MATH3082 Optimisation Coursework 2019-20

Rohan Bungre

29466423

1 Integer Linear Programming

Linear programming is a method to maximise or minimise a mathematical model represented by a linear relationship. More formally, linear programming is a technique for the optimisation of a linear objective function, subject to linear equality and linear inequality constraints.

$$\begin{aligned} \max z = 5x_1 + 6x_2 \\ x_1 + x_2 \leq 5 \\ 4x_1 + 7x_2 \leq 28 \\ x_1, x_2 \geq 0 \end{aligned} \tag{1}$$

Equation 1 shows an example model, where the optimal values of x_1, x_2 can be found to maximise the objective function. Using a simplex method the solution to equation 1 can be found to be $x_1 = 7/3$ and $x_2 = 8/3$, giving an objective function maximum value of $z^* = 27.7$. Figure 1 shows the graphing method of solving the optimisation problem.

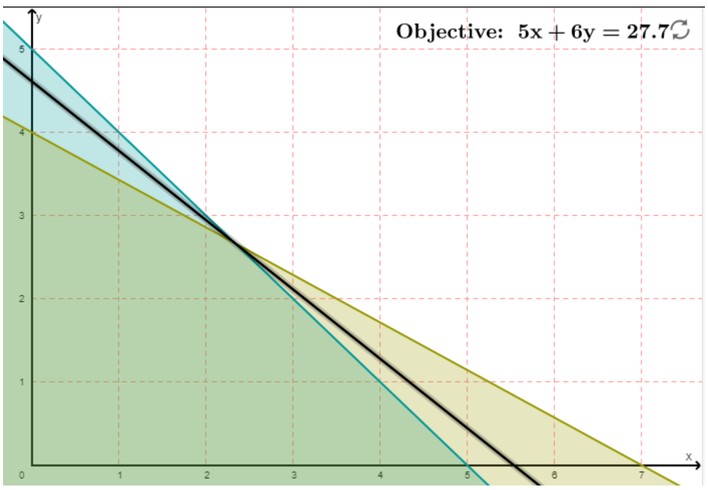

Figure 1: Graphing method of linear programming

In equation 1, the variables x_1, x_2 are free to be any number ≥ 0. However, in some cases the variables must be constrained to an integer value because they represent a discrete unit, such as the number of staff in a workplace. Solving the problem when

$x_1, x_2 \geq 0$ is simple using the simplex method, however, when x_1, x_2 are required to be integer values additional steps are required.

The trivial solution would be to try every integer pair within the feasible region but this is not scalable with a large number of variables. Another solution could be to solve the optimisation problem normally, obtain non integer results and then round them down, however this does not guarantee to produce the optimal solution. As figure 2 demonstrates, the rounded integer values of $x_1 = 2$, $x_2 = 2$ gives $z = 22$ however, the optimal integer values of $x_1 = 3$, $x_2 = 2$ gives $z^* = 27$.

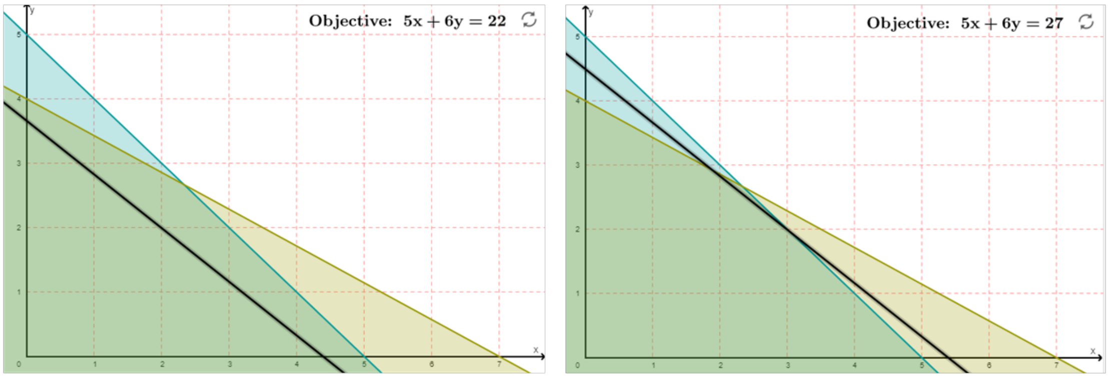

Figure 2: Rounding down vs optimal integer solution

$$\begin{aligned} \max z = 5x_1 + 6x_2 \\ x_1 + x_2 \leq 5 \\ 4x_1 + 7x_2 \leq 28 \\ x_1, x_2 \geq 0 \\ x_1, x_2 \text{ are integers} \end{aligned} \tag{2}$$

Equation 2 shows the updated formulation of the integer linear programming problem. To solve integer linear programming problems there are two methods available.

1. Branch and Bound Methods
2. Cutting Plane Methods

2 Branch and Bound Methods

The branch and bound method aims to split the feasible region into several sub-regions hoping to find an integer solution within one of them. Sometimes, the sub-regions must be split further if no integer solution is found.

To solve equation 2, the first step is to drop the requirement that all variables must be integers. This is known as the linear programming relaxation and forms the original equation 1. The linear programming relaxation will be used to estimate the optimal solution of the integer programming problem.

As shown in chapter 1, the solution to the linear relaxation is the solution to equation 1, which is $x_1 = 2.33$ and $x_2 = 2.67$, giving an objective function maximum value of $z^* = 27.7$. Choosing the variable to split the feasible region on is up for debate amongst academics. For this example x_2 is selected as the variable to split the feasible region on as is contains the larger decimal proportion.

As $x_2 = 2.67$, it can be bound by $x_2 \leq 3$ and $x_2 \geq 2$ for an integer solution, this causes two possible branches in the feasible region. Recalculating the linear relaxation problem using the simplex method with the extra bound constraints, the optimal solution can be calculated for each new sub region. Figure 3 shows this process using a graphical method.

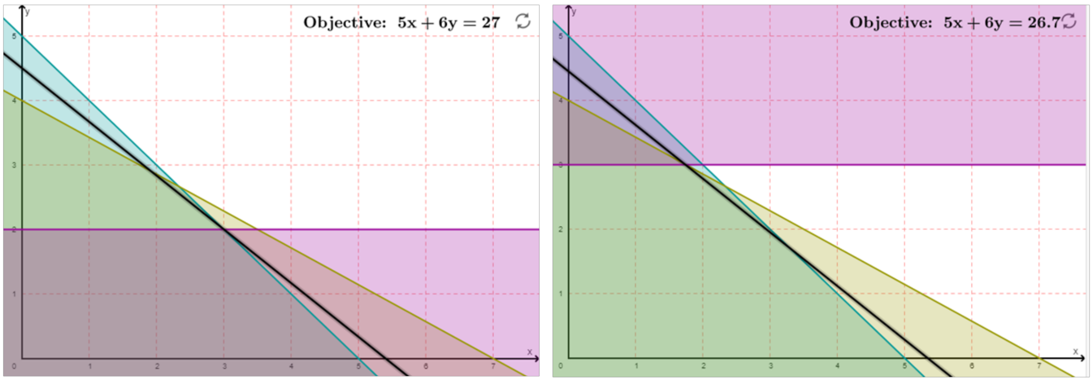

Figure 3: Comparing sub-region 2 with sub-region 3

Figure 4 shows the branching and bounding, stored in an enumeration tree. Region 2 has found an integer solution so it would not branch again. To ensure the optimal integer solution was found, Region 3 would branch and the process would iterate until no branching could happen, therefore all integer solutions had been found. Region 2 found the optimal integer programming solution of $x_1 = 3$, $x_2 = 2$ and $z^* = 27$.

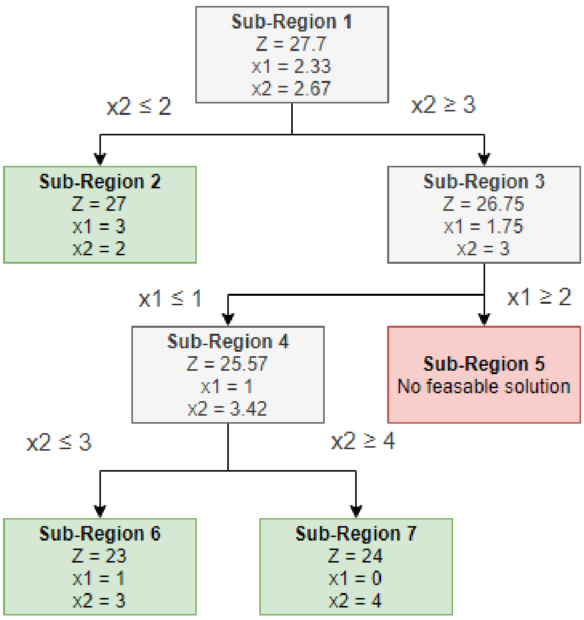

Figure 4: Enumeration tree for branch and bound methods

3 Cutting Plane Methods

The cutting plane method aims to solves integer linear programming problems by modifying the linear relaxation solutions until an integer solution is found. Unlike the branch and bound approach, it does not split the feasible region into sub-regions. It instead tries to refine a single linear problem, by adding new constraints.

These new constraints act as plane, cutting off parts of the feasible region that do not contain an integer solution. This successively reduces the feasible region until an integer optimal solution is found, demonstrated in figure 5. In practice, the branch-and-bound method always outperforms the cutting-plane algorithm.

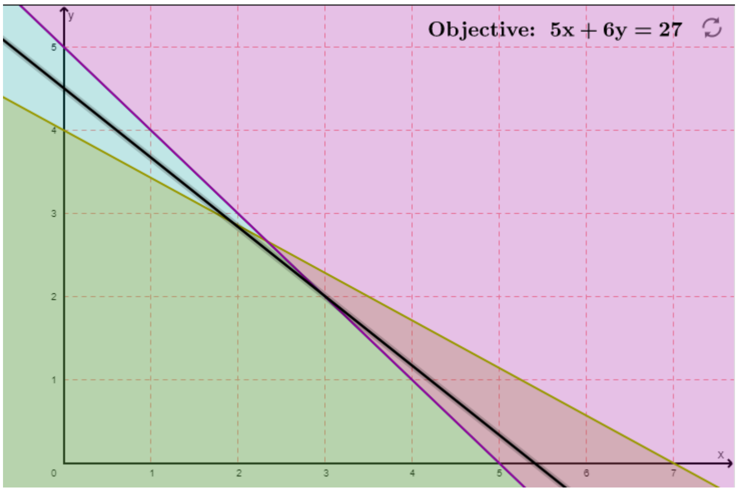

Figure 5: Cutting plane method shown graphically

An integer linear problem shown in equation 2 can be solved using the cutting plane method by first solving its linear relaxation formulation as shown in equation 1 using a simplex method. This produces a tabular shown in table 1.

BV	**x1**	**x2**	**s1**	**s2**	**RHS**
x1	1	0	7/3	-1/3	7/3
x2	0	1	-4/3	1/3	8/3
Z	0	0	11/3	1/3	83/3

Table 1: Original simplex tabular

The next step is to pick one of the basic variables that has a non-integer solution in the RHS. In the case of equation 3, the x_1 row was chosen. The equation from the row needs to be written out in a form so that the fractional values are written as the summation of an integer value and a fractional portion. This is demonstrated by equation 3.

$$\begin{aligned} x_1 + \frac{7}{3}s_1 - \frac{1}{3}s_2 = \frac{7}{3} \\ x_1 + (2 + \frac{1}{3})s_1 + (-1 + \frac{2}{3})s_2 = 2 + \frac{7}{3} \end{aligned} \tag{3}$$

To create the cutting plane equation, equation 3 needs to be rearranged so that anything with an integer coefficient is moved to the LHS and anything with fractional coefficients is moved to the RHS. This is demonstrated by equation 4.

$$x_1 + 2s_1 - s_2 - 2 = -\frac{1}{3}s_1 - \frac{2}{3}s_2 + \frac{1}{3} \tag{4}$$

As the slack variables s_1 and s_2 are always ≥ 0, this means that the RHS of equation 4 will always be $\leq \frac{1}{3}$. As the LHS variables will only take integer values the $\leq \frac{1}{3}$ will be rounded to its nearest integer value of ≤ 0. This leaves the cutting plane equation, shown in equation 5.

$$-\frac{1}{3}s_1 - \frac{2}{3}s_2 + \frac{1}{3} \leq 0 \tag{5}$$

This cutting plane equation can now become a new constraint for the optimal tabular in table 1. Placing the constraint in tabular produces table 2.

BV	**x1**	**x2**	**s1**	**s2**	**s3**	**RHS**
x1	1	0	7/3	-1/3	0	7/3
x2	0	1	-4/3	1/3	0	8/3
s3	0	0	-1/3	-2/3	1	-1/3
Z	0	0	11/3	1/3	0	83/3

Table 2: Simplex tabular with new cutting plane constraint

The dual simplex method can now solve table 2 and the previous steps can iterate until integer solutions are found. Table 3 shows the final tabular producing the same optimal integer solution as the graphical and branch and bound methods.

BV	**x1**	**x2**	**s1**	**s2**	**s3**	**s4**	**RHS**
x1	1	0	3	0	0	-1	3
x2	0	1	-2	0	0	1	2
s1	0	0	2	1	0	-3	2
s2	0	0	1	0	1	-2	1
Z	0	0	3	0	0	1	**27**

Table 3: Simplex tabular with optimal cutting plane constraints

The final tabular found the optimal integer programming solution of $x_1 = 3$, $x_2 = 2$ and $z^* = 27$. The cutting plane method can be used in conjunction with the branch and bound method, to help reduce the number of branches required. The cutting plane will remove non-integer solutions before branching and bounding needs to be performed, reducing the number of interactions required to find an optimal solution.

4 Staff Scheduling Task

The aim of this task is to minimise the labour cost of a cafeteria, whilst ensuring there are enough staff to maintain customer satisfaction.

The important information and assumptions made:

- The cafe is open from 9am-5pm which includes 8 hour long periods.
- A full time employee works for 7 hour long periods plus an additional hour long break.
- A part time employee works for 4 consecutive hour long periods with no breaks.
- A full time employee is paid £12 per hour and is paid for their break totaling £96 per day
- A part time employee is paid £7.5 per hour totaling £30 per day.
- There must be at least 4 full time employees hired per day.

To form a model that would produce the optimal results, the full time workers would have to be able take their break at any point during the day. The same applies for the 4 hour blocks that part time workers would have to work.

As there were 8 different periods that full time workers could have their break in, there would be 8 different types of full time workers $F_1 \rightarrow F_8$. As there were 5 different ways of working 4 consecutive hours, there were 5 different types of part time employees $P_1 \rightarrow P_5$.

This could be formulated as an integer linear programming problem with a model shown in equation 6, where A is the decision matrix; x is the vector of basic variables $F_1 \rightarrow F_8$ and $P_1 \rightarrow P_5$; and b is the vector of constraints.

$$\begin{aligned} \min Z = 96\sum_{n=1}^{8} F_n + 30\sum_{n=1}^{5} P_n \\ \text{subject to } Ax \leq b \\ x \geq 0 \text{ as integers} \\ \sum_{n=1}^{8} F_n \geq 4 \end{aligned} \tag{6}$$

Equation 6 is able to model the problem of minimising the labor cost per day, whilst having enough staff per hour period; having at least 4 full time staff, ensuring the optimal result.

Using Microsoft Excel and its linear programming solver, the optimal integer solution to equation 6 was found. Microsoft Excel was used because of the University closures and not having reliable access to the Xpress-IVE software.

Figure 6 shows the Excel model, which allows for a visualisation of the decision matrix A (shown in white), the constraints vector b (shown in yellow) and the basic variables vector x (shown in green).

Firstly, the solution found that there was no over-staffing issues with the current schedule. The number of staff required per period is equal to the staff hired per period, meaning customers will remain content.

Secondly, the solution found that four full time workers must be hired, with two taking their lunch break from 3pm-4pm and the other two taking their lunch break from 10am-11am. Although these are not ideal lunch hours, it does provide the optimal solution.

Finally, the solution found that 6 part time worker must be hired, with two working from 9am-1pm; one working from 10am-2pm; another one working from 12noon-4pm and the final two working from 1pm-5pm.

	Type of Staff	F1	F2	F3	F4	F5	F6	F7	F8	P1	P2	P3	P4	P5			
Period	# Per Type	0	2	0	0	0	0	2	0	2	1	0	1	2	# Per Period		# Required
9am-10am		1	1	1	1	1	1	1	0	1	0	0	0	0	6	>=	6
10am-11am		1	1	1	1	1	1	0	1	1	1	0	0	0	5	>=	5
11am-12noon		1	1	1	1	1	0	1	1	1	1	1	0	0	7	>=	7
12noon-1pm		1	1	1	1	0	1	1	1	1	1	1	1	0	8	>=	8
1pm-2pm		1	1	1	0	1	1	1	1	0	1	1	1	1	8	>=	8
2pm-3pm		1	1	0	1	1	1	1	1	0	0	1	1	1	7	>=	7
3pm-4pm		1	0	1	1	1	1	1	1	0	0	0	1	1	5	>=	5
4pm-5pm		0	1	1	1	1	1	1	1	0	0	0	0	1	6	>=	6
	Cost Per Day	96	96	96	96	96	96	96	96	30	30	30	30	30			

Figure 6: Staff scheduling model in Excel

Figure 7 shows the minimised labour cost to be £564 per day, when hiring the 10 staff in the proposed manner. Due to the flexibility of being able to decided when each employee could work and take breaks, it allowed for the most optimal integer solution given the provided constraints.

Type of Staff	# Hours	Salery/Hour
Full Time (F1-F8)	8	12
Part Time (P1-P5)	4	7.5
Labour Cost/Day	# Full Time	#Part Time
564	4	6

Figure 7: Minimum labour cost and number of staff hired per day

Most full time employees would probably prefer to have their lunch break during the hours of 12noon-1pm or 1pm-2pm. By constructing a new model in Excel, following the same constraints as equation 6, with an updated decision matrix A, and basic variable vector x, this can be investigated.

Figure 8 shows the updated Excel model, which has a much simpler formation due to the smaller number of basic variables. Now that the employees can only have a lunch break during the typical lunch hours of 12noon-1pm or 1pm-2pm, there is an over-staffing problem.

During those lunch break hour, more part time employees must be hired to maintain customer satisfaction. This has caused 3 of the hour periods, 10am-11am, 11am-12noon and 3pm-4pm to be overstaffed.

	Type of Staff	F1	F2	P1	P2	P3	P4	P5			
Period	# Per Type	2	2	2	3	1	0	2	# Per Period		# Required
9am-10am		1	1	1	0	0	0	0	6	>=	6
10am-11am		1	1	1	1	0	0	0	9	>=	5
11am-12noon		1	1	1	1	1	0	0	10	>=	7
12noon-1pm		0	1	1	1	1	1	0	8	>=	8
1pm-2pm		1	0	0	1	1	1	1	8	>=	8
2pm-3pm		1	1	0	0	1	1	1	7	>=	7
3pm-4pm		1	1	0	0	0	1	1	6	>=	5
4pm-5pm		1	1	0	0	0	0	1	6	>=	6
	Cost Per Day	96	96	30	30	30	30	30			

Figure 8: Updated staff scheduling model in Excel

Figure 9 shows that the updated minimum labour cost per day has increased to £624. This is caused by the additional 2 part time workers having to be hired to cover the full time staff whilst they were on a lunch break.

These results show the trade off that many businesses have to make, between minimising labour costs and maintaining employee satisfaction.

Type of Staff	# Hours	Salery/Hour
Full Time (F1-F2)	8	12
Part Time (P1-P5)	4	7.5
Labour Cost/Day	# Full Time	#Part Time
624	4	8

Figure 9: Updated minimum labour cost and number of staff hired per day

Further investigations could look into part time employees having flexible working hours with potentially some being shorter, as that would have helped reduce labour costs. Another investigation could look at reducing the number of full-time staff required as the part-time staff gain experience. This would save a lot more money for the cafe.