Mixed Integer Linear Programming in Unit Commitment

Aryan Agal
Department of Energy Science and Energy Engineering
India Institute of Technology, Bombay
Email: aryanagal@iitb.ac.in

***Abstract*—With the recent development of solution techniques, there is a trend to tackle the UC problem by using MILP approaches. This article goes over the unit commitment problem, the various constraints involved in it, and its mixed integer linear programming formulation.**

***Index Terms*—Unit Commitment, Mixed-Integer Linear Programming, High-Dimenional Constrained Optimization**

I. Introduction

The Unit Commitment(UC) problem is a mathematical optimization problem, inherent to power generation, where a given set of electrical generators are to be operated in order to meet the load demand(the forecasted load demand, to be accurate), while optimizing towards an objective (which is generally economic). Besides achieving minimum total production cost, a generation schedule needs to satisfy a number of operating constraints. Generally, unit commitment is planned as per the availability of load forecasts: this may be hourly, and in recent times, every quarter-hourly. The unit commitment problem is generally a difficult one, since: (a) There are a large number of connected generators, (b) Generating units may be of many types, each coming with their own sets of constraints and different cost curves, (c) Generation is over a large geographical area and can affect the response of the electrical grid, and checking whether the load can be sustained and what the losses are requires highly complex power flow computations.

The exact solution to the problem can be obtained by complete enumeration, which cannot be applied to realistic power systems due to its computational burden. Over the past five decades, multiple algorithms have been invented and/or applied to the UC problem.[1] These methods can be divided into three main categories: classical/deterministic, non-classical/stochastic approaches and hybrid techniques based on classical and non-classical approaches.

In general, the UC problem falls into the category of large-scale and non-convex problems that are extremely difficult to solve in an accurate and efficient way.[2] Up to date, major approaches that used for solving UC problems include dynamic programming, decomposition, nonlinear programming, mixed integer linear programming, and meta-heuristic-based methods.The major limitations on the convergence of these methods are the size of the problem or its discrete nature.

Elements of the unit-commitment problem include a set of generating units with their cost/emission curves, a load profile to be satisfied, reliability constraints, financial constraints and time horizon along which decisions have to be made. The constraints that the decisions must meet include system requirements such as load balance and spinning reserve and unit limitations including generation and ramp-rate limits.

II. Mixed Integer Programming

Mixed Integer Programming (MIP) or MILP is often posed in the form[3]:

$$\min_x cx$$

subject to:

$$Ax \leq B$$

where $x \in \mathbb{Z}^n \times \mathbb{R}^p$

It can be noted that objective function and all constraints are linear. Moreover, some variables are integers, some variables are continuous. In order to completely model the UC problem as an MILP one, multiple objectives and constraints must be altered. The min uptime and downtime constraints, as well as the stepped startup and shutdown costs must be linearized. The objective function itself, being often non-linear, must be linearized piecewise, and used.

The common solution algorithm used for the UC problem is the Branch and Bound algorithm. It is a tree search algorithm, which follows three stages: (a)Branch: Pick a variable and divide the problem in two sub-problems at this variable. (e.g. if $x \in \{0, 1\}$ solve the problem with $x = 0$ and the problem $x = 1$) (b)Bound: Solve the LP-relaxation to determine the best possible objective value for the node (c)Prune: Prune the branch of the tree (i.e.the tree will not develop further in this node) if either the sub-problem is infeasible OR the best achievable objective value is worse than a known optimum.

The MATLAB® algorithm for MILP is slightly advanced [4]; it builds upon the Branch and Bound technique. It starts by LP preprocessing i.e. reducing number of variables and constraints (and sometimes may detect infeasiblity). This is followed by solving the relaxed(non-integer) LP. Some integer preprocessing is done to tighten bounds and discard inequalities. LP is further tightened using Cut-Generation and heuristics are used to get integer-feasible solutions. Finally the branch and bound algorithm is used to process the problem systematically.

The MILP formulation has gained interest recently because of the drastic improvement in numerical solution times of commercial MIP solvers [5]. The UC problem is highly suitable to be written as an MIP. It makes the problem easily and clearly accessible for adaptation. MIP has been put in use recently by ISOs in several markets including the PJM market(regional transmission organization in the United States).

III. Problem Formulation

A. Notation

N	$=$	Number of thermal units
T	$=$	Number of time steps
J	$=$	Number of segments of cost-curve
$a_{k,i}$	$=$	Coefficients of fuel cost function, for unit i
$f_i(P_i^t)$	$=$	Fuel cost of unit i at time step t, with generation output being P_t^i
u_i^t	$=$	On/off status of unit i at time step t, (binary)
p_i^t	$=$	MW output of unit i at time step t
$\underline{p_i}$	$=$	Minimum MW output of unit i
$\overline{p_i}$	$=$	Maximum MW output of unit i
R_{ui}	$=$	MW output ramp-up limit of unit i
R_{di}	$=$	MW output ramp-down limit of unit i
Ut_i	$=$	Minimum up time of unit i (hour)
Dt_i	$=$	Minimum down time of unit i (hour)
y_i^t	$=$	Startup status at time step t of unit i (binary)
z_i^t	$=$	Shutdown status at time step t of unit i (binary)
v_i^t	$=$	Cold startup status at time step t of unit i (binary)
$C_{u,i}$	$=$	Startup cost of unit i
$C_{d,i}$	$=$	Shutdown cost of unit i
$T_{cold,i}$	$=$	Cold startup hours for unit i
$T_{off,i}$	$=$	Time since unit i is off
$f_{cold,i}$	$=$	Cold start cost for unit i
$f_{hot,i}$	$=$	Hot start cost for unit i
$C_{d,i}$	$=$	Shutdown cost of unit i
U_i^0	$=$	Initial Up-time status for unit i
S_i^0	$=$	Initial Down-time status for unit i
D^t	$=$	Total system demand at time step t
R^t	$=$	Total system spinning reserve at time step t

B. System constraints

For a UC problem, we often have to look at the entire system to optimize commitment, and we must meet demands as well as the reserve and ensure that both of these are met at every time step.

1) Overall Load Balance: Tverall he total unit generation output must satisfy the system load demand requirement at each time step t, therefore

$$\sum_{i=1}^{N} p_i^t = D^t$$

2) Spinning Reserve: Spinning reserve is the term used to describe the total amount of generation available from all units synchronized (i.e., spinning) on the system, minus the present load and losses being supplied [6]. Units are required to

$$\sum_{i=1}^{N} u_i^t(\overline{p}_i - p_i^t) \geq R^t$$

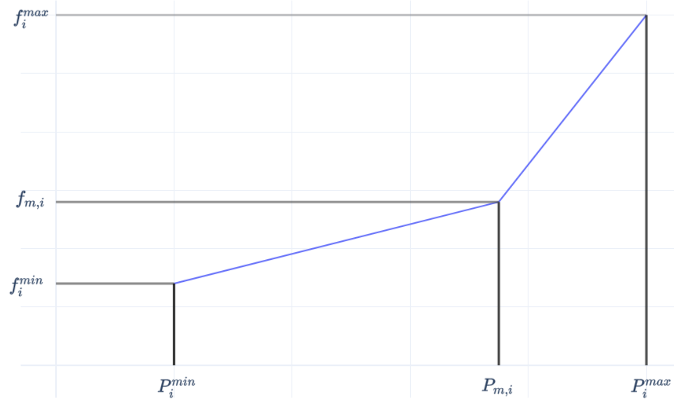

Fig. 1: Piece wise Linear Approximation of a Convex Curve

C. Unit Model

The thermal generation model describes a thermal unit. Each individual plant is considered separately, and multiple constraints are applied. This section describes the MILP formulation for this model.

1) Cost during operation: Units are generally modeled as non-linear functions, often convex quadratic functions[7]. Since MILP requires requires a linear objective function, we take multiple pieces of the function and approximate them linearly. The figure given shows one such approximation. The unit cost function can be given by:

$$f_i(P_t^i) = a_i + b_i P_t^i + c_i (P_t^i)^2$$

To approximate as a two-segment linear function, we can use the following constraints:

$$p_i^t = \underline{p_i^t} + \sum_{j=1}^{2} p_{j,i}^t$$

$$\underline{p_i^t} = \underline{p_i} u_i^t$$

$$0 \leq p_{1,i}^t \leq (p_{m,i} - \underline{p_i}) u_i^t$$

$$0 \leq p_{2,i}^t \leq (\overline{p_i} - p_{m,i}) u_i^t$$

This approach can be extended further as number of pieces required.

2) Unit Maximum/Minimum Generation Limits: For each committed unit, the power generation p_i^t should be within the generation limits of the unit, i.e. between its minimum and maximum possible generation. This can be expressed as:

$$u_i^t \underline{p_i} \leq p_i^t \leq u_i^t \overline{p_i} \ \forall i \in N, \ t \in 1...T$$

3) Unit Ramp Limits: Thermal units often are limited by the amount of stress that can be put to switch their power generation. This stress may be mechanical and/or thermal. This is formulated as:

$$p_i^{t+1} - p_i^t \leq R_{ui} \ \forall i \in N, \ t \in 1...T$$

$$p_i^t - p_i^{t+1} \leq R_{di} \ \forall i \in N, \ t \in 1...T$$

4) Unit Startup and Shutdown Constraint: We may already have a unit on/off variable, but often that is not enough. To model constraints such as minimum up-time and startup/shutdown cost, we require two new variables, denoting the startup of the unit and shutdown of the unit. We use binary variables y_i^t to denote startup and z_i^t to denote shutdown[8]. Now, a unit cannot startup and shutdown in the same time step. The startup constraints can be modelled as:

$$u_i^t - u_i^{t-1} = y_i^t - z_i^t \ \forall i \in N, \ t \in 2...T$$

$$y_i^t + z_i^t \leq 1 \ \forall i \in N, \ t \in 1...T$$

5) Unit Minimum-Up/Minimum-Down Time Constraints: Once a plant turns on, it must stay on for a certain number of hours before it can be turned off again. Similarly, once off it must stay off for a certain number of hours before it can be turned on again. These constraints can be modeled as: [9]

$$\forall i \in N, \ t \in 2...T$$

$$Minimum \ Uptime$$

$$\sum_{t=1}^{\zeta_i} 1 - u_i^t = 0,$$

$$\sum_{t=k}^{k+U_{t_i}-1} u_i^t \geq Ut_i \ y_i^k, \ \forall k = \zeta_i + 1...T - Ut_i + 1$$

$$\sum_{t=k}^{T} u_i^t - y_i^t \geq 0, \ \forall k = T - Ut_i + 2...T$$

$$\zeta_i = min\{T, (Ut_i - U_i^0) u_{i,t=0}\}$$

$$Minimum \ Downtime$$

$$\sum_{t=1}^{\eta_i} u_i^t = 0,$$

$$\sum_{t=k}^{k+D_{t_i}-1} 1 - u_i^t \geq Dt_i \ z_i^k, \ \forall k = \eta_i + 1...T - Dt_i + 1$$

$$\sum_{t=k}^{T} 1 - u_i^t - z_i^t \geq 0, \ \forall k = T - Ut_i + 2...T$$

$$\eta_i = min\{T, (Dt_i - S_i^0)[1 - u_{i,t=0}]\}$$

where U_{t_i} and D_{t_i} the corresponding minimum-up time and minimum-down time for the unit i, respectively.

6) Unit Hot/Cold Start Constraints: A recently shut-down plant generally is quicker and more efficient to start-up than a cooled one. This difference in cost can be modeled in the cost-function. We assume a step-function for the cost. If a plant is turned off within some time period, it only requires the cold start cost, else the hot-start cost. This is expressed below:

$$StartupCost = \begin{cases} hotstartcost, & \text{if down-time} \leq \text{cold start T} \\ coldstartcost, & \text{otherwise} \end{cases}$$

These constraints may be linearized.[10] To do so, we introduce an $T_{off,i}^t$ variable. To determine number of off hours:

$$T_{off,i}^t \leq M[1 - y_i^t]$$

$$|T_{off,i}^t - T_{off,i}^{t-1} - 1 + u_i^t| \leq M y_i^t$$

Now to determine if there is a cold start:

$$v_i^t \geq y_i^t - 1 - \epsilon 1 + \epsilon 2[T_{off,i}^{t-1} - T_{cold,i} + 1]$$

On the basis of a few inequalities, we can choose the values of $\epsilon 1$ and $\epsilon 2$ as:

$$\epsilon 1 = \frac{1}{N + 1 + \Delta}$$

$$\epsilon 2 = \frac{1}{N + 1}$$

where Δ is a positive number.

Other than those proposed above, unit constraints such as offline time, maintenance scheduling, security constraints etc. can also be modeled. In cases where fuel constraints are present, they can also be added and modelled linearly, although we might need to piecewise approximate.

D. Objective Function

When working on an optimization problem, we work towards maximizing or minimizing an objective function. Here the objective function is the total cost incurred and our goal is to minimize it. Hence:

$$Minimize$$

$$\sum_{t=1}^{T} \sum_{i=1}^{N} [\ f_i(p_i^t) + y_i^t C_{u,i} + z_i^t C_{d,i}\ ]$$

$$\sum_{t=1}^{T} \sum_{i=1}^{N} [\ f_i(\underline{p}_i) + \sum_{j=1}^{2} S_{j,i}\ p_{j,i}^t + y_i^t f_{hot,i} + v_i^t (f_{cold,i} - f_{hot,i}) + z_i^t C_{d,i}\ ]$$

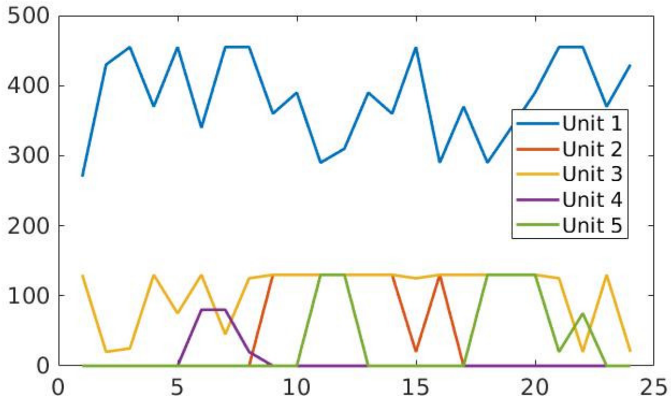

Fig. 2: Generation of each unit - 5-unit problem

IV. My experiments with MILP-UC

Much of my work has been in studying and formulating the MILP-UC problem. To solve the problem I decided to use MATLAB® 2018a. The software provides an optimization toolbox, equipped with a MILP solver, which greatly reduces the burden. I recommend avoiding the matrix form of $intlinprog$(the MILP solver) unless the problem being solved is already in matrix form / can be easily put in matrix form.

I tested the MILP-UC problem on two different data sets both obtained from [11]; one with the 5-unit problem and the other, 10-unit problem.

TABLE I: 5-unit problem

	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
Pmax(MW)	455	130	130	80	55
Pmin(MW)	150	20	20	20	55
a($/h)	1000	700	680	370	660
b($/MWh)	16.19	16.60	16.50	22.26	25.92
c($/MW^{2}h)	0.00048	0.002	0.00211	0.00712	0.00413
min up(h)	8	5	5	3	1
min down(h)	8	5	5	3	1
hot start cost($)	4500	550	560	170	30
cold start cost($)	9000	1100	1120	3400	60
cold start hrs(h)	5	4	4	2	0
initial status(h)	8	-5	-5	-3	-1

Hour	1	2	3	4	5	6	7	8
Demand	400	450	480	500	530	550	580	600
Hour	9	10	11	12	13	14	15	16
Demand	620	650	680	700	650	620	600	550
Hour	17	18	19	20	21	22	23	24
Demand	500	550	600	650	600	550	500	450

The shutdown costs are taken to be negligible.

The results obtained for the 5-unit problem can be seen below:

```
LP:                Optimal objective value is
                   114973.602650.

Cut Generation:    Applied 18 Gomory cuts,
                   43 implication cuts, 3 clique
                   cuts, 26 cover cuts, 22 mir
                   cuts, and 4 flow cover cuts.
                   Lower bound is 125433.985227.

Branch and Bound:

   nodes     total   num int     integer     relative
explored   time(s)   soln           fval      gap (%)
      51     0.61      1    1.284954e+05    2.382462
     284     0.95      2    1.276350e+05    1.724371
     629     1.53      3    1.269426e+05    1.188335
     787     1.81      4    1.268397e+05    1.108223
     814     1.84      5    1.267941e+05    1.072646
    1978     4.08      6    1.266438e+05    0.931158
    2064     4.23      7    1.266129e+05    0.9012234
    2154     4.37      8    1.265164e+05    0.8256493
    2376     4.78      9    1.264881e+05    0.8016210
    3211     6.46     10    1.264856e+05    0.6320921
    4228     8.30     10    1.264856e+05    0.000000

Optimal solution found.
```

The resulting Power generation and on-off status:

Time(h)	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
1	1	0	1	0	0
2	1	0	1	0	0
3	1	0	1	0	0
4	1	0	1	0	0
5	1	0	1	0	0
6	1	0	1	1	0
7	1	0	1	1	0
8	1	0	1	1	0
9	1	1	1	0	0
10	1	1	1	0	0
11	1	1	1	0	1
12	1	1	1	0	1
13	1	1	1	0	0
14	1	1	1	0	0
15	1	1	1	0	0
16	1	1	1	0	0
17	1	0	1	0	0
18	1	0	1	0	1
19	1	0	1	0	1
20	1	0	1	0	1
21	1	0	1	0	1
22	1	0	1	0	1
23	1	0	1	0	0
24	1	0	1	0	0

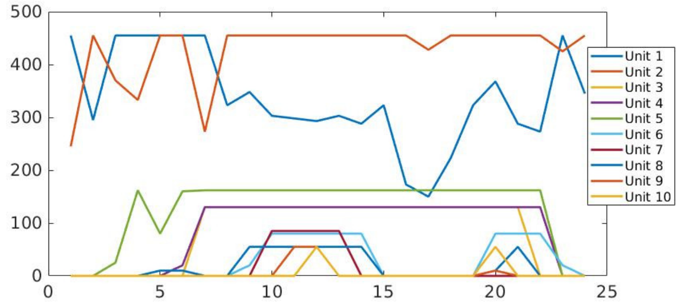

Fig. 3: Generation of each unit - 10-unit problem

TABLE II: 10-unit problem

	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
Pmax(MW)	455	455	130	80	162
Pmin(MW)	150	150	20	20	25
a($/h)	1000	970	680	370	450
b($/MWh)	16.19	17.26	16.60	16.50	19.70
c($/MW^{2}h)	0.00048	0.00031	0.0002	0.00211	0.00398
min up(h)	8	8	5	5	6
min down(h)	8	8	5	5	6
hot start cost($)	4500	5000	560	560	900
cold start cost($)	9000	10000	1100	1120	1800
cold start hrs(h)	5	5	4	4	4
initial status(h)	8	8	-5	-5	-6
	Unit 6	Unit 7	Unit 8	Unit 9	Unit 10
Pmax(MW)	80	85	55	55	55
Pmin(MW)	20	25	10	10	10
a($/h)	370	480	660	665	670
b($/MWh)	22.26	27.74	25.92	27.27	27.79
c($/MW^{2}h)	0.000712	0.00079	0.00413	0.00222	0.00173
min up(h)	3	3	1	1	1
min down(h)	3	3	1	1	1
hot start cost($)	170	260	30	30	30
cold start cost($)	340	550	60	60	60
cold start hrs(h)	2	2	0	0	0
initial status(h)	-3	-3	-1	-1	-1

Hour	1	2	3	4	5	6	7	8
Demand	700	750	850	950	1000	1100	1150	1200
Hour	9	10	11	12	13	14	15	16
Demand	1300	1400	1450	1500	1400	1300	1200	1050
Hour	17	18	19	20	21	22	23	24
Demand	1000	1100	1200	1400	1300	1100	900	800

The shutdown costs are taken to be negligible.

The results obtained for the 10-unit problem can be seen below:

```
LP:                 Optimal objective value is
                    236950.640921.

Cut Generation:     Applied 29 Gomory cuts,
                    81 implication cuts, 8 clique
                    cuts, 121 cover cuts, 30 mir
                    cuts, and 26 flow cover cuts.
                    Lower bound is 248986.695015.
```

```
Branch and Bound:

   nodes      total   num int    integer        relative
explored    time(s)   soln.       fval           gap (%)
    1608      7.78      1    2.616771e+05    4.849602e+00
    2138     10.14      2    2.568537e+05    3.062814e+00
    7080     28.79      3    2.547415e+05    2.218441e+00
    7268     29.42      4    2.532667e+05    1.649045e+00
   12333     43.47      5    2.532285e+05    1.634225e+00
   13978     47.48      6    2.524286e+05    1.322514e+00
   13980     47.48      7    2.523803e+05    1.303626e+00
   16478     57.94      8    2.522366e+05    1.247395e+00
   26478    109.65      8    2.522366e+05    9.141902e-01
   36478    155.88      8    2.522366e+05    6.890370e-01
   36629    156.51      9    2.522167e+05    6.812215e-01
   42278    184.59     10    2.521664e+05    5.596289e-01
   52278    225.06     10    2.521664e+05    3.539964e-01
   52641    226.65     11    2.521298e+05    3.395244e-01
   60441    257.54     12    2.521252e+05    2.239176e-01
   70441    291.45     12    2.521252e+05    1.024813e-01
   77099    313.08     12    2.521252e+05    2.799749e-03
   77177    313.55     12    2.521252e+05    0.000000e+00

Optimal solution found.
```

The resulting Power generation and on-off status:

Time(h)	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
1	1	1	0	0	0
2	1	1	0	0	0
3	1	1	0	0	1
4	1	1	0	0	1
5	1	1	0	0	1
6	1	1	0	1	1
7	1	1	1	1	1
8	1	1	1	1	1
9	1	1	1	1	1
10	1	1	1	1	1
11	1	1	1	1	1
12	1	1	1	1	1
13	1	1	1	1	1
14	1	1	1	1	1
15	1	1	1	1	1
16	1	1	1	1	1
17	1	1	1	1	1
18	1	1	1	1	1
19	1	1	1	1	1
20	1	1	1	1	1
21	1	1	1	1	1
22	1	1	0	1	1
23	1	1	0	0	0
24	1	1	0	0	0

Time(h)	Unit 6	Unit 7	Unit 8	Unit 9	Unit 10
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	1	0	0
6	0	0	1	0	0
7	0	0	0	0	0
8	0	0	0	0	0
9	1	0	1	0	0
10	1	1	1	0	0
11	1	1	1	1	0
12	1	1	1	1	1
13	1	1	1	0	0
14	1	0	1	0	0
15	0	0	0	0	0
16	0	0	0	0	0
17	0	0	0	0	0
18	0	0	0	0	0
19	0	0	0	0	0
20	1	0	1	1	1
21	1	0	1	0	0
22	1	0	0	0	0
23	1	0	0	0	0
24	0	0	0	0	0

V. Conclusion

With the result of my two experiments, I have successfully utilized the MILP-UC formulation to obtain a global minimum over the cost, with individual units start/stop provided. The given solutions can be shown to satisfy all the different unit/system constraints. The time to converge for the first case is much smaller than the second. This is because the complexity of solution increases exponentially with the number of units. Due to the assumption that we have a piecewise linear cost-function, we are bound to get some error in the results, but we can increase the accuracy with an increase in number of pieces. This however, has a tradeoff in terms of compute time, since it would increase the number of integer constraints. This project can be further be worked upon in terms of other constraints, with renewable energy sources and fuel constraints.

Acknowledgment

I would like to thank Prof. Zakir Rather for a chance to work on this problem, under his guidance. I would also like to thank Hazem Karbouj for his insights.

References

[1] Rammohan Mallipeddi and Ponnuthurai Suganthan. "Unit commitment - A survey and comparison of conventional and nature inspired algorithms". In: *Int. J. of Bio-Inspired Computation* 6 (Jan. 2014), pp. 71–90. DOI: 10.1504/IJBIC.2014.060609.

[2] G. W. Chang et al. "A practical mixed integer linear programming based approach for unit commitment". In: *IEEE Power Engineering Society General Meeting, 2004.* June 2004, 221–225 Vol.1. DOI: 10.1109/PES.2004.1372789.

[3] Julia Borghoff. "Mixed Integer Programming: Algorithms and Applications". In: *ECRYPT II Summer School on Tools* (June 2012), pp. 6–16.

[4] MATLAB. *version 9.4.0 (R2018a)*. Natick, Massachusetts: The MathWorks Inc., 2018.

[5] William Dhaeseleer Erik Delarue. "Advanced Priority Listing Versus Mixed Integer Programming In Solving The Unit Commitment Problem". In: *Working Paper* (Dec. 2007), pp. 4–5.

[6] Allen J Wood et al. *Power generation, operation, and control*. English. Third edition / Allen J. Wood, Bruce F. Wollenberg, Gerald B. Sheble. ProQuest Ebook Central Rental. Hoboken, New Jersey : Wiley-IEEE, 2013. ISBN: 9781118733912 (ePub). URL: https://www.lib.uts.edu.au/goto?url=http://UTS.eblib.com.au/patron/FullRecord.aspx?p=1418364.

[7] Lei Wu. "A Tighter Piecewise Linear Approximation of Quadratic Cost Curves for Unit Commitment Problems". In: *IEEE Transactions on Power Systems* 26 (2011), pp. 2581–2583.

[8] L. L. Garver. "Power Generation Scheduling by Integer Programming-Development of Theory". In: *Transactions of the American Institute of Electrical Engineers. Part III: Power Apparatus and Systems* 81.3 (Apr. 1962), pp. 730–734. ISSN: 0097-2460. DOI: 10.1109/AIEEPAS.1962.4501405.

[9] Alireza Soroudi. *Power System Optimization Modeling in GAMS*. Springer International Publishing, 2017.

[10] R. D. Naidoo. "A Mixed Integer Programming formulation of generator startup costs". In: *2007 International Power Engineering Conference (IPEC 2007)*. Dec. 2007, pp. 173–176.

[11] S. A. Kazarlis, A. G. Bakirtzis, and V. Petridis. "A genetic algorithm solution to the unit commitment problem". In: *IEEE Transactions on Power Systems* 11.1 (Feb. 1996), pp. 83–92. ISSN: 0885-8950. DOI: 10.1109/59.485989.