

Controlling Characteristics of the Slot Wave by Virtual Capacity Setting: LIP

Solving Slot Allocation Based on Data Splitting Method

ZhiJian Ye^{1*} Wenpeng Zhai² XinXin Zheng³ Shihao Huang⁴

- 1 College of Air Traffic Management, Civil Aviation University of China, Tianjin300300, China; zjye@cauc.edu.cn
- 2 College of Air Traffic Management, Civil Aviation University of China, Tianjin300300, China; wpZhai@cauc.edu.cn
- 3 College of Cabin attendant, Civil Aviation University of China, Tianjin300300, China; ruby2621@163.com
- 4 College of Air Traffic Management, Civil Aviation University of China, Tianjin300300, China; 641982845@qq.com

Correspondence: ZhiJian Ye. zjye@cauc.edu.cn; +86-13022226938

Abstract: Management of demand and capacity for airport operation is important. The effect of capacity setting on slot adjustment, maximum adjustment and flight delay has been extensively investigated and discussed in previous slot adjustment models. But its effect on characteristics of flight waves has not been studied deeply. At present, in order to resolve the problem that Linear Integer Programming (LIP) is limited in slot allocation due to its fast dimension growth and large memory consumption, the general practice is to optimize the slot application according to data-splitting technology. However, the relationship among data splitting method, capacity setting and characteristics of slot wave has not been investigated in detail, which limits the application of LIP in slot allocation. Through the detailed analysis and testing, we found that the waveform and amplitude of slot wave can be controlled and designed by appropriate virtual capacity settings, while the number of capacity constraints and the computing time is reduced by adopting appropriate data feeding method. Our research also provides clues for further research on the construction of the flight waves for airlines operating at specific airports in order to establish direct or indirect connectivity, and increase hit rate by using our approaches.

Keywords: Slot Allocation; Characteristics of Slot Wave; Capacity Setting; Data Feeding Methods; Linear Integer Programming

1 Introduction

Most of the busiest airports worldwide experience serious congestion and delay problems. The experienced imbalance between supply and demand for air transport services forces all aviation stakeholders to drastically rethink airport capacity and its utilization and readdress the issue of experienced or anticipated capacity shortages [1, 2]. Results of past research have proved that demand management could provide significant benefits at busy worldwide airports by permitting large delay reductions through limited interference with airline competitive scheduling [3] or by evenly small substantial increase in declared capacity [4]. But the latter aiming to build new capacity are capital intensive solutions, require significant implementation time, and are often subject to heated political debates. The need for an immediate relief to seriously congested airports calls for short to medium-term, demand-side solutions that are based on the optimum allocation and use of

available airport capacity [5]. To control over-capacity scheduling, the most common demand management schemes fall into two categories: (i) approaches introducing market-driven or pure economic instruments (e.g., slot trading, auctions, congestion pricing) aiming to allocate capacity among competing users by considering real market (or approximations of) valuations of access to congested airport facilities [6-11]. (ii) efforts aiming to improve the efficiency by using administrative allocation mechanism [3-5, 12-18].

In order to control the excessive demand of airports, Chinese Civil Aviation (CAA) has enacted slot regulation since 2010. But due to various reasons, satisfied effect has not been achieved. Consequently, the new slot regulation of CAA similar to that of International Air Transport Association (IATA) was introduced in April 2018. Although flight delay in Chinese airport has been improved after the implementation of these slot management method, the unreasonable slot structure still exists [19-22]. As the analysis of these articles indicated, the typical characteristics of slot structure at large Chinese airports are that the departure of flights is concentrated in the morning and the arrival of flights is concentrated in the evening. Because of this unreasonable schedule, the low punctuality rate of flights in the morning and evening rush hours is doomed. So, in order to further improve the arrival and departure punctuality rate, it is still necessary to further reasonably adjust the slot structure while accurately assesses and uses declared capacity.

The effect of capacity setting on slot adjustment, maximum adjustment[15] [18, 23] and flight delay[16, 17] has been extensively investigated and discussed in previous slot adjustment models. But its effect on flight waves has not been studied deeply. Because of many constraints and complex variables, the Linear Integer Programming (LIP) Algorithms for slot adjustment model needs a large amount of memory to calculate. At present, to resolve this problem, the general practice is to optimize the slot application according to data-splitting technology. However, the relationship between data splitting method (equal to data feeding method) and capacity setting has not been investigated in detail, which limits the application of LIP in slot allocation.

The main contribution of this paper is two folds. The first is that the number of capacity constraints and the computing time is reduced by adopting appropriate data feeding method. Five data feeding methods are analyzed and the best one is selected for our algorithm according to the number of feedable application. It solves the problem that LIP is difficult to be used in large-scale slot allocation because of prohibited memory consuming. Secondly, through detailed analysis and testing, we found that the waveform and amplitude of slot wave can be controlled and designed through virtual capacity settings. This, in turn, eliminates the concerns of uncontrollable demand while reducing capacity constraints. These finding is very useful to control demand that do not exceed declared capacity and reduce dimension of constrains by virtual capacity setting.

The remainder of this paper is organized as follows. Section 2 formulates the model. Section 3 is the iterative Linear Integer Programming (LIP) Algorithms according to data-splitting technology based on the detailed analysis of five data feeding methods. Section 4 presents the test results with different capacity setting and the effect of capacity setting on characteristics of slot wave. Section 5 summarizes our work and indicates directions for future research.

2 Notations and slot displacement models

Before describing the model, notations are stated as follows:

p_m : Priority of movement m .

f_m^D : Flight implementation difficulties

I_m^D : Flight difficulty index of one displacement unit

τ_m : Interval that movement m required.

t_m : Interval that movement m is scheduled.

n : Seats of flights corresponding movement m .

π_m : The average flight elapse time of movement m , to (arrival flight) or from (departure flight) this airport.

L_a : Coordination level parameters of this airport (main coordinator, auxiliary coordinated airport and uncoordinated airport; 7, 4, 1).

L_b : Coordination level parameters of associated airports.

x_m^t : { 0, 1}, if movement m is scheduled at t interval.

C_{60} : Hourly capacity constraint;

C_{15} : The 15-minutes capacity constraint;

C_5 : The 5-minutes capacity constraint;

b_{me} : Corridor capacity constraints, $e = (1, 2 \dots E)$.

a_m^d : Movement m plans to operate on day d of a series day, usually series day expressed as [1, 2, 3, 4, 5, 6, 7]. a_m^d is set mandatorily as 1, which means the same flight operating in one day has two movement number, such as m_1 and m_2 .

b_{mc} : The amount of the kind of capacity c consumed by movement m . In our model, c may be hourly capacity, 15-minutes capacity, 5-minutes capacity, or corridor capacity. In our model, the amount of each capacity consumed by a movement is 1, which means all of $b_{mc_{60}}$, $b_{mc_{15}}$, b_{mc_5} and b_{me} are set as 1.

n : Number of seats.

The coordination time interval represents the unit of time (e.g., 5-min, 15-min, 60-min) used as the basis for capacity determination and slot allocation. Usually each time interval contains multiple slots. A movement corresponds to a takeoff or landing activity. A slot refers specifically to the interval occupied by one movement.

Flight implementation difficulties index I_m^D and difficulty f_m^D brought by difficulty index expressed as follow:

$$I_m^D = \left\lceil \left(\frac{n}{\pi_m} \right)^{\frac{1}{2}} \cdot (L_a L_b)^{\frac{3}{2}} \right\rceil \quad (1)$$

$$f_m^D = |t_m - \tau_m| \cdot I_m^D = |t_m - \tau_m| \cdot \left\lceil \left(\frac{n}{\pi_m} \right)^{\frac{1}{2}} \cdot (L_a L_b)^{\frac{3}{2}} \right\rceil \quad (2)$$

As mentioned before, cost of one movement in most of traditional slot allocation models is as following:

$$f_m^O = |t_m - \tau_m| \cdot 1 \quad (3)$$

In expression (3), f_m^O is the interval displacement when required interval τ_m is replaced with scheduled interval t_m .

When priority is considered as the cost of displacing the unit time, cost of one movement can be presented as following:

$$f_m^P = |t_m - \tau_m| \cdot p_m \quad (4)$$

In our model, we integrate cost of displacement, difficulties and priorities as a whole in order to facilitate calculation and comparison. At the same time, three weight factors (w_1, w_2, w_3) are introduced artificially for the same reason. Performance comparing under different target weights have been done in our previous articles[24], so this paper sets the weight to [1, 0, 0], only changes the capacity settings in constrains to observe the waveform changes. The comprehensive displacement

cost of a movement, displaced from required interval τ_m to scheduled interval t_m , could be expressed as following:

$$\delta_m^{ODP} = (w_1 \cdot 1 + w_2 \cdot I_m^D + w_3 \cdot p_m) = (w_1 \cdot 1 + w_2 \cdot \left[\left(\frac{n}{\pi_m} \right)^{\frac{1}{2}} (L_a L_b)^{\frac{3}{2}} \right] + w_3 \cdot p_m) \tag{5}$$

$$f_m^t = w_1 \cdot f_m^O + w_2 \cdot f_m^D + w_3 \cdot f_m^P = |t_m - \tau_m| \delta_m^{ODP} \tag{6}$$

We call δ_m^{ODP} as the comprehensive displacement cost factor with the consideration of implementation difficulties and priority. It is important to find that δ_m^{ODP} is a constant, which makes it possible to solve the following slot allocation model by Linear Integer Programming (LIP).

minimize

$$\sum_{m \in M} \sum_{t \in T} f_m^t x_m^t = \sum_{m \in M} \sum_{t \in T} |t_m - \tau_m| \delta_m^{ODP} x_m^t \tag{7}$$

subject to

$$\sum_{t \in S} x_m^t = 1, \quad m \in M \tag{8}$$

$$\sum_{m \in M} \sum_{t \in T_c^s} a_{mc}^d b_{mc} x_m^t \leq u_c^{ds}, \quad c \in C, d \in D, s \in T_c \tag{9}$$

$$x_m^t \in \{0,1\}, \quad m \in M, \quad t \in S \tag{10}$$

The objective function (7) minimizes the overall displacement cost of all flights. Constraint (8) stipulates that every movement must be allocated to one interval. Constraint (9) specifies that total movement consumption cannot exceed capacity, for each constraint, day and interval. Constraint (10) ensures that this model can be solved by integer programming method. There is no detailed description of the coefficient matrix in previous papers. Thus, in the next section, we will elaborate proposed approach.

3 LIP for slot displacement models

In order to limit the flow of corridors, it is necessary to determine which corridor should be allocated to each flight. Firstly, the courses of all flights from the airport to the linked airport are calculated. Then, the courses for arrival or departure are sorted descending. If there are m corridors for arrival (or departure), then the arrival (or departure) courses are classified into m categories.

In general, the number of corridors for arrival is equal to the number of corridors for departure. Each corridor entrance must meet the capacity constraints. All movement is classified according to the number of corridors and assigned evenly to corridor before executing LIP. By designing the variable $y_i^e = \{0,1\}, i = 1 \dots M, e = 1 \dots E$, the relationship matrix of movement and corridors is constructed as shown in table 1. The sum of each column must be 1, that is, each movement i must be assigned to a corridor e . The sum of each row is limited by the capacity of the corridor in every interval. This makes it easy to solve the capacity constraints of the corridor with linear programming, but with the increase of the number of flights and the number of corridors, the dimension of the constraints increases rapidly.

Table 1. relationship matrix of movement i and corridors e

<div>e \ i</div>	1	2	3	4	...	M	C_e
1	1	0	0	0	...	0	C_1
2	0	0	1	0	...	1	C_2
3	0	0	0	0	...	0	C_3
...
E-1	0	0	0	1	...	0	C_{E-1}
E	0	1	0	0	...	0	C_E

Considering that the flight execution cycle is usually at least once a week, and that most flights operate every day, in order to reduce the computing time with the proposed approach, it is possible to determine the calendar days with the same set of requests, and then represent these as a single calendar day. In order to keep the accumulated rolling volume of flights from exceeding the hourly capacity, we consider three kinds of capacities in brackets (hourly capacity, 15-minutes capacity and 5-minutes capacity) to prevent this from happening. For preventing memory overflow, these three capacities are arranged separately and in the order of hourly capacity, 15-minutes capacity and 5-minute capacity sequentially, that is to say, the other two are not active in the arrangement of the third capacity. Because of the fact that 5-minutes capacity has greatest effect on the uniform distribution of time, the check of 5-minutes capacity is put at the end. When constrains of 5-minutes capacity is check, the dimension of the constraint is so large that sometimes the memory is insufficient to execute. Therefore, we activate constrains of runway capacity and corridors capacity in batches according to the order of priority. The corresponding capacity is updated after each batch of arrangement. This top-down decomposition of these constraints is shown in Figure 3.

There are five ways to feed data for equation (9) in order to split data and reduce memory requirements.

- (1) The first is to satisfy capacity constraints from top to bottom, as we do, which is equivalent to allocating applications to each hourly interval according to the hourly capacity constraints. On the basis of the results, as many applications as possible are allocated to 15-minutes intervals in batches according to 15-minutes capacity constraints. Finally, on the basis of the previous results, according to the 5-minutes capacity constraints, as many applications are allocated to 5-minutes intervals in batches as possible. This feeding process is shown in Figure 1.

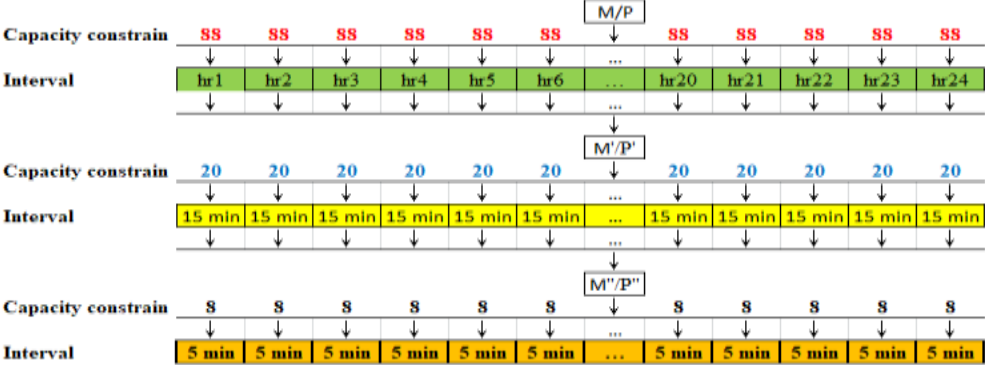


Figure 1. Data feeding method 1

- (2) The second is to consider three capacity constraints at the same time. Data feeding method of this kind is shown in Figure 2.

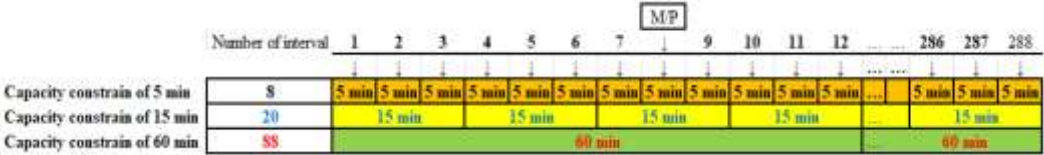


Figure 2. Data feeding method 2

- (3) The third is to satisfy capacity constraints from top to bottom as the first one, which is equivalent to allocating applications to each hourly point according to the hourly capacity constraints. This stage is the same as the first method. Then, according to the 15-minutes capacity constraints, the application within one hour is allocated to the 15-minutes time points, and finally the application within 15 minutes is allocated to the 5-minutes time points according to the 5-minutes capacity constraints. Data feeding method of this kind is shown in Figure 3.

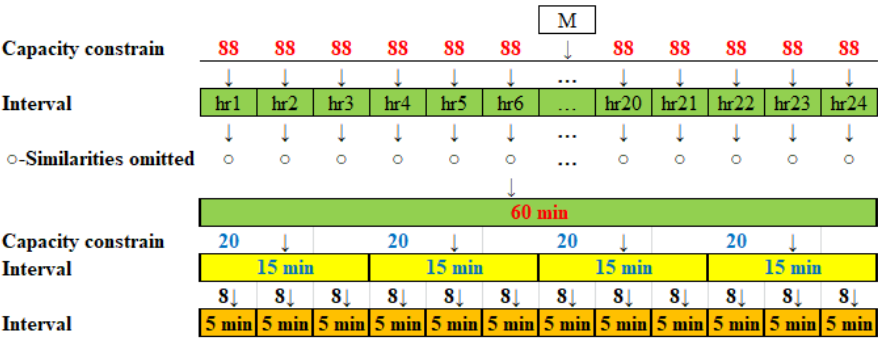


Figure 3. Data feeding method 3

- (4) The fourth is to reverse the first method of feeding.
(5) The fifth is to reverse the third method of feeding.

It is obvious that the first data feeding method is better than the third and fifth methods, because the more applications are fed each time, the greater the possibility of approaching the global optimum. Similarly, the fourth method is better than the fifth one. The first data feeding method is better than the fourth method, because by controlling the application volume in a small interval makes it impossible to overcapacity in a large interval with the first data feeding method. Conversely, small interval may exceed capacity with the fourth data feeding method. The first is better than the second, which we will discuss at the end. So we use the data feeding method 1 in procedures of slot allocation.

The following is the procedures of iterative linear integer programming algorithms based on data-splitting method 1.

Column generation procedure

Priority calculation f_m^P

For $i=1:7$

Load movement in day 1

Set up hourly capacity C_{60} , 15-minutes capacity C_{15} , 5-minutes capacity C_5

Calculating the minimum total capacity, $Z = \min(24 \times C_{60}, 96 \times C_{15}, 288 \times C_5)$

Compared with M and Z , if M is larger than Z , the number of discarded requests is equal to $M - Z$

Use simplex method to arrange all remaining applications into each hourly period

Use simplex method to arrange the 15-minutes period on the basis of the hour schedule

Use simplex method to arrange the 5-minutes period on the basis of the 15-minutes schedule

End

Table 2 is an example of describing the coefficient matrix in detail in the objective function and constraints within hours.

Table 2. Coefficient matrix and detailed expression when slot is based on hour

Execution batch equals $P=[M/Q]$	$\sum_{m \in M} \sum_{t \in T} f_m^t x_m^t$ $= \sum_{m \in M} \sum_{t \in T} t_m - \tau_m \delta_m^{DP} x_m^t$	f_m^t	$f_m^t x_m^t = f_1^1 x_{11} + f_1^2 x_{12} + f_1^3 x_{13} + \dots + f_1^{24} x_{124} +$ $f_2^1 x_{21} + f_2^2 x_{22} + f_2^3 x_{23} + \dots + f_2^{24} x_{224} +$ \dots $f_m^1 x_{m1} + f_m^2 x_{m2} + f_m^3 x_{m3} + \dots + f_m^{24} x_{m24}$	(a.0)
	$\sum_{m \in M} \sum_{t \in T_c^s} a_m^d b_{mc} x_m^t \leq u_c^{ds}$, $c \in \text{hour}, d \in [1, 2, \dots, 7]$, $S \in [1, 2, \dots, 24]$ $a_m^d = 1, b_{mc} = 1$	A_1	$x_{11} + x_{21} + x_{31} + \dots + x_{m1} \leq u_{hour}^1$	(a.1)
			$x_{12} + x_{22} + x_{32} + \dots + x_{m2} \leq u_{hour}^2$	(a.2)
			\vdots	\vdots
			$x_{124} + x_{224} + x_{324} + \dots + x_{m24} \leq u_{hour}^{24}$	(a.24)
	$\sum_{m \in M} \sum_{t \in T_c^s} a_m^d b_{mc} x_m^t \leq u_c^{ds}$, $c \in c1, \text{corridor } 1$ $d \in [1, 2, \dots, 7]$, $S \in [1, 2, \dots, 24]$ $a_m^d = 1, b_{mc} = 1$	Y	$y_1^1 x_{11} + y_2^1 x_{21} + y_3^1 x_{31} + \dots + y_m^1 x_{m1} \leq u_{c1}^1$	(b.1)
			$y_1^1 x_{12} + y_2^1 x_{22} + y_3^1 x_{32} + \dots + y_m^1 x_{m2} \leq u_{c1}^2$	(b.2)
			\vdots	\vdots
			$y_1^1 x_{124} + y_2^1 x_{224} + y_3^1 x_{324} + \dots + y_m^1 x_{m24} \leq u_{c1}^{24}$	(b.24)
			\vdots	\vdots
			$y_1^8 x_{11} + y_2^8 x_{21} + y_3^8 x_{31} + \dots + y_m^8 x_{m1} \leq u_{c8}^1$	(b.169)
			$y_1^8 x_{12} + y_2^8 x_{22} + y_3^8 x_{32} + \dots + y_m^8 x_{m2} \leq u_{c8}^2$	(b.170)
			\vdots	\vdots
			$y_1^8 x_{124} + y_2^8 x_{224} + y_3^8 x_{324} + \dots + y_m^8 x_{m24} \leq u_{c8}^{24}$	(b.216)
	$\sum_{t \in T} x_m^t = 1, m \in Q$ $x_m^t \in \{0, 1\}, m \in Q$ $t \in [1, 2, \dots, 24]$	A_{eq}	$x_{11} + x_{12} + x_{13} + \dots + x_{124} = 1$	(d.1)
			$x_{21} + x_{22} + x_{23} + \dots + x_{224} = 1$	(d.2)
			\vdots	\vdots
			$x_{m1} + x_{m2} + x_{m3} + \dots + x_{m24} = 1$	(d.m)

4 Testing results and analysis of the effect of capacity setting on waveform of slot

The purpose of the test covers two folds. First is to observe the feasibility and efficiency of the proposed algorithm. Secondly, by changing the hourly capacity, 15-minutes capacity and 5-minutes capacity, the change of flight wave structure can be observed and analyzed. Test parameter setting is show in Table 3. The test data comes from the OAG database. The data is the actual operation data, not the schedule data, and only part of the data for a week, with 1418 movements per day. Capacity settings are assumed for testing purposes, not actual operating capacity.

Table 3. Relationship between capacity setting and figures of results of slot allocation algorithm

Test number	Capacity setting(M) C60, C15, C5	Figure	Interval (min)	Calculation time (s)
Test 1	88(84↑),20(21↓),8(7↑)	Figure 4-6	60,15, 5	4882
Test 2	60(84↓),20(21↓),8(7↑)	Figure 7-9	60,15, 5	4988
Test 3	84,21,7 (Benchmark)	Figure 10-12	60,15, 5	4866
Test 4	76(84↓),21,7	Figure 13-15	60,15, 5	5055
Test 5	84,19(21↓10%),7	Figure 16-18	60,15, 5	4890

Test 6	84,23(21↑10%,),7	Figure 19-21	60,15, 5	4846
Test 7	88(84↑5%),21,7	Figure 22-24	60,15, 5	4850
Test 8	88(84↑5%),23(21↑10%),7	Figure 25-27	60,15, 5	4848
Test 9	88(84↑5%),18(21↓14%),7	Figure 28-30	60,15, 5	4896

4.1 **Figures of results of slot allocation algorithm with different virtual capacity setting**

There are eight tests in total, and each test produces a timetable. The hour distribution, 15-minutes distribution and 5-minutes distribution of the timetable are plotted and shown in the figures resulting from following tests.

4.1.1 **Test 1: three kind of capacity set as 88(84↑),20(21↓),8(7↑)**

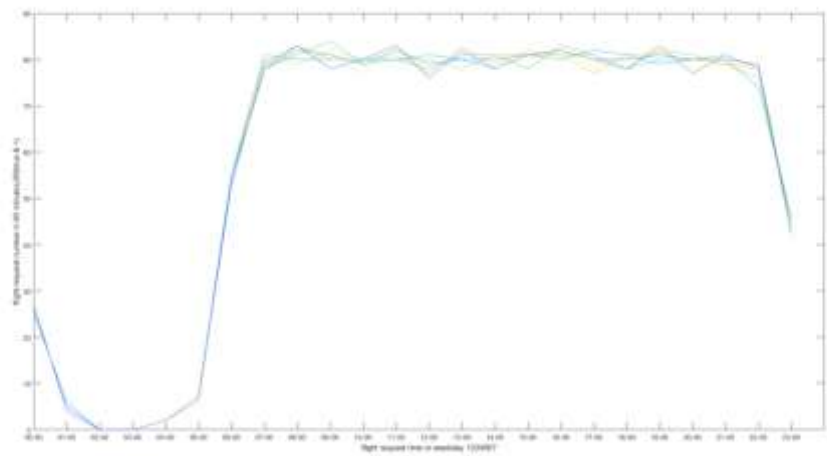


Figure 4. The hourly wave of capacity set as 88,20,8

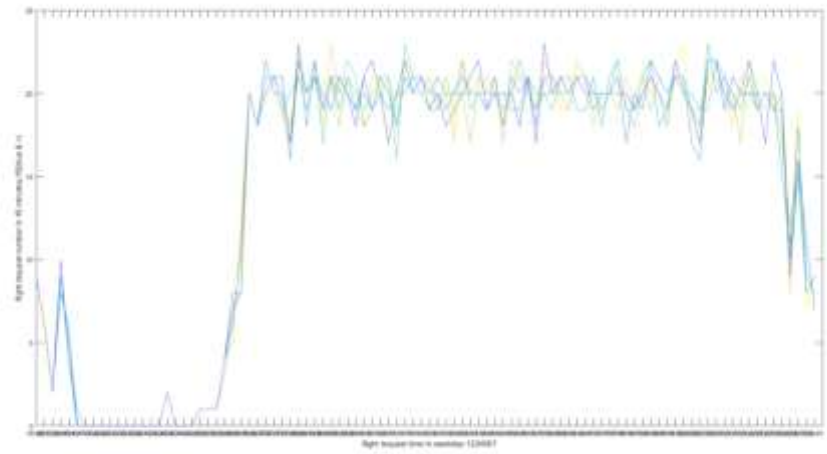


Figure 5. The 15minutes wave of capacity set as 88,20,8

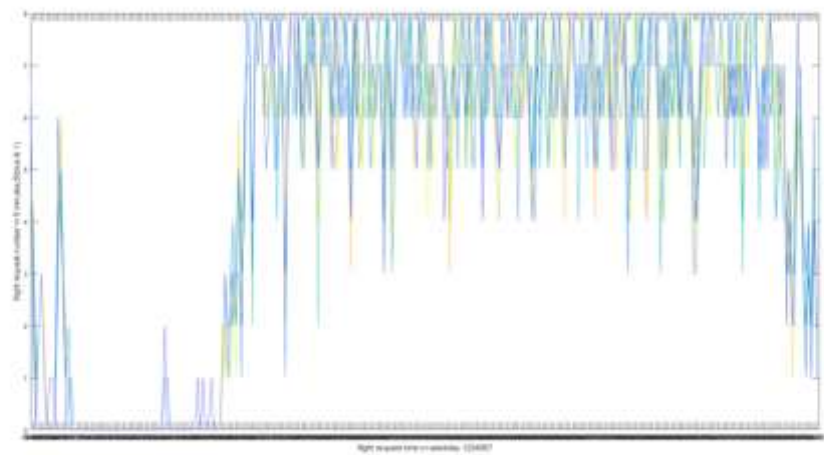


Figure 6. The 5minutes wave of capacity set as 88,20,8

4.1.2 Test 2: three kind of capacity set as 60(84↓),20(21↓),8(7↑)

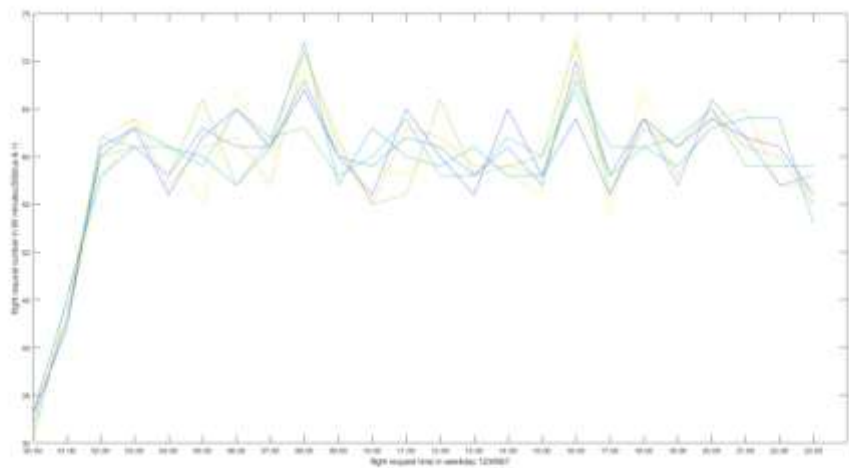


Figure 7. The hourly wave of capacity set as 60,20,8

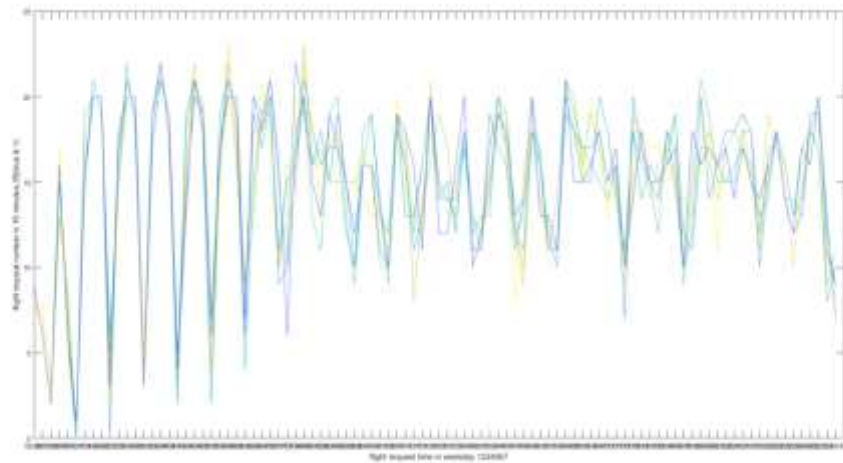


Figure 8. The 15minutes wave of capacity set as 60,20,8

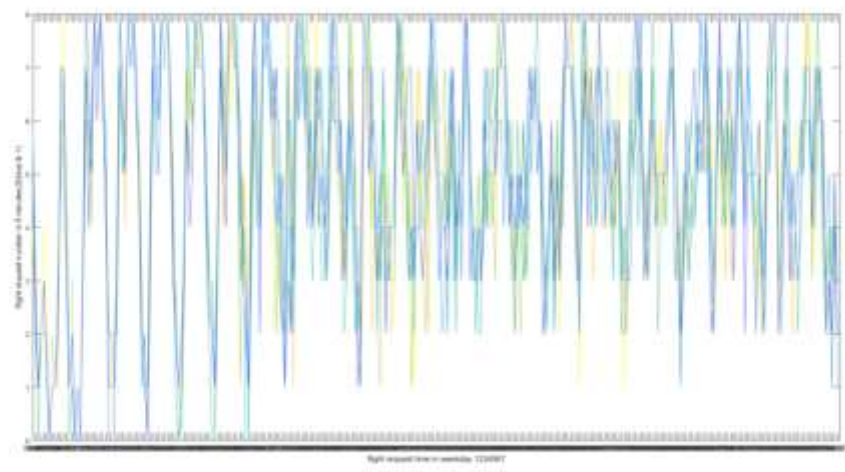


Figure 9. The 5minutes wave of capacity set as 60,20,8

4.1.3 Test 3: three kind of capacity set as 84,21,7 (Benchmark)

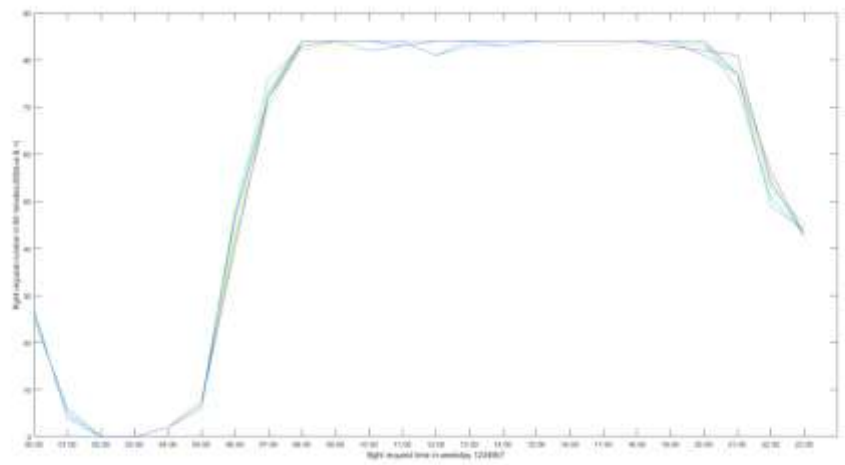


Figure 10. The hourly wave of capacity set as 84,21,7

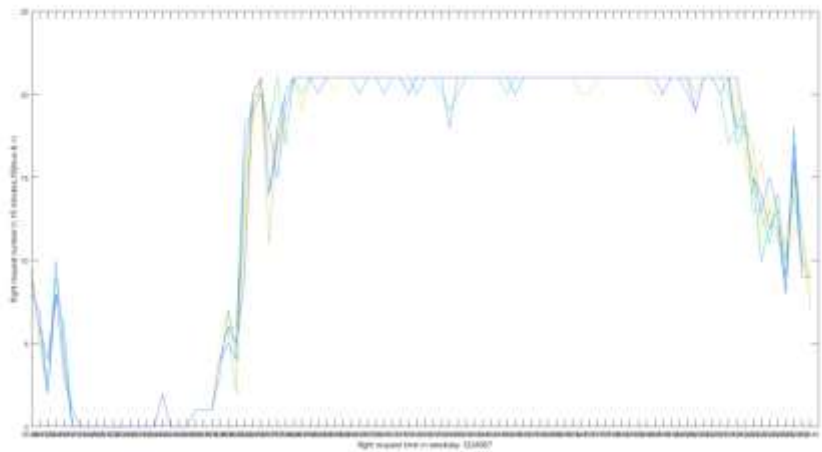


Figure 11. The 15minutes wave of capacity set as 84,21,7

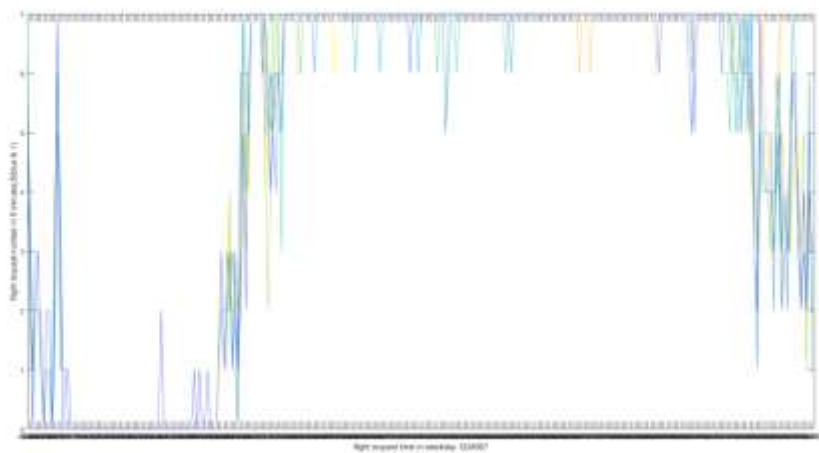


Figure 12. The 5minutes wave of capacity set as 84,21,7

4.1.4 Test 4: three kind of capacity set as 76(84↓10%),21,7

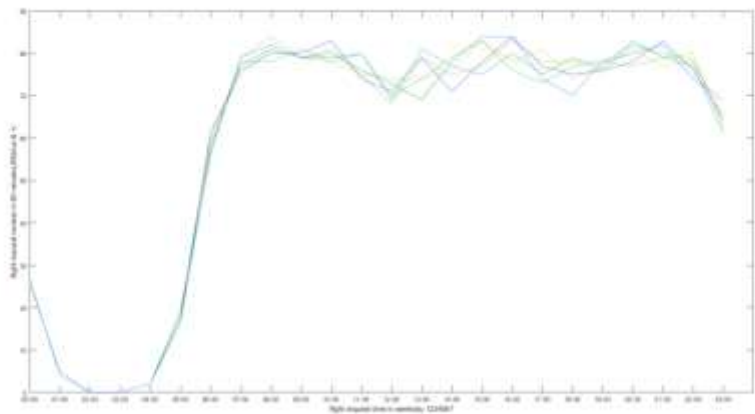


Figure 13. The hourly wave of capacity set as 76(84↓10%),21,7

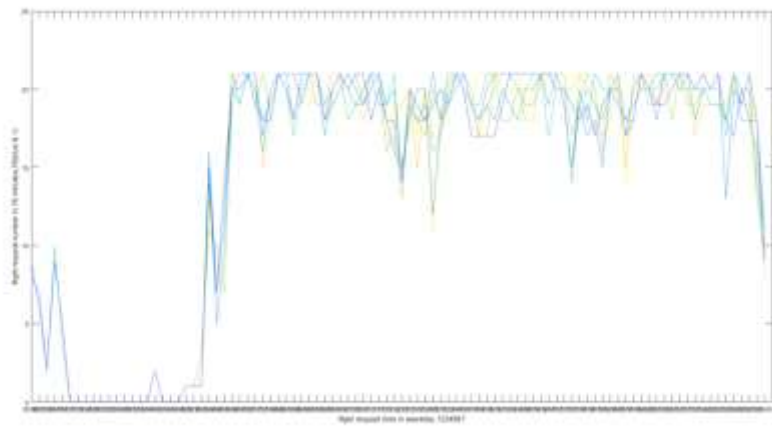


Figure 14. The 15minutes wave of capacity set as 76(84↓10%),21,7

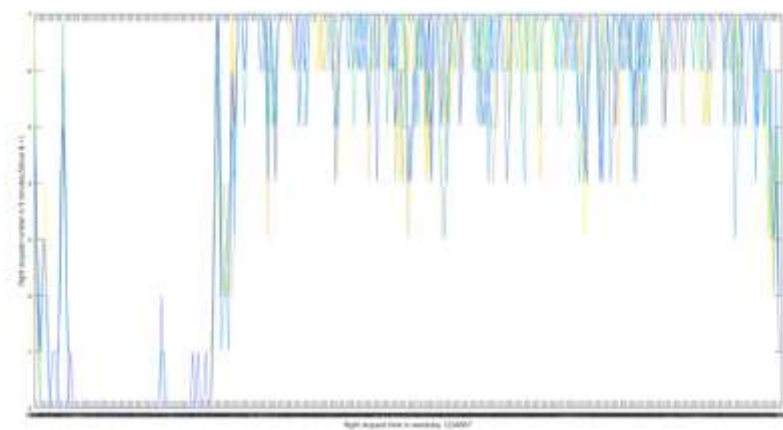


Figure 15. The 5minutes wave of capacity set as 76(84↓10%),21,7

4.1.5 Test 5: three kind of capacity set as 84,19(21↓10%),7

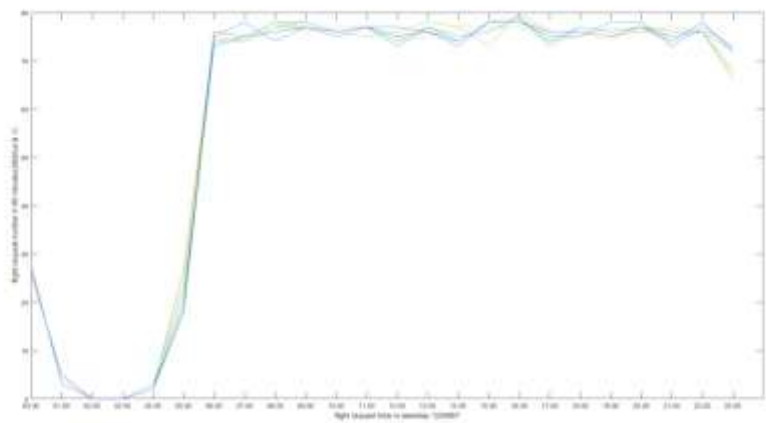


Figure 16. The hourly wave of capacity set as 84,19(21↓10%),7

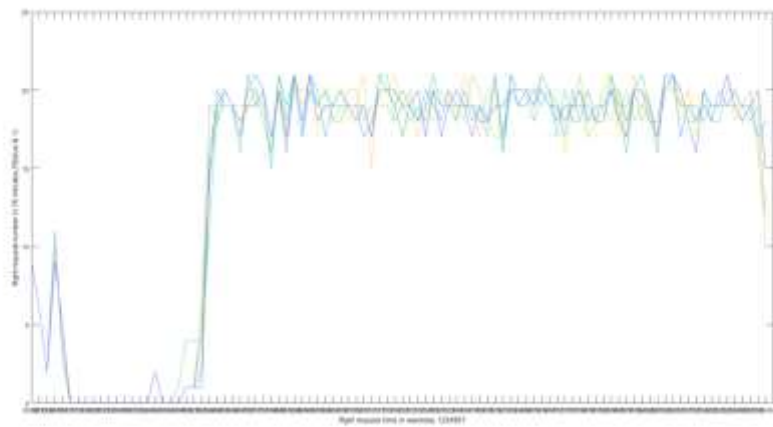


Figure 17. The 15minutes wave of capacity set as 84,19(21↓10%),7

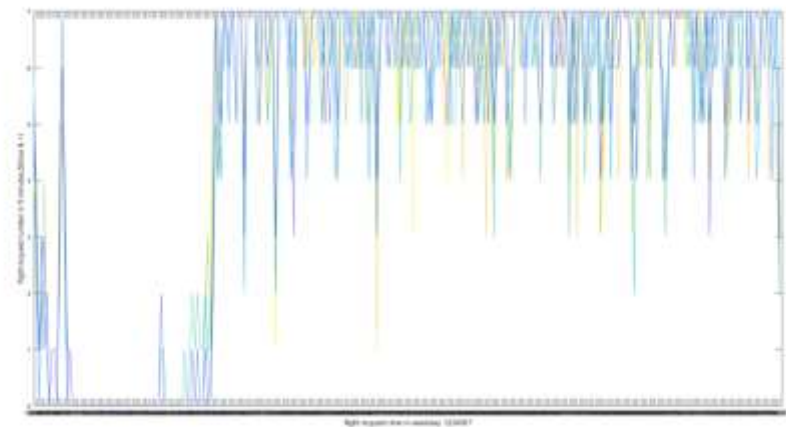


Figure 18. The 5minutes wave of capacity set as 84,19(21↓10%),7

4.1.6 Test 6: three kind of capacity set as 84,23(21↑10%),7

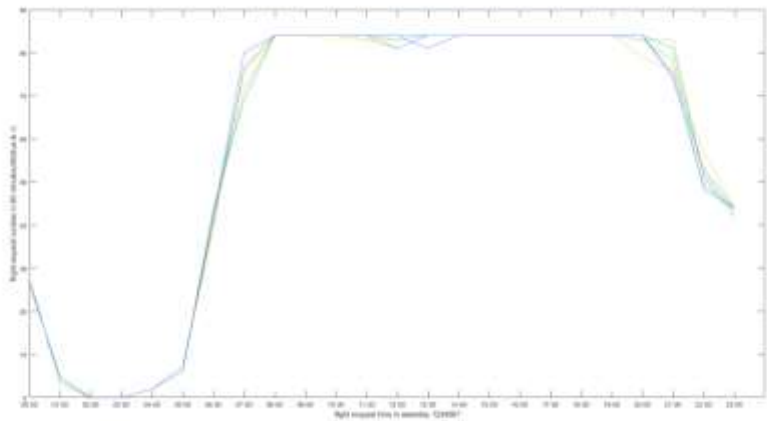


Figure 19. The hourly wave of capacity set as 84,23,7

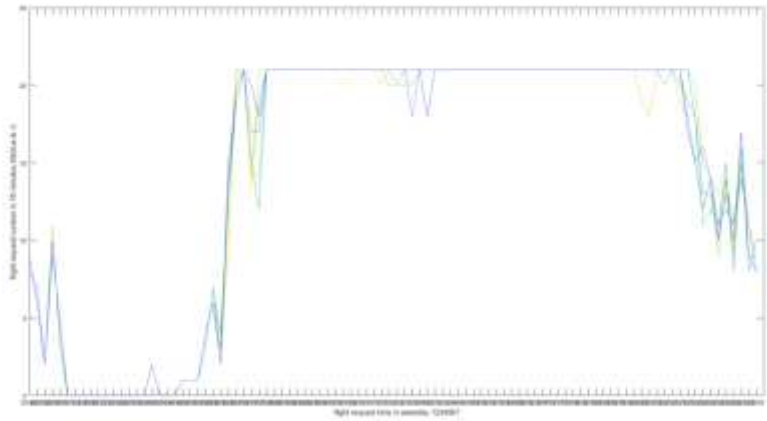


Figure 20. The 15minutes wave of capacity set as 84,23,7

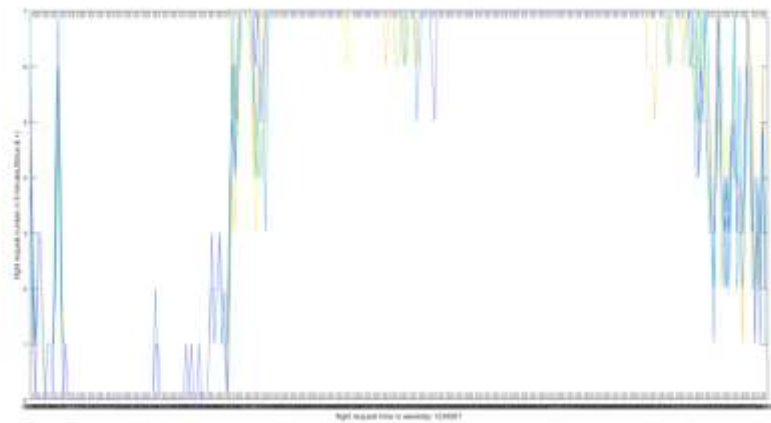


Figure 21. The 5minutes wave of capacity set as 84,23,7

4.1.7 Test 7: three kind of capacity set as 88(8415%),21,7

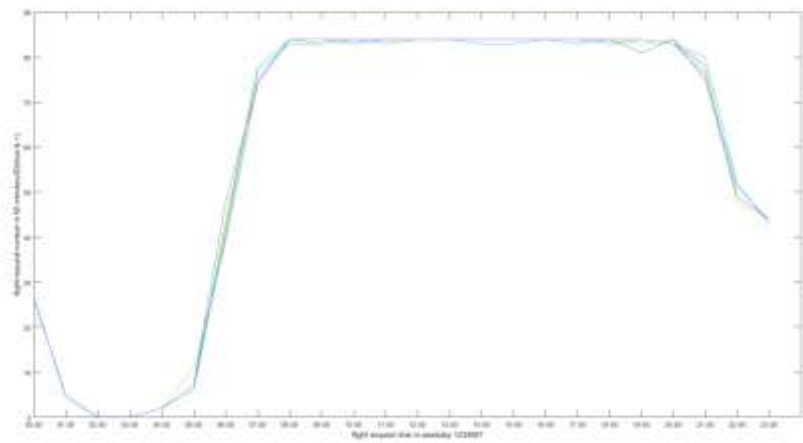


Figure 22. The hourly wave of capacity set as 88(8415%),21,7

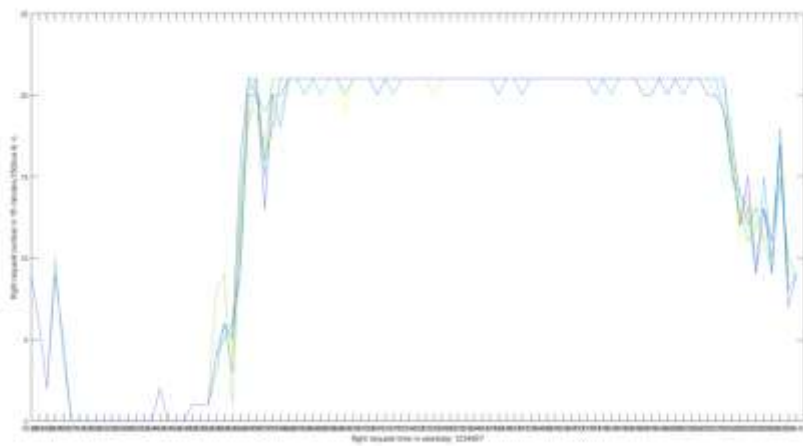


Figure 23. The 15minutes wave of capacity set as 88(8415%),21,7

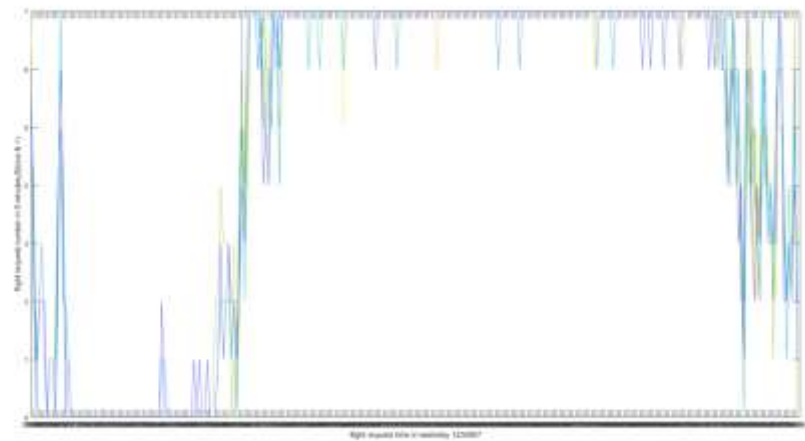


Figure 24. The 5minutes wave of capacity set as 88(84±5%),21,7

4.1.8 Test 8: three kind of capacity set as 88(84±5%),23(21±10%),7

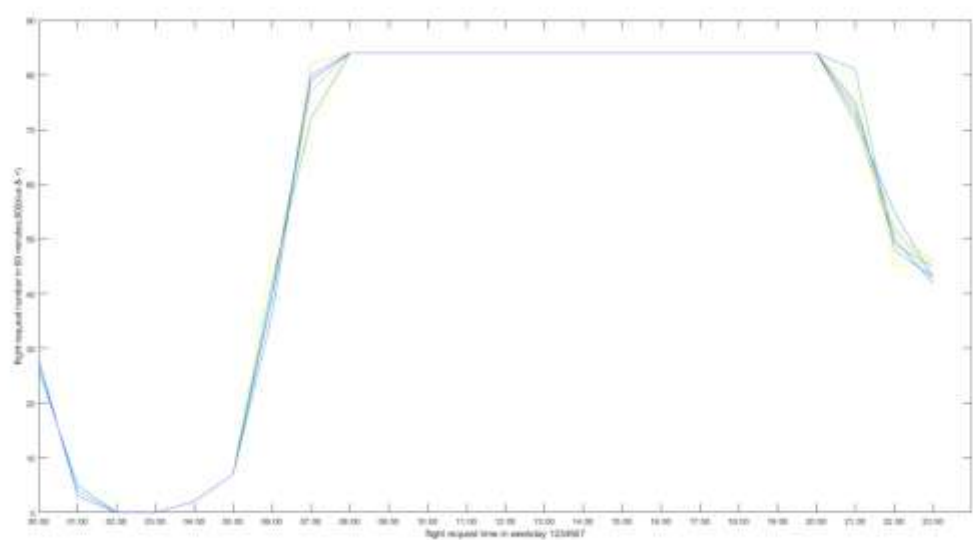


Figure 25. The hourly wave of capacity set as 88,23(21±10%),7

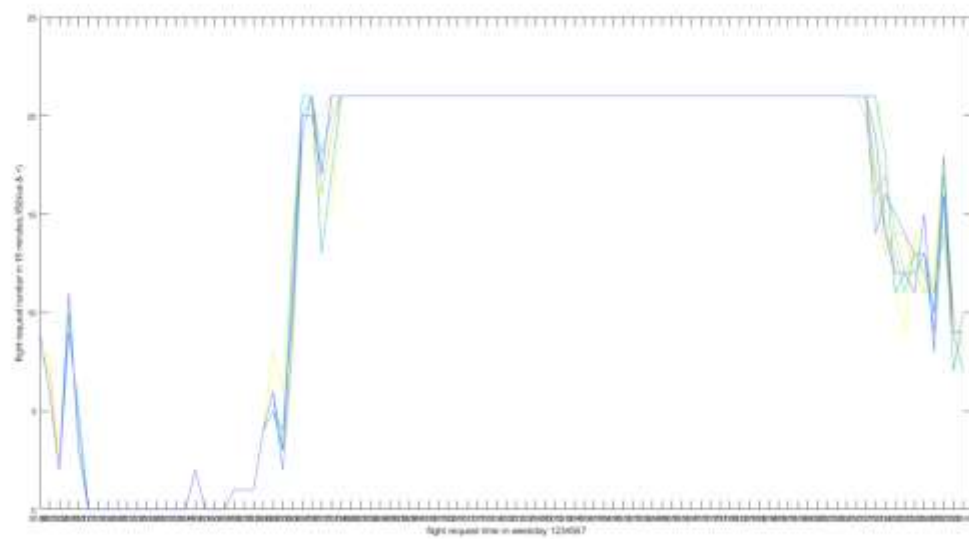


Figure 26. The 15minutes wave of capacity set as 88,23(21↑10%),7

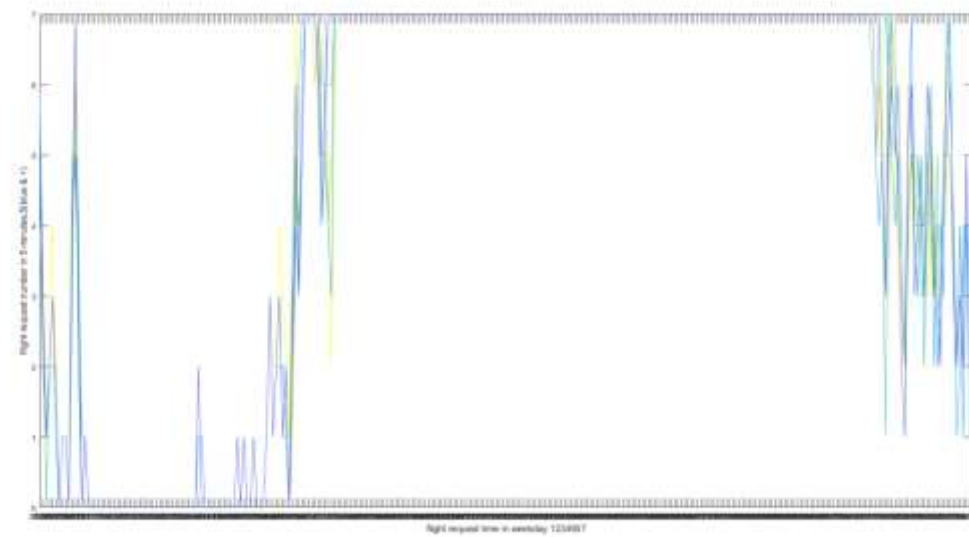


Figure 27. The 5minutes wave of capacity set as 88,23(21↑10%),7

4.1.9 Test 9: three kind of capacity set as 88(84↑5%),18(21↓14%),7

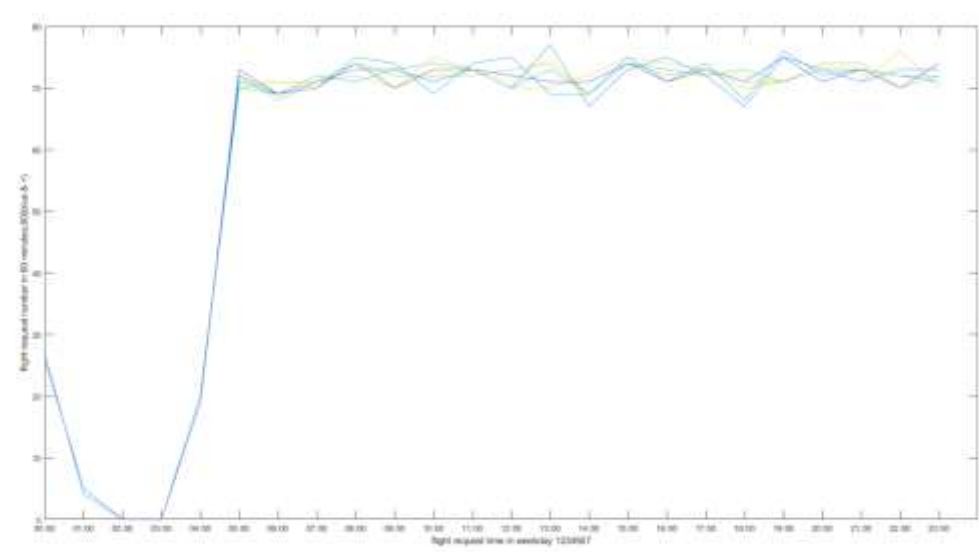


Figure 28. The hourly wave of capacity set as 88(84↑5%),18(21↓14%),7

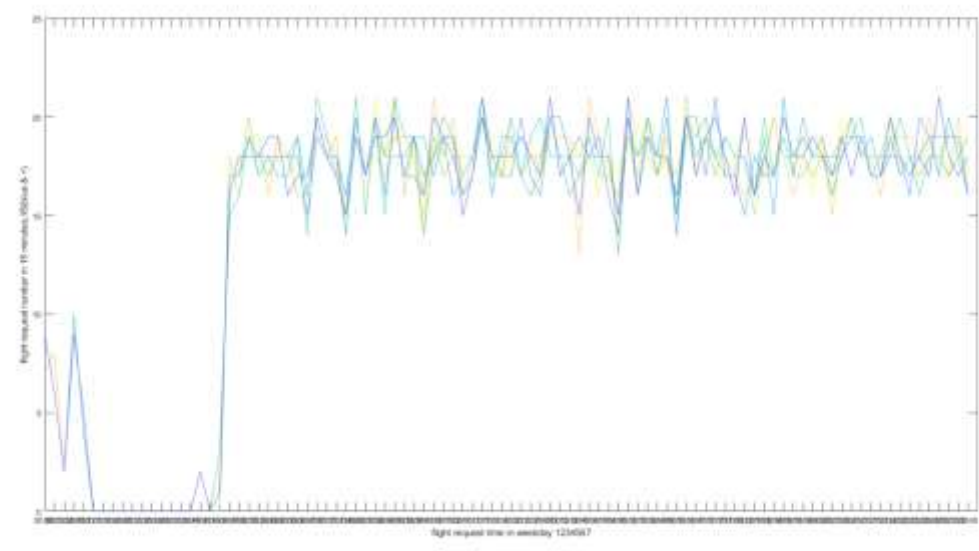


Figure 29. The 15minutes wave of capacity set as 88(84↑5%),18(21↓14%),7

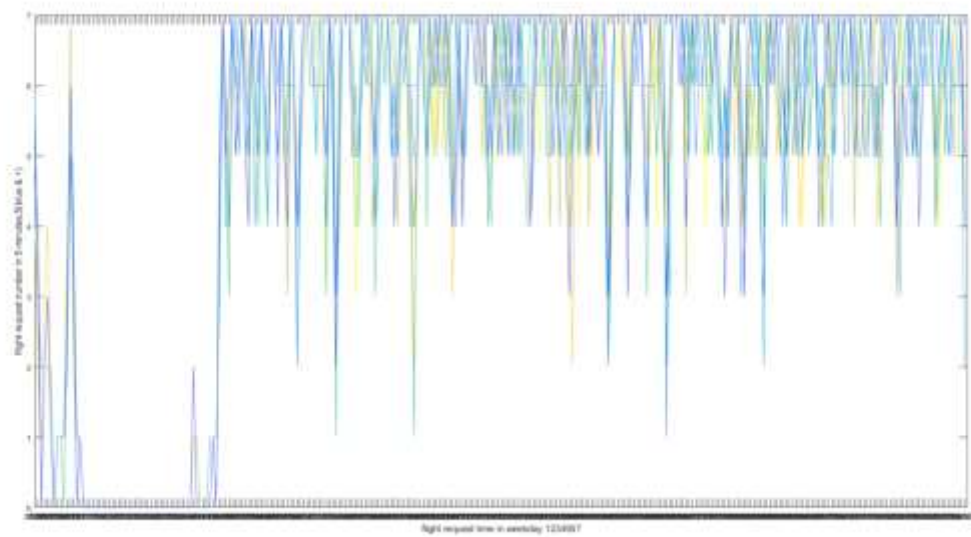


Figure 30. The 5minutes wave of capacity set as 88(84±5%),18(21±14%),7

4.2 Analysis of the effect of capacity setting on waveform of slot

From the results of test 1-8, we found that slot wave has specific characteristics under different capacity setting as following Table 4.

Table 4. Characteristics of slot wave under different capacity setting

Test	[C60, C15, C5] (U, V, W)	Relationship	Waveform characteristics	\bar{A}_1 Ψ_1 ω_1	\bar{A}_2 Ψ_2 ω_2	\bar{A}_3 Ψ_3 ω_3
1	[88,20,8] (88,80,96)	$V < U < W$ $C15 < R < Q$ $T < S < C5$	Lots of shock waves are capped by [U,Q,C5]	$\bar{A}_1 = V$ $\psi_1 \leq U - V$	$\bar{A}_2 = C15$ $\psi_2 \leq Q - C15$	$\bar{A}_3 = T$ $\psi_3 \leq C5 - T$
2	[60,20,8] (60,80,96)	$U < V < W$ $R < C15 < Q$ $S < T < C5$	Lots of shock waves are capped by [U,R,S]	$\bar{A}_1 = U$ $\psi_1 \leq V - U$	$\bar{A}_2 = R$ $\psi_2 \leq Q - R$	$\bar{A}_3 = S$ $\psi_3 \leq C5 - S$
3	[84,21,7] (84,84,84)	$U = V = W$ $R = C15 = Q$ $S = T = C5$	A few sawtooth waves are capped by [U, R, S]. The amplitude of this kind of wave is very small in busy period.	$\bar{A}_1 = U = V = W$ $\psi_1 = 0$ $\omega_1 \leq U$ $\omega_1 _{bp} \rightarrow 0$	$\bar{A}_2 = Q = R = C15$ $\psi_2 = 0$ $\omega_2 \leq R$ $\omega_2 _{bp} \rightarrow 0$	$\bar{A}_3 = T = C5$ $\psi_3 = 0$ $\omega_3 \leq S$ $\omega_3 _{bp} \rightarrow 0$
4	[76,21,7] (76,84,84)	$U < V = W$ $R < C5 = Q$ $S < T = C5$	Lots of shock waves are capped by [U,R,S]	$\bar{A}_1 = U$ $\psi_1 \leq V - U$	$\bar{A}_2 = R$ $\psi_2 \leq Q - R$	$\bar{A}_3 = S$ $\psi_3 \leq C5 - S$
5	[84,19,7] (84,76,84)	$V < U = W$ $C15 < R = Q$ $T < S = C5$	Lots of shock waves are capped by [W,Q,C5]	$\bar{A}_1 = V$ $\psi_1 \leq W - V$	$\bar{A}_2 = C15$ $\psi_2 \leq Q - C15$	$\bar{A}_3 = T$ $\psi_3 \leq C5 - T$
6	[84,23,7] (84,92,84)	$U = W < V$ $R = Q < C15$ $S = C5 < T$	A few sawtooth waves are capped by [U, R, S]. The amplitude of this kind of wave is very small in busy	$\bar{A}_1 = U$ $\psi_1 = 0$ $\omega_1 \leq U$ $\omega_1 _{bp} \rightarrow 0$	$\bar{A}_2 = R$ $\psi_2 = 0$ $\omega_2 \leq R$ $\omega_2 _{bp} \rightarrow 0$	$\bar{A}_3 = S$ $\psi_3 = 0$ $\omega_3 \leq S$ $\omega_3 _{bp} \rightarrow 0$

			period.			
7	[88,21,7] (88,84,84)	W=V<U C15=Q<R T=C5<S	A few sawtooth waves are capped by [W,Q,C5]. The amplitude of this kind of wave is very small in busy period.	$\bar{A}_1=W$ $\psi_1=0$ $\omega_1 \leq W$ $\omega_1 _{bp} \rightarrow 0$	$\bar{A}_2=Q$ $\psi_2=0$ $\omega_2 \leq Q$ $\omega_2 _{bp} \rightarrow 0$	$\bar{A}_3=C5$ $\psi_3=0$ $\omega_3 \leq C5$ $\omega_3 _{bp} \rightarrow 0$
8	[88,23,7] (88,92,84)	W<U<V Q<R<C15 C5<S<T	Very few sawtooth waves are capped by [W,Q, C5]. The amplitude of this kind of wave almost disappears in busy period.	$\bar{A}_1=W$ $\psi_1=0$ $\omega_1 \leq W$ $\omega_1 _{bp} \rightarrow 0$	$\bar{A}_2=Q$ $\psi_2=0$ $\omega_2 \leq Q$ $\omega_2 _{bp} \rightarrow 0$	$\bar{A}_3=C5$ $\psi_3=0$ $\omega_3 \leq C5$ $\omega_3 _{bp} \rightarrow 0$
9	[88,18,7] (88,72,84)	V<W<U C15<Q<R T<C5<S	Lots of shock waves are capped by [W,Q,C5]	$\bar{A}_1=V$ $\psi_1 \leq W-V$	$\bar{A}_2=C15$ $\psi_2 \leq Q-C15$	$\bar{A}_3=T$ $\psi_3 \leq C5-T$

*Notation: U= C60 (60min);V= C15×4(60 min);W= C5×12(60 min);Q= C5×3(15 min);R= C60÷4(15 min);S= C60÷12(5 min);T= C15÷3(5 min); \bar{A}_1 is Hourly wave oscillation axis; \bar{A}_2 is 15-minutes wave oscillates axis; \bar{A}_3 is 5-minutes wave oscillates axis; ψ_1 is upper half amplitude of hourly wave; ψ_2 is upper half amplitude of 15-minutes wave; ψ_3 is upper half amplitude of 5-minutes wave.

Bp: busy period

From the Table 4 we can see that wave oscillates around $\bar{A}_1 = \min[U, V, W]$, $\bar{A}_2 = \min[R, C15, Q]$, $\bar{A}_3 = \min[S, T, C5]$ and they are all capped by their second largest number.

Test 1-9 are classified according to the characteristic parameters of the wave and are shown in Table 5. The waveforms of class 1-3 are similar, but the characteristic parameters are slightly different. So do class 4-5. Class 3, 6, 7, 8 can be categorized into one group ($W \leq V$) and class 2, 4, 1, 5, 9 can be categorized into another group ($V \leq W$) when classification is based on the shape of the wave.

Table 5. Classification of tests according to characteristic parameters of the wave and grouping according to shape of the wave

Class	\bar{A}_1 ψ_1 ω_1	\bar{A}_2 ψ_2 ω_2	\bar{A}_3 ψ_3 ω_3	Waveform characteristics	Relationship	Test	group
1	$\bar{A}_1=U=V=W$ $\psi_1=0$ $\omega_1 \leq U$ $\omega_1 _{bp} \rightarrow 0$	$\bar{A}_2=Q=R=C15$ $\psi_2=0$ $\omega_2 \leq R$ $\omega_2 _{bp} \rightarrow 0$	$\bar{A}_3=T=C5$ $\psi_3=0$ $\omega_3 \leq S$ $\omega_3 _{bp} \rightarrow 0$	A few sawtooth waves are capped by [U, R, S]. The amplitude of this kind of wave is very small in busy period.	U=V=W	3	
2	$\bar{A}_1=U$ $\psi_1=0$ $\omega_1 \leq U$ $\omega_1 _{bp} \rightarrow 0$	$\bar{A}_2=R$ $\psi_2=0$ $\omega_2 \leq R$ $\omega_2 _{bp} \rightarrow 0$	$\bar{A}_3=S$ $\psi_3=0$ $\omega_3 \leq S$ $\omega_3 _{bp} \rightarrow 0$	A few sawtooth waves are capped by [U, R, S]. The amplitude of this kind of wave is very small in busy period.	U=W<V	6	$W \leq V$
3	$\bar{A}_1=W$ $\psi_1=0$ $\omega_1 \leq W$ $\omega_1 _{bp} \rightarrow 0$	$\bar{A}_2=Q$ $\psi_2=0$ $\omega_2 \leq Q$ $\omega_2 _{bp} \rightarrow 0$	$\bar{A}_3=C5$ $\psi_3=0$ $\omega_3 \leq C5$ $\omega_3 _{bp} \rightarrow 0$	A few sawtooth waves are capped by [W,Q,C5]. The amplitude of this kind of wave is very small in busy period.	W=V<U W<U<V	7,8	

				period.			
4	$\bar{A}_1=U$ $\psi_1\leq V-U$	$\bar{A}_2=R$ $\psi_2\leq Q-R$	$\bar{A}_3=S$ $\psi_3\leq C5-S$	Lots of shock waves are capped by [U,R,S]	$U<V<W$ $U<V=W$	2,4	
5	$\bar{A}_1=V$ $\psi_1\leq U-V$	$\bar{A}_2=C15$ $\psi_2\leq Q-C15$	$\bar{A}_3=T$ $\psi_3\leq C5-T$	Lots of shock waves are capped by [U,Q,C5] or [W,Q,C5]	$V<U<W$ $V<U=W$ $V<W<U$	1,5,9	$V<W$

*Notations are same as table 4.

5 Discussion

5.1 Memory consumption analysis

If the hourly capacity, 15-minutes capacity and 5-minutes capacity constraints are met at the same time, the capacity constraints in expression (9) must be based on Table 6. And expression (9) is expanded in the following three expressions (11-13).

Table 6. Coefficient Matrix using in Capacity constrains						
M	S	1	2	...	288	Variable constrains
	1	$x11$	$x12$...	$x1288$	1
	2	$x21$	$x22$...	$x2288$	1
	3	$x31$	$x32$...	$x3288$	1
	
m		$xm1$	$xm2$...	$xm288$	1

The 5-minutes capacity constraints are as following:

$$\sum_{i=1}^m x_{ij} \leq C_{-5}, j = 1,2 \dots 288 \tag{11}$$

The 15-minutes capacity constraints are as following:

$$\sum_{i=1}^m x_{ij} + \sum_{i=1}^m x_{i,j+1} + \sum_{i=1}^m x_{i,j+2} \leq C_{-15}, j = 1,2 \dots 286 \tag{12}$$

The hourly capacity constraints are as following:

$$\sum_{i=1}^m x_{ij} + \sum_{i=1}^m x_{i,j+1} + \dots + \sum_{i=1}^m x_{i,j+11} \leq C_{-60}, j = 1,2 \dots 276 \tag{13}$$

Obviously, if three kinds of capacity constrains must be met at the same time, the memory consumption will increase greatly, and the number of applications that can be fed at one time will greatly reduce. This implies that the optimization effect will be reduced, and the minimum value in [C60, C15×4, C5×12] will play a dominant role in these constraints.

On the contrary, the capacity constraints used in our algorithm are much simpler. Three kinds of capacity constraints are shown as following in the sequence hourly, 15 minutes and 5 minutes.

$$\sum_{i=1}^m x_{ij} \leq C_{-60}, j = 1,2 \dots 24 \tag{14}$$

$$\sum_{i=1}^m x_{ij} \leq C_{-15}, j = 1,2 \dots 96 \tag{15}$$

$$\sum_{i=1}^m x_{ij} \leq C_{-5}, j = 1, 2 \dots 288 \quad (16)$$

Data feeding method used in our algorithm not only simplifies the coding, but also reduces the dimension of constraints. This implies that the optimization effect will be increased, because of the fact that more applications can be fed to the program at a time. Therefore, by combining the previous analysis in section 3, it can be concluded that the data feeding method 1 is better than the second one, and it is the best of the five data feeding methods proposed in section 3.

Since there is no detailed description of data feeding method in previous articles, it is difficult to compare optimization effect with previous methods. But, based on the analysis of possible data splitting methods, we still could figure out which data feeding method is good or bad from the feedable data quantity. In the future research, we need to use test data to prove which data feeding method proposed in section 3 has the best optimization effect.

Our LIP algorithm was solved by MATLAB 2018. All tests were run on a desktop with 16 GB RAM and AMD RYZEN 5 1600 6-Core 3.2 GHz. According to table 3, using our LIP algorithm based on the first data feeding method, the average computing time was 1.36 hours per time with 9926 applications in 7 days, and was 12 minutes per time with 1418 applications per day. The calculation time includes the time to read and write excel files. The most time-consuming function in profile summary report for our LIP algorithm is the 'matrixconfig' function for constructing constraints matrix, as shown in Figure 31.

Function Nam	Calling Number	Total Time	Self-Use Time	Total Time Graph Dark Band:Self-Use Time
calcu newpek main 2018 08 04	1	4882.290 s	0.853 s	
matrixconfig	133	4427.547 s	4427.547 s	
dayi solve5	7	4309.164 s	0.052 s	
subassign5	63	4309.107 s	1.120 s	
int01quihua5	63	4307.911 s	7.823 s	
request adjust data plot731	1	306.276 s	0.181 s	
xlswrite	396	284.793 s	4.579 s	
xlswrite>Execute\Write	396	279.565 s	34.047 s	
iofun\private\openExcel\Workbook	397	239.310 s	100.280 s	
fifteenminutecapcheck	7	199.505 s	7.014 s	
write2excelltab new5 jieguo	1	193.785 s	0.159 s	
dayi solve anycap	7	191.221 s	0.050 s	
subassign 15	63	191.161 s	0.805 s	
int01quihua 15	63	190.308 s	2.544 s	
registerevent	397	74.058 s	0.057 s	
registerevent>addevent	397	73.455 s	73.455 s	
write2excelltab new15 jieguo	1	73.406 s	0.109 s	
winfun\private\comeventcallback	519673	64.865 s	55.309 s	
intlinprog	133	54.127 s	0.014 s	
...hAndCut>IntlinprogBranchAndCut.run	133	53.869 s	0.009 s	
slbiClient	133	53.556 s	0.066 s	
slbiClient>changeConstraintForm	133	45.589 s	45.458 s	
write2excelltab new60 last result	1	33.864 s	26.352 s	
write2excelltab new60 jieguo	1	21.246 s	0.091 s	

Figure 31. Profile summary report for our LIP algorithm

5.2 Effect of virtual capacity setting on slot waveform based on our the data feeding method

By observing the waveforms of these different capacity settings, we observed that according to our slot allocation algorithm based on the data feeding method 1, different settings of the three kinds of capacities have dominant effect on the waveform. Through nine tests and analyses in section 4, some interesting findings are as follows.

- A. The most important finding is that under the data feeding mode we choose, there is no shock wave in busy period when $W \leq V$, but when $W > V$, there will be shock wave in busy period.
- B. Three kinds of shock waves exist or do not exist at the same time with different capacity setting.
- C. Three kinds of shock waves oscillates around $\bar{A}_1 = \min[U, V, W]$, $\bar{A}_2 = \min[R, C15, Q]$, $\bar{A}_3 = \min[S, T, C5]$ and they are all capped by their second largest number.
- D. When there is a shock wave, the amplitude of hourly wave is proportional to $|U - V|$. Hourly wave oscillates around $\bar{A}_1 = \min[U, V, W]$. The 15-minutes wave oscillates around $\bar{A}_2 = \min[R, C15, Q]$. The 5-minutes wave oscillates around $\min[S, T, C5]$.

6 Conclusion

The data feeding method 1 is the best in five data feeding methods proposed in section 3. Computing time of our LIP algorithm is acceptable. The amplitude of the wave could be controlled to wanted range by the capacity setting. In this way, the rolling capacity constraints could be avoided and the computing time and complexity could be greatly reduced. It solves the problem that LIP is difficult to be used in large-scale slot allocation because of prohibited memory consuming.

Our study suggests that the proposed methods could be used to control and design the amplitude of the slot wave. This means that virtual capacity settings could be used to control demand within declared capacity. Our research also provides clues for further research on the construction of the flight waves to improve the hit rate[25]. It is hopeful to further improve the reliability of the flight waves for airlines operating at specific airports in order to establish direct or indirect connectivity, and increase hit rate by choosing preferred virtual capacity settings. In addition, the precise quantitative relationship between the period, the phase of waves and capacity setting needs to be further studied.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (grant number 71571182), the National Natural Science Youth Foundation of China (grant number 61603396), the National Natural Science Foundation of China and the Civil Aviation Grant (grant number U1633124) and the research projects of the social science, humanity on Young Fund of the ministry of Education of China (grant number 14YJC630185) and the Fundamental Research Funds for the Central Universities (grant number ZXH2011C009).

Reference

- [1] Zografos KG, Madasz MA, Salourasx Y. A decision support system for total airport operations management and planning. *J Adv Transport*. 2013;47(2):170-89.
- [2] Barnhart C, Fearing D, Odoni A, Vaze V. Demand and capacity management in air transportation. *EURO Journal on Transportation and Logistics*. 2012;1(1):135-55.
- [3] Jacquillat A, Odoni AR. An Integrated Scheduling and Operations Approach to Airport Congestion Mitigation. *Operations Research*. 2015;63(6):1390-410.
- [4] Zografos KG, Salouras Y, Madas MA. Dealing with the efficient allocation of scarce resources at congested airports. *Transportation Research Part C-Emerging Technologies*. 2012;21(1):244-56.

- [5] Zografos KG, Madas MA, Androutsopoulos KN. Increasing airport capacity utilisation through optimum slot scheduling: review of current developments and identification of future needs. *J Scheduling*. 2017;20(1):3-24.
- [6] Basso LJ, Zhang AM. Pricing vs. slot policies when airport profits matter. *Transport Res B-Meth*. 2010;44(3):381-91.
- [7] Verhoef ET. Congestion pricing, slot sales and slot trading in aviation. *Transport Res B-Meth*. 2010;44(3):320-9.
- [8] Zhang AM, Zhang YM. Airport capacity and congestion pricing with both aeronautical and commercial operations. *Transport Res B-Meth*. 2010;44(3):404-13.
- [9] Czerny AI, Zhang AM. Airport congestion pricing and passenger types. *Transport Res B-Meth*. 2011;45(3):595-604.
- [10] Le L, Donohue G, Hoffman K, Chen CH. Optimum airport capacity utilization under congestion management: A case study of New York LaGuardia airport. *Transport Plan Techn*. 2008;31(1):93-112.
- [11] Vaze V, Barnhart C. Modeling Airline Frequency Competition for Airport Congestion Mitigation. *Transportation Science*. 2012;46(4):512-35.
- [12] Jacquillat A, Odoni AR. Congestion Mitigation at John F. Kennedy International Airport in New York City: Potential of Schedule Coordination. *Transport Res Rec*. 2013(2400):28-36.
- [13] Pyrgiotis N, Odoni A. On the Impact of Scheduling Limits: A Case Study at Newark Liberty International Airport. *Transportation Science*. 2016;50(1):150-65.
- [14] Ribeiro NA, Jacquillat A, Antunes AP, Odoni AR, Pita JP. An optimization approach for airport slot allocation under IATA guidelines. *Transport Res B-Meth*. 2018;112:132-56.
- [15] Zografos KG, Androutsopoulos KN, Madas MA. Minding the gap: Optimizing airport schedule displacement and acceptability. *Transportation Research Part A: Policy and Practice*. 2018;114:203-21.
- [16] Jacquillat A, Odoni AR. Endogenous control of service rates in stochastic and dynamic queuing models of airport congestion. *Transportation Research Part E-Logistics and Transportation Review*. 2015;73:133-51.
- [17] Jacquillat A, Odoni AR. A roadmap toward airport demand and capacity management. *Transportation Research Part a-Policy and Practice*. 2018;114:168-85.
- [18] Benlic U. Heuristic search for allocation of slots at network level. *Transportation Research Part C-Emerging Technologies*. 2018;86:488-509.
- [19] Li X, Chen X, Li D, Wei D. Classification and characteristics of flights taking off and landing waveforms. *FLIGHT DYNAMICS*. 2016;34(02):90-4.
- [20] Sun CL, Su X. Analysis & Optimization of Beijing Capital Airport's flight waves. *China Civil Aviation*. 2013(10):30-1.
- [21] Hu M, Yi T, Ren Y. Optimization of airport slot based on improved Hungarian algorithm. *Application Research of Computers*. 2019(08):1-7.
- [22] Huang J, Wang J. A comparison of indirect connectivity in Chinese airport hubs: 2010 vs. 2015. *Journal of Air Transport Management*. 2017;65:29-39.
- [23] Jacquillat A, Vaze V. Interairline Equity in Airport Scheduling Interventions. *Transportation Science*. 2018;52(4):941-64.
- [24] Ye ZL, Y.; Bai, J.; Zheng, X. Performance Comparing and Analysis for Slot Allocation Model. *Preprints* 20192019.
- [25] Goedecking P. Networks in aviation: Strategies and structures. *Expert Opinion on Biological Therapy*. 2010;8(2):213.

