Unit Commitment Optimization Problem

Li Linfeng 1000920 linfeng_li@mymail.sutd.edu.sg
Liu Sudian 1000909 sidian_liu@mymail.sutd.edu.sg
Wang Yiran 1000906 yiran_wang@mymail.sutd.edu.sg

1 Introduction

This project is aimed to implement and solve a small-scale unit commitment problem. We made use of the data set of a US city electricity network that is available online and used open-source solver to solve the problem.

1.1 Data Availability

We simulated unit commitment optimization problem with data sourced from the US DA electricity market. It is a test set that helps in summer electricity forecast [1].

1.2 Problem Framing

The data set is gigantic which comprises of 1011 generators and 3753 buses. Due to the limited computational power of our laptops, we decided to run the simulation on a micro-scale that includes the first 10 generators and buses. What is more, the pricing strategy provided by generators is dynamic that changes with the energy output. This increased the complication in problem formulation, therefore we computed and used the weighted average of the operating cost instead. In addition, we removed reserve constraints to further simplify the problem.

2 Method

2.1 Formulation

The variables are defined as the following:

- t: time period
- T: max time period
- i: generator index
- N: total numbers of generators
- $S_i(t)$: power production of generator i at period t
- $C_i(P)$: cost of generator i when producing power of P
- $C_{s,i}(t)$: start-up cost of generator i

- $D(t)$: total demand
- $r_i(t)$: reserve of generator i at period t
- $u_i(t)$: state of generator i at period t, with 0 meaning off and 1 meaning on
- $\Delta_{i,U}$: ramp-up limit of generator i
- $\Delta_{i,D}$: ramp-down limit of generator i
- $t_{i,up}(t)$: up-time of generator i at period t
- $t_{i,down}(t)$: down-time of generator i at period t
- $T_{i,up}$: minimum up-time of generator i
- $T_{i,down}$: minimum down-time of generator i

The "standard" unit commitment problem is formulated as:

$$
\begin{array}{ll}
\min & \sum_{t=1}^{T}\sum_{i=1}^{N}\left[u_i(t)C_i(S_i(t)) + C_{s,i}(t)\right] \\
\text{s.t.} & \forall\ t,\ i \\
& \sum_{i=1}^{N} u_i(t)S_i(t) - D(t) = 0 \\
& R(t) - \sum_{i=1}^{N} u_i(t)r_i(t) \le 0 \\
& S_i(t) - S_i^{\max}(t) \le 0 \\
& S_i^{\min}(t) - S_i \le 0 \\
& u_i(t)u_i(t+1)[(S_i(t+1) - S_i(t))] - (\Delta_{i,U} - r_i(t+1)) \le 0 \\
& u_i(t)u_i(t+1)[-(S_i(t+1) - S_i(t))] - (\Delta_{i,D} - r_i(t+1)) \le 0 \\
& u_i(t) = 1, \text{ if } 1 \le t_{i,\mathrm{up}}(t) \le T_{i,\mathrm{up}} \\
& u_i(t) = 0, \text{ if } 1 \le t_{i,\mathrm{down}}(t) \le T_{i,\mathrm{down}} \\
& t_{i,\mathrm{up}}(t) = \begin{cases} t_{i,\mathrm{up}}(t-1) + 1 & \text{if } u_i(t-1) = 1 \\ 0 & \text{otherwise} \end{cases} \\
& t_{i,\mathrm{down}}(t) = \begin{cases} t_{i,\mathrm{down}}(t-1) + 1 & \text{if } u_i(t-1) = 0 \\ 0 & \text{otherwise} \end{cases}
\end{array}
\tag{1}
$$

where the start-up cost is given as

$$
C_{s,i}(t) = \begin{cases} C_{i,\text{start-up}} & \text{if } u_i(t) - u_i(t-1) > 0, \\ 0 & \text{otherwise.} \end{cases}
\tag{2}
$$

In this project, the reserve is ignored. The formulation without reserve is given as:

$$\begin{array}{ll}
\min & \sum_{t=1}^{T}\sum_{i=1}^{N}\left[u_i(t)C_i(S_i(t))+C_{s,i}(t)\right] \\
\text{s.t.} & \forall\ t,\ i \\
& \sum_{i=1}^{N} u_i(t)S_i(t)-D(t)=0 \\
& S_i(t)-S_i^{\max}(t)\le 0 \\
& S_i^{\min}(t)-S_i\le 0 \\
& u_i(t)u_i(t+1)[(S_i(t+1)-S_i(t))]-(\Delta_{i,U})\le 0 \\
& u_i(t)u_i(t+1)[-(S_i(t+1)-S_i(t))]-(\Delta_{i,D})\le 0 \\
& u_i(t)=1,\ \text{if } 1\le t_{i,\text{up}}(t)\le T_{i,\text{up}} \\
& u_i(t)=0,\ \text{if } 1\le t_{i,\text{down}}(t)\le T_{i,\text{down}} \\
& t_{i,\text{up}}(t)=\begin{cases} t_{i,\text{up}}(t-1)+1 & \text{if } u_i(t-1)=1 \\ 0 & \text{otherwise}\end{cases} \\
& t_{i,\text{down}}(t)=\begin{cases} t_{i,\text{down}}(t-1)+1 & \text{if } u_i(t-1)=0 \\ 0 & \text{otherwise}\end{cases}
\end{array} \tag{3}$$

However, the above formulation cannot be properly handled by the solver. There are many "if" conditions which cannot be accepted by the solver. The "if" conditions must be replaced.

Up and down time are defined by the following constraints:

$$t_{i,\text{up}}(t)=\begin{cases} t_{i,\text{up}}(t-1)+1 & \text{if } u_i(t-1)=1, \\ 0 & \text{otherwise.}\end{cases} \tag{4}$$

$$t_{i,\text{down}}(t)=\begin{cases} t_{i,\text{down}}(t-1)+1 & \text{if } u_i(t-1)=0, \\ 0 & \text{otherwise.}\end{cases} \tag{5}$$

They are replaced by the following constraints:

$$t_{i,up}(t)=(t_{i,up}(t-1)+1)u_i(t-1), \tag{6}$$

$$t_{i,down}(t)=(t_{i,down}(t-1)+1)(1-u_i(t-1)). \tag{7}$$

Up and down time constrain the state of the generators through the following constraints:

$$u_i(t)=1,\ \text{if } 1\le t_{i,\text{up}}(t)\le T_{i,\text{up}}, \tag{8}$$

$$u_i(t)=0,\ \text{if } 1\le t_{i,\text{down}}(t)\le T_{i,\text{down}}. \tag{9}$$

$$\tag{10}$$

They are changed to:

$$(u_i(t-1)-u_i(t))(t_{i,up}(t-1)-T_{i,up})\ge 0, \tag{11}$$

$$-(u_i(t-1)-u_i(t))(t_{i,down}(t-1)-T_{i,down})\ge 0. \tag{12}$$

The start-up cost is defined as:

$$C_{s,i}(t)=\begin{cases} C_{i,\text{start-up}} & \text{if } u_i(t)-u_i(t-1)>0, \\ 0 & \text{otherwise.}\end{cases} \tag{13}$$

At first, we attempted to change it to

$$C_{s,i}(t) = C_{i,start-up}\ \max(u_i(t) - u_i(t-1), 0). \tag{14}$$

However, the solver cannot handle the operation max. Hence, we define a auxiliary variable $w_i(t)$ under constraints

$$w_i(t) \geq u_i(t) - u_i(t-1), \tag{15}$$
$$w_i(t) \geq 0. \tag{16}$$

Then the start-up cost is defined as

$$C_{s,i}(t) = w_i(t) C_{i,start-up} \tag{17}$$

The optimal solution under the constraints of the auxiliary variables would be the same as the optimal solution of the original formulation. The solver can accept the new formulation.

2.2 Solving the Optimization Problem

Optimization software Julia was used as a platform for our MILP. It provides handy packages that were able to solve complex and multi-variables problems. Specifically, we used JuMP and CoinOptServices, the first targets algorithm optimization and the second solves mixed-integer and nonlinear problem efficiently.

Bonmin solver is selected to solve this problem. The solver regarded the problem as a mixed-integer nonlinear program. The solver tries to relax the integer constraints through linear programming relaxation, which would change $u \in \{0,1\}$ to $u \in [0,1]$. The solve also uses branch and bound methods.

3 Result

The following tables (Figure 1) showed the UC results on 4 to 7 generators regarding their on/off status (u(i,t)) and energy output amount (S(i,t)) from 1 to 24 hours planning. We were intended to solve ten generators UC scheduling, however linear program relaxation was too expensive for ten generators. The Julia platform was either incapable to reach the constraints/variable limits or the running time is too long to solve. The demand requirements were low for the first twelve hours while the demand increased in the other 12 hours, and the generators utilization has reflected this pattern. As the plot on solving time v.s. Generators showed (Figure 2), the solving time was not proportional to the scale of planned generators. However, the solving time was replying on the attributes of each generators. For example, the first four generators have similar capacity and operating costs while the fifth generator has a significant lower operation cost. Hence, the problem become easier to solve in a five generator scheduling scenario.

u(l,t)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
G1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G4	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
S1	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6
S2	258.6	237.1	220.3	216.4	215.5	214.8	216.4	226.1	262.8	289.9	312.6	331.6	347.3	358.8	364.8	373.0	376.6	378.3	371.8	359.2	358.1	349.5	316.7	285.8
S3	274.2	258.9	247.1	229.5	233.0	232.3	234.2	244.7	277.0	300.5	317.9	332.2	344.1	353.1	357.6	364.9	367.9	369.2	362.7	350.6	350.0	343.7	316.0	291.2
S4	9.114	9.114	9.114	9.114	0.0	0.0	0.0	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114

(a) 4 generators.

u(l,t)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
G1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G4	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G5	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
S1	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6
S2	254.7	235.2	220.5	209.4	214.6	213.8	215.6	222.1	256.8	195.4	216.8	235.0	250.1	261.6	267.6	275.9	279.2	280.8	274.1	261.4	260.4	252.3	219.4	188.7
S3	270.3	253.0	239.2	228.8	233.9	233.3	235.0	241.1	275.2	215.7	234.4	249.6	261.9	271.0	275.5	282.6	285.9	287.3	281.0	269.1	268.3	261.6	233.8	209.0
S4	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114
S5	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75

(b) 5 generators.

v	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
G1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G4	1	1	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G5	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
S1	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6
S2	211.036	191.31	176.948	171.6	171.6	171.6	172.39	178.67	213.3	238.2	258.16	275.61	290.45	301.85	307.96	316.45	320.14	321.7	314.56	301.22	299.71	291.35	259.68	231.31
S3	227.998	210.91	196.75	191.4	191.4	191.4	192.19	198.44	232.68	258.51	278.59	294.52	307.12	316.34	320.73	327.66	330.53	332.07	326.18	314.87	314.6	308.05	279.18	252.0
S4	9.114	9.114	9.114	9.114	9.1	9.1	9.1	9.1	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114
S5	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75
S6	86	86	86	84.304	85.508	84.129	86	86	86	86	86	86	86	86	86	86	86	86	86	86	86	86	86	86

(c) 6 generators

u(l,t)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
G1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
G4	1	1	1	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G5	1	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G7	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
S1	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6	171.6
S2	190.147	191.18	176.928	171.6	171.6	171.6	172.39	187.12	215.68	171.6	176.81	171.6	183.63	194	199.32	207.11	210.42	211.98	205.39	192.95	192.02	185.31	171.6	277.74
S3	209.991	211.04	196.769	191.4	191.4	191.4	192.2	206.85	235.11	191.4	196.55	191.4	203.45	213.69	218.87	226.51	229.75	231.3	224.85	212.64	211.79	205.14	191.4	191.4
S4	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114	9.114
S5	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75
S6	86	86	86	75.19	77.758	84.129	86	86	81.191	48.105	77.789	82.631	86	86	86	86	86	86	86	86	86	84.456	51.37	81.054
S7	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896	38.896

(d) 7 generators

Figure 1: Results of different numbers of generators.

4 Discussion

The unit commitment problem carries several complications in its nature. Firstly, the scale of the electricity is large and the number of variables and constraints are numerous. Solving this problem is a time consuming task. Secondly, mix integer programing is a NP-hard problem. The cutting edge algorithms such as branch and cut, Lagrangian relaxation still have their limitations and sometimes do not give satisfying results. Our solution is a simplified version of an implementation of these methods and by all means have space of improvement. However, in the process of problem formulation and looking for suitable solvers, we improved our comprehension of mixed-integer programing and the key constraints at stake in solving this kind of network problem. It extended our knowledge in both maths and electricity market application.

The Unit Commitment problems will become more complex by taking other constraints such as reservation and greenhouse gas emission constraints. Hence, it is essential for us to research on a simpler solving method - MILP. The multiplication of multiple integer variables

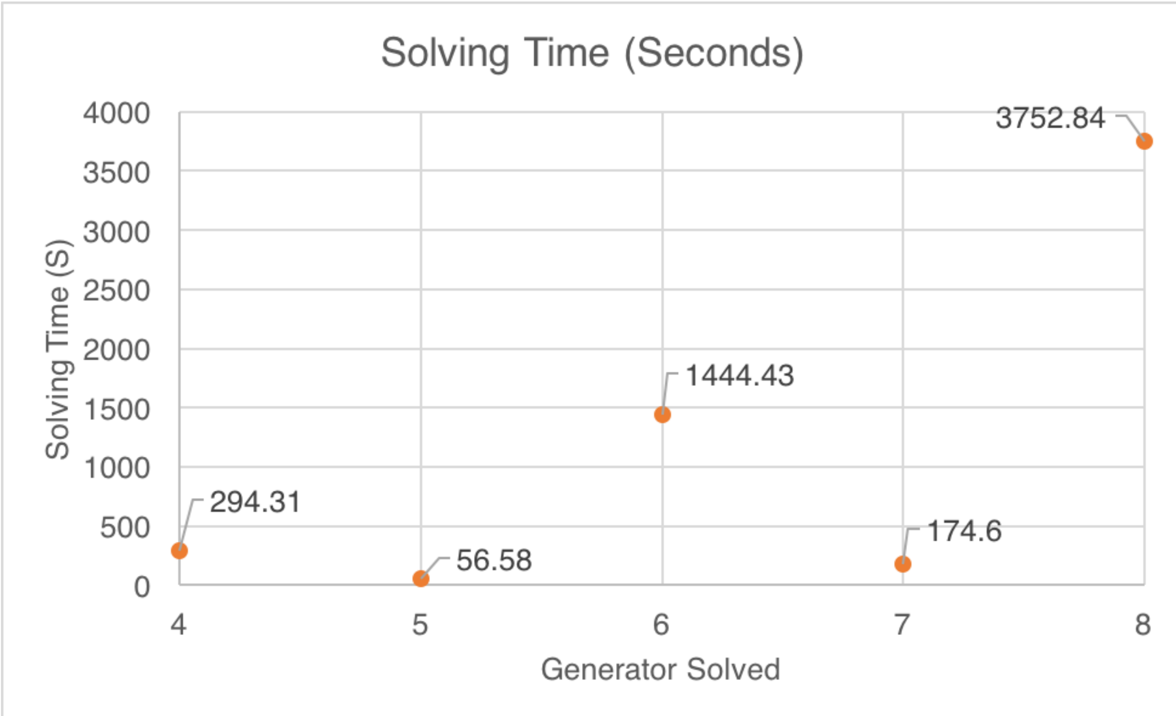

Figure 2: Solving time.

and a single continuous variable shall be regarded as linear. Moreover, the linear program relaxation method used by the solver appears not suitable for this problem since it increases the possible values of variables tremendously.

References

[1] Federal Energy Regulatory Commission. Text size small medium largerto unit commitment test system. URL https://www.ferc.gov/industries/electric/indus-act/market-planning/rto-commit-test.asp.