vector of Lagrange multipliers associated with the maximum output constraints.

To solve the problem by hand using the Lagrangian function, you would set the derivative of the Lagrangian with respect to each variable (the P_i, γ, μ_i, and ν_i) equal to zero and solve the resulting system of equations. This gives you the values of the variables that minimize the objective function subject to the constraints. Note that the inequality constraints are included in the Lagrangian function as penalty terms, with the Lagrange multipliers μ_i and ν_i ensuring that the penalties are applied only when the constraints are violated. The minus sign in front of the μ_i term is because it is a lower bound constraint, and the plus sign in front of the ν_i term is because it is an upper bound constraint. The Lagrange multipliers μ_i and ν_i must be greater than or equal to zero to ensure that the constraints are satisfied.