5 Integer Programming

We shall consider a linear programming problem which has been given the additional constraint that the variables must take integer values. We will assume that all numbers appearing in the statement of the problem are integers: if the objective function or one or more of the constraints have coefficients which are fractions, then we can multiply each such objective function/constraint by a suitable integer (for example the least common multiple of the denominators of the fractions appearing in it), so that all coefficients become integers. Thus we shall consider the following problem.

P: Find $\mathbf{x} \in \mathbf{Z}^{\mathbf{n}}$ to maximise $z = \mathbf{c}^{\mathbf{T}}\mathbf{x}$ subject to $\mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}$, where $\mathbf{c} \in \mathbf{Z}^n$, $\mathbf{b} \in \mathbf{Z}^m$, and $\mathbf{A}$ is an $m \times n$ matrix with integer entries.

The *linear relaxation* of P is the linear programming problem P_1 we obtain by ignoring the constraint of P that the variables should take integer values. This gives the problem:

P_1: Find $\mathbf{x} \in \mathbf{R}^{\mathbf{n}}$ to maximise $z = \mathbf{c}^{\mathbf{T}}\mathbf{x}$ subject to $\mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}$.

Clearly, every feasible solution of P is also a feasible solution of P_1. Hence the optimal value of z in P_1 is a lower bound for the optimal value of z in P.

5.1 Graphical Solution for problems with two variables

5.1.1 Example

Find $x_1, x_2 \in Z$ to maximise $z = 2x_1 + x_2$ subject to

$$x_1 + x_2 \leq 7, x_1 - 2x_2 \leq 2, x_1 \geq 0, x_2 \geq 0.$$

The feasible region for this problem is shown in Figure 1.

The family of lines $2x_1 + x_2 = k$ move through the feasible region as k increases. Thus the optimal solution to the linear relaxation of the problem is given by the point $B = (16/3, 5/3)$. The optimal solution to the integer programming problem is given by the last point with integer co-ordinates in the feasible region which lies on one of these lines. Thus the optimal integer solution occurs at the point $A = (5, 2)$ and the maximum value of $z = 2x_1 + x_2$ is 12.

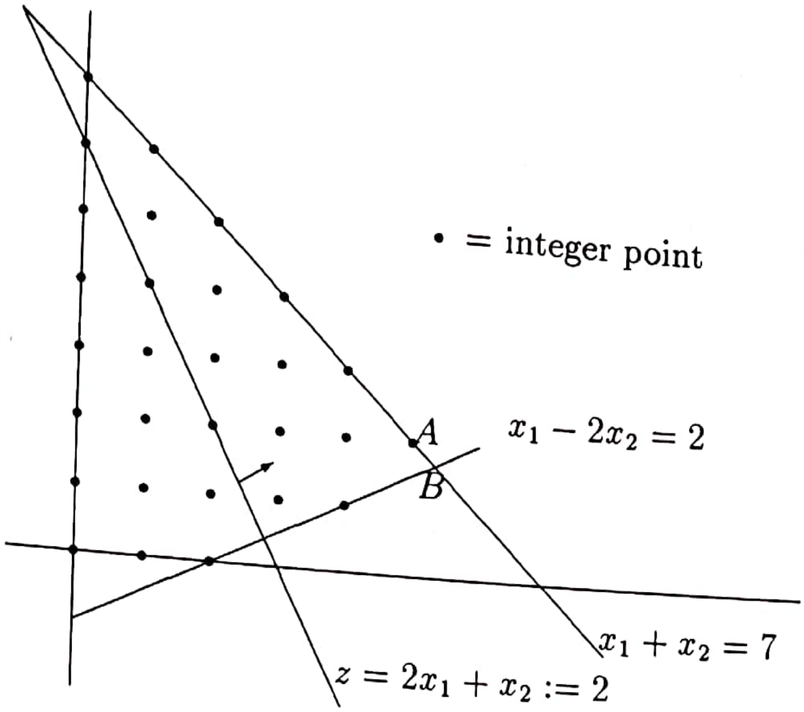


Figure 1: the optimal integer solution occurs at A while the optimal solution to the linear relaxation occurs at B.

5.1.2 Note

In Example 5.1.1, the optimal integer solution is the closest integer point A to the optimal fractional solution, B. In general, however, the optimal integer solution can lie a long way from the optimal fractional solution. In Figure 2, the optimal integer solution occurs at C while the optimal solution to the linear relaxation occurs at D.

5.1.3 Note

The graphical method for solving integer programming problems with two variables is no more difficult than the corresponding method for linear programming problems. However it also indicates that integer programming may be more difficult in higher dimensions since we no longer have a nice algebraic characterization of the optimal solution as an extreme point of the feasible region. Indeed, the general integer programming problem is known to be NP-complete so there is probably no fast (polynomial) algorithm which will solve all integer programming problems.

There are, however, many known algorithms for solving integer programming problems (which in a worst case, may take an exponential amount of

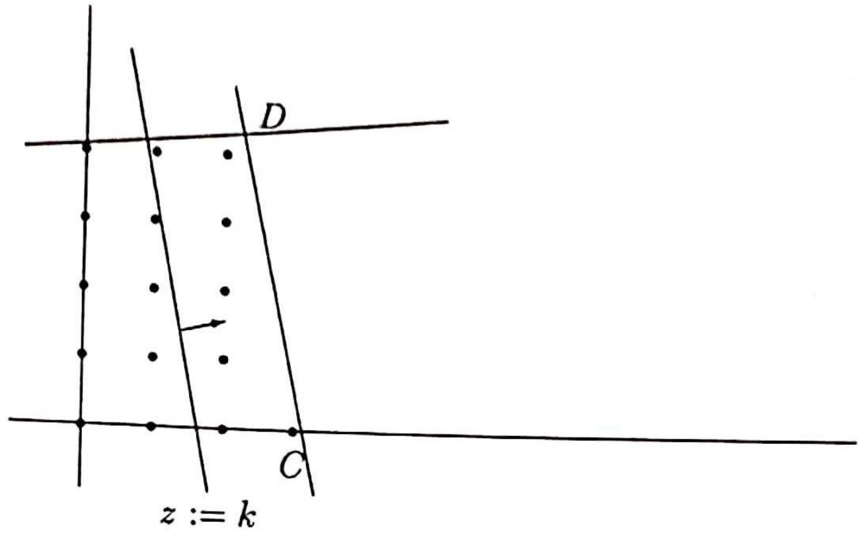


Figure 2: the optimal integer solution occurs at C while the optimal solution to the linear relaxation occurs at D.

running time to solve a particular problem). We shall consider two such algorithms.

5.2 Gomory's Cutting Plane Algorithm

Suppose we want to solve the integer programming problem:

P: Find $\mathbf{x} \in \mathbf{Z^n}$ to maximise $z = \mathbf{c^T x}$ subject to $\mathbf{Ax} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}$,
where $\mathbf{c} \in \mathbf{Z}^n$, $\mathrm{b} \in \mathbf{Z}^m$, and $\mathbf{A}$ is an $m \times n$ matrix with integer entries.

We first use the simplex algorithm to solve the linear relaxation P_1 of P. If the optimal solution to P_1 is an integer solution, then it is also the optimal integer solution to P. If not then we add a new constraint, called the *cutting plane constraint*, to P_1, to form a new linear programming problem P_2. This reduces the size of the feasible region and makes the old optimal solution infeasible. However, the cutting plane constraint is chosen such that it preserves all integer feasible solutions to P_1 (and hence all feasible solutions to P). We then solve P_2 using the Dual Simplex Algorithm starting from the final feasible dictionary for P_1. If the new optimal solution has integer values then it gives the optimal solution to the integer programming problem. If not we add another cutting plane constraint. We continue this process until we find an integer optimal solution. It can be shown that the algorithm will find the optimal integer solutions after a finite (but possibly exponential) number of iterations.

Solution to Example 5.1.1

To solve the linear realaxation P_1 of this problem we first add slack variables and obtain the following feasible dictionary.

$$\begin{array}{rcrcrcr} x_3 & = & 7 & - & x_1 & - & x_2 \\ x_4 & = & 2 & - & x_1 & + & 2x_2 \\ z & = & & - & 2x_1 & - & x_2 \end{array}$$

Note that the above dictionary implies that x_3, x_4 are integer valued whenever x_1, x_2 are integer valued. Using the simplex algorithm to solve P_1, we obtain the following feasible dictionary:

$$\begin{array}{rcrcrcr} x_1 & = & \frac{16}{3} & - & \frac{2}{3}x_3 & - & \frac{1}{3}x_4 \\ x_2 & = & \frac{5}{3} & - & \frac{1}{3}x_3 & + & \frac{1}{3}x_4 \\ z & = & \frac{37}{3} & - & \frac{5}{3}x_3 & - & \frac{1}{3}x_4 \end{array} \tag{9}$$

Thus the optimal solution to P_1 is $x_1 = \frac{16}{3} = 5\frac{1}{3}, x_2 = \frac{5}{3} = 1\frac{2}{3}$. Choose the basic variable which has the largest fractional part in this optimal solution. (The *fractional part* of a real number r is given by $\{r\} = r - \lfloor r \rfloor$, thus for example $\{5\frac{1}{3}\} = 5\frac{1}{3} - 5 = \frac{1}{3}$ and $\{-\frac{1}{3}\} = -\frac{1}{3} - (-1) = \frac{2}{3}$.) We have $\{\frac{15}{3}\} = \frac{1}{3}$ and $\{\frac{5}{3}\} = \frac{2}{3}$ so we choose x_2.

We next rewrite the x_2-equation in (9) by first expressing it in the form $x_2 = \frac{5}{3} - \mathbf{a}^{\mathbf{T}}\mathbf{x}_{\mathbf{N}}$ and then writing each coefficient a_i on the right hand side as a sum of its integer and fractional parts i.e. $a_i = \lfloor a_i \rfloor + \{a_i\}$. This gives:

$$\begin{aligned} x_2 &= \tfrac{5}{3} - \tfrac{1}{3}x_3 - (-\tfrac{1}{3})x_4 \\ &= (1 + \tfrac{2}{3}) - (0 + \tfrac{1}{3})x_3 - (-1 + \tfrac{2}{3})x_4. \end{aligned}$$

We next gather the fractional parts on the left hand side of this equation and the integer parts on the right hand side:

$$-\tfrac{2}{3} + \tfrac{1}{3}x_3 + \tfrac{2}{3}x_4 = 1 - x_2 + x_4 \tag{10}$$

Now for each INTEGER solution to P_1, the right hand side of (10) is an integer. Thus the left hand side of (10) must also be an integer. On the other hand, since $x_3 \geq 0$ and $x_4 \geq 0$, the left hand side of (10) is at least $-\frac{2}{3}$. Thus $-\frac{2}{3} + \frac{1}{3}x_3 + \frac{2}{3}x_4$ is an integer which is at least $-\frac{2}{3}$ so we must have

$$-\tfrac{2}{3} + \tfrac{1}{3}x_3 + \tfrac{2}{3}x_4 \geq 0 \tag{11}$$

for all integer solutions to the problem. We can hence add the constraint (11) to P_1 without loosing any integer solutions. Let the new linear programming problem be P_2. Since the optimal solution $x_1 = 16/3, x_2 = 5/3, x_3 = 0 = x_4$ to P_1 does not satisfy (11), it is not a feasible solution to P_2.

Adding the constraint (11) to P_1 gives rise to a new slack variable x_5 given by

$$x_5 = -\tfrac{2}{3} + \tfrac{1}{3}x_3 + \tfrac{2}{3}x_4. \tag{12}$$

Note that equation (10) gives $x_5 = 1 - x_2 - x_4$. Since whenever x_1, x_2 are integer valued, x_3, x_4 are also integer valued, we deduce that x_5 is also integer valued for every integer solution to P_1.

We may solve P_2 by adding the equation (12) to the feasible dictionary (9) and using the Dual Simplex Algorithm to move to the new optimal solution. We choose x_5 as leaving variable and x_4 as entering variable giving

$$\begin{array}{rcrcrcr} x_4 & = & 1 & - & \frac{1}{2}x_3 & + & \frac{3}{2}x_5 \\ x_1 & = & 5 & - & \frac{1}{2}x_3 & - & \frac{1}{2}x_5 \\ x_2 & = & 2 & - & \frac{1}{2}x_3 & + & \frac{1}{2}x_5 \\ z & = & 12 & - & \frac{3}{2}x_3 & - & \frac{1}{2}x_5 \end{array}$$

This has an integer optimal solution and so also gives the optimal solution for the original integer programming problem P: $x_1 = 5$, $x_2 = 2$ and the maximum value of z is 12.

Geometric Interpretation of the Cutting plane

The cutting plane constraint $-\frac{2}{3} + \frac{1}{3}x_3 + \frac{2}{3}x_4 \geq 0$ is equivalent, by (10), to $1 - x_2 + x_4 \geq 0$. Since the slack variable x_4 is originally defined by $x_4 = 2 - x_1 + 2x_2$, the cutting plane constraint is

$$1 - x_2 + (2 - x_1 + 2x_2) \geq 0,$$

giving $x_1 - x_2 \leq 3$. Thus the addition of the cutting plane constraint can be illustrated graphically as in Figure 3.

Summary of Gomory's Cutting Plane Algorithm

Suppose we want to solve:

P: Find $\mathbf{x} \in \mathbf{Z}^\mathbf{n}$ to maximise $z = \mathbf{c}^\mathbf{T}\mathbf{x}$ subject to $\mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}$,
where $\mathbf{c} \in \mathbf{Z}^n$, $\mathbf{b} \in \mathbf{Z}^m$, and $\mathbf{A}$ is an $m \times n$ matrix with integer entries.

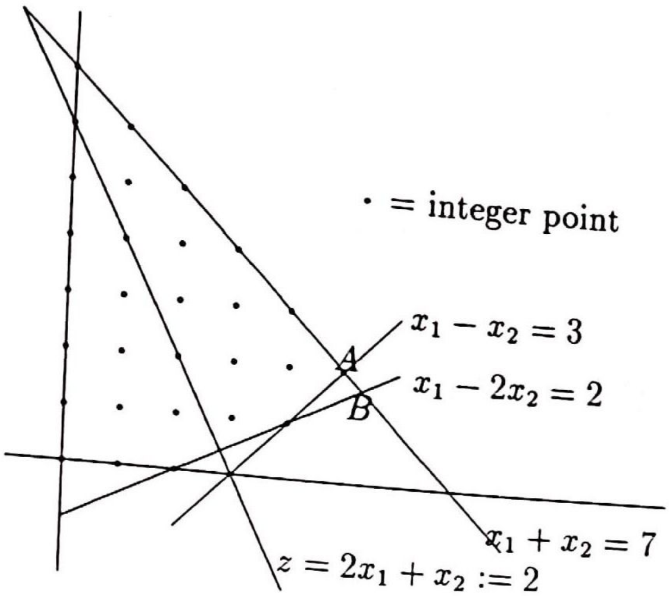


Figure 3: the addition of the cutting plane constraint $x_1 - x_2 \leq 3$ to P_1 gives a linear programming problem P_2 for which the optimal solution is the integer point A.

We do this by solving a sequence of linear programming problems P_1, P_2, $\ldots$, P_m, where P_1 is the linear relaxation of P, and P_{i+1} is obtained by adding a single constraint, the *cutting plane constaint* to P_i for $1 \leq i \leq m-1$. Each integer feasible solution to P_i remains an integer feasible solution to P_{i+1}, but the optimal solution to P_i is no longer a feasible solution to P_{i+1}. The algorithm terminates when one of the linear programming problems P_i has an optimal solution in which the variables have integer values.

Step 1 We first solve the linear relaxation P_1 of P.

Step 2 Suppose we have solved the linear programming problem P_r for some $r \geq 1$. If the optimal solution, $\mathbf{x} = \mathbf{x}^*$ for P_r is an integer solution then $\mathbf{x}^*$ is the optimal solution to P and we stop.

Step 3 If $\mathbf{x}^*$ is not integer valued, then choose a component x_i^* of $\mathbf{x}^*$ with the largest fractional part. Let the x_i-equation in the final feasible dictionary of P_r be

$$x_i = x_i^* - \sum_{x_j \in N} a_j x_j.$$

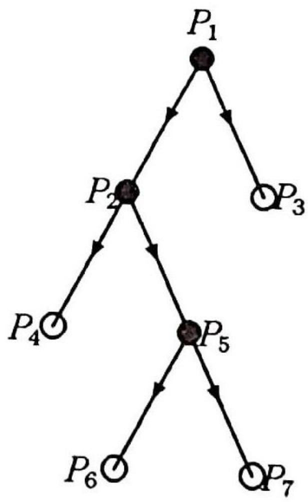
Form a new linear programming problem P_{r+1} by adding the following constraint to P_r:

$$-\{x_i^*\} + \sum_{x_j \in N} \{a_j\} x_j \geq 0,$$

where $\{x\}$ denotes the fractional part of x.
Solve P_{r+1} using the Dual Simplex Algorithm.
Return to Step 2 with $r := r + 1$.

5.3 The Branch and Bound Algorithm

We saw in the last section that Gomory's cutting plane algorithm searches for the optimal solution to the integer programming problem P by proceeding along a path of linear programming problems $P_1, P_2, \ldots, P_m$ with the properties that each integer feasible solution to P is a feasible solution to P_i for all $1 \leq i \leq m$. The algorithm terminates when we find that one of the P_i has an integer valued optimal solution. In contrast, the branch and bound algorithm searches for the optimal solution to P by proceeding along a binary tree of linear programming problems with the property that each integer feasible solution to P is a feasible solution to exactly one of linear programming problems which are currently end vertices of the tree. The algorithm terminates when all the branches of the tree have been 'terminated', as illustrated by the following figure (in which terminated branches are indicated by open circles).



The constuction of a branch and bound tree for solving P can be summarised as follows. We begin with the linear relaxation P_1 of P as the root vertex. As we draw the tree we continually update a *lower bound* M on the optimal value for z in P. (At the start of the algorithm we set $M = -\infty$. The value of M is updated as we find better integer feasible solutions for P.) We terminate a branch if the linear programming problem corresponding to its end-vertex either has an integer solution, or is infeasible, or if it has optimal value strictly less than $M+1$. (The latter alternative implies that the optimal solution to P is not a feasible solution to any problem further down this branch.)

Suppose we have solved a problem P_i corresponding to an end vertex of the tree, and found the optimal solution $\mathbf{x} = \mathbf{x}^*$ and optimal value $z^* = z(\mathbf{x}^*)$. We extend the tree from P_i as follows:

- If $\mathbf{x}^* \in \mathbf{Z}^n$ then we terminate this branch at P_i and set $M = \max\{M, z^*\}$.
- If $\mathbf{x}^*$ has at least one co-ordinate which is not integer valued and $z^* < M+1$, or if P_i is infeasible, then we terminate this branch at P_i since it cannot contain the optimal integer solution.
- If $\mathbf{x}^*$ has at least one co-ordinate which is not integer valued and $z^* \geq M+1$ then we must explore this branch. We choose a basic variable x_j such that the value of x_j in the optimal solution, say $x_j = x_j^*$, is not an integer. (We call x_j the *branching variable.*) We construct two new problems P_i' and P_i'' from P_i by adding each of the two constraints $x_j \leq \lfloor x_j^* \rfloor$ and $x_j \geq \lfloor x_j^* \rfloor + 1$, respectively, to P_i. Each integer solution to P_i will be an integer solution to either P_i' or P_i''. However the old optimal solution $\mathbf{x} = \mathbf{x}^*$ of P_i is not a feasible solution of P_i' or P_i''.

We continue this process until all branches have been terminated.

5.3.1 Solution to Example 5.1.1

The optimal solution to P_1 is given by the feasible dictionary (9): $x_1 = \frac{16}{3}$ and $x_2 = \frac{5}{3}$. Since both x_1 and x_2 take non-integer values in this optimal solution, we may choose either x_1 or x_2 as the first branching variable. We choose x_1 arbitrarily.
We form P_2 and P_3 from P_1 by adding the constraints $x_1 \leq \lfloor 16/3 \rfloor = 5$ to give P_2 and $x_1 \geq \lfloor 16/3 \rfloor + 1 = 6$ to give P_3.

- Solution to P_2. We introduce a new slack variable x_5 satisfying $x_1 + x_5 = 5$. Then

$$x_5 = 5 - x_1 = 5 - (\tfrac{16}{3} - \tfrac{2}{3}x_3 - \frac{1}{3}x_4) = -\tfrac{1}{3} + \tfrac{2}{3}x_3 + \tfrac{1}{3}x_4.$$

We add this equation to (9) Then using the Dual Simplex Algorithm, we choose x_5 as leaving variable, x_4 as entering variable. The next feasible dictionary is

$$\begin{array}{rcrcrcr} x_4 & = & 1 & - & 2x_3 & + & 3x_5 \\ x_1 & = & 5 & & & + & x_5 \\ x_2 & = & 2 & - & x_3 & + & x_5 \\ z & = & 12 & - & x_3 & - & x_5 \end{array}$$

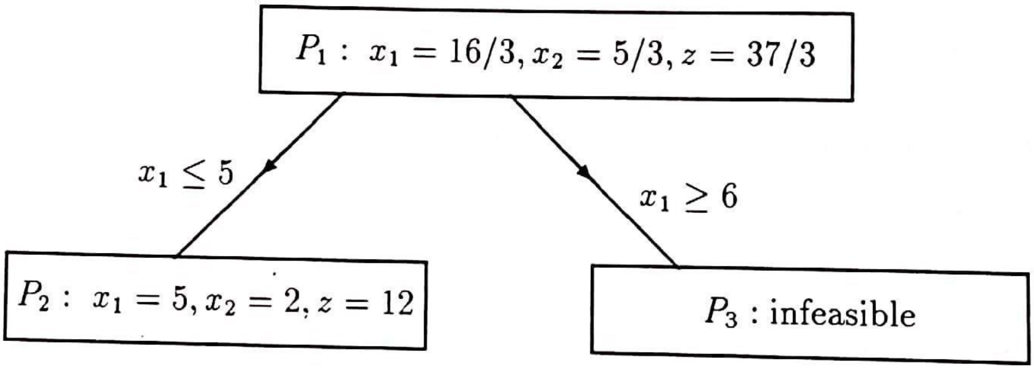
Thus the optimal solution is $x_1 = 5$, $x_2 = 2$ and $z = 12$. This is integer valued so we terminate the P_2 branch and set $M := 12$.

- Solution to P_3. We introduce a new slack variable x_5 satisfying $x_1 - x_5 = 6$. Thus

$$x_5 = -6 + x_1 = -6 + (\tfrac{16}{3} - \tfrac{2}{3}x_3 - \tfrac{1}{3}x_4) = -\tfrac{2}{3} - \tfrac{2}{3}x_3 - \tfrac{1}{3}x_4.$$

Adding this equation to (9) we have no choice for entering variable so P_3 is infeasible and we terminate the P_3 branch.

Now all branches have been terminated. So the optimal solution to P is $x_1 = 5$, $x_2 = 2$, and $z = 12$. The solution can be illustrated by the following branch and bound tree.



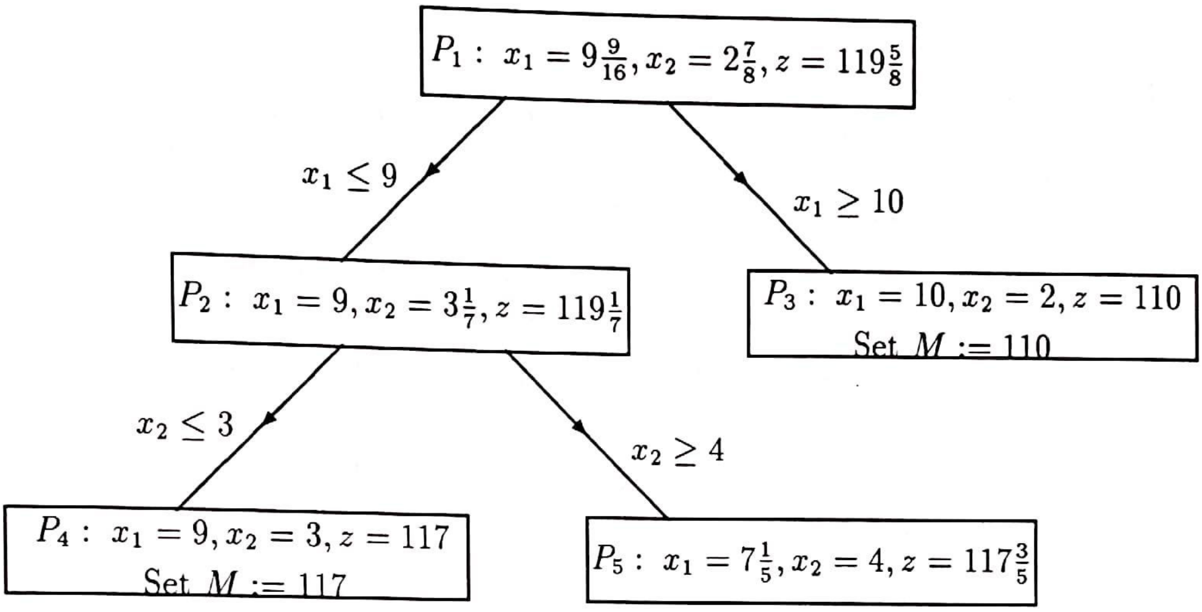
Exercise Give a graphical interpretation for the branch and bound solution given above.

5.3.2 Example

The example given in subsection 5.3.1 is perhaps too simple to give a full appreciation of the branch and bound algorithm. A better example is

P: Find $x_1, x_2 \in \mathbf{Z}$ to maximise $z = 8x_1 + 15x_2$ subject to $10x_1 + 21x_2 \leq 156$, $2x_1 + x_2 \leq 22$, and $x_1, x_2 \geq 0$.

The optimal solution to the linear relaxation P_1 of P is $x_1 = 9\frac{9}{16}, x_2 = 2\frac{7}{8}$, and $z = 119\frac{5}{8}$. Choosing x_1 as the first branching variable we obtain the following branch and bound tree.



Although P_5 has optional solution $117\frac{3}{5}$ which is larger than the current value of $M = 117$, we can use the fact that z has integer coefficients to deduce that all integer solutions x^* to P_5 will satisfy $z(x^*) \leq 117\frac{3}{5} = 117$. Thus, if we only wish to find one optimal solution to P we may terminate the P_5-branch and deduce that $x_1 = 9, x_2 = 3$ and $z = 117$ is an optimal solution to P. Note however that there may be other optimal solutions in the P_5-branch. If we want to find all the optimal solutions then we will need to explore the P_5-branch further.

5.4 The Knapsack Problem

This is an integer programming problem of the following simple form:

Find $\mathbf{x} \in \mathbf{Z^n}$ to maximize $z = \mathbf{c^T x}$ subject to $\mathbf{a^T x} \leq b$ and $\mathbf{x} \geq \mathbf{0}$,
where $\mathbf{c}, \mathbf{a} \in \mathbf{Z^n}$, $\mathbf{c}, \mathbf{a} \geq \mathbf{0}$ are given vectors, and $b \geq 0$ is a given integer.

5.4.1 Example

A thief breaks into a warehouse containing TV's, CD's, radios and videos. These have values and weights given by the following table:

	TV	CD	RADIO	VIDEO
value ($)	180	160	80	130
weight (Kgm)	10	8	5	7

The thief can carry at most 92 kgm of items. What should he take in order to maximise the total value? Letting x_1, x_2, x_3, x_4 be the numbers of TV's, CD's, radios and videos which he takes we obtain the knapsack problem:

Find $x_1, x_2, x_3, x_4 \in \mathbf{Z}$ to maximize $z = 18x_1 + 16x_2 + 8x_3 + 13x_4$ subject to $10x_1 + 8x_2 + 5x_3 + 7x_4 \leq 92$ and $x_i \geq 0$ for $1 \leq i \leq 4$.

We shall solve the problem using a variation of the branch and bound algorithm. We first compute the *efficiency* of each variable x_i. This is defined to be c_i/a_i (where $\mathbf{c} = (c_1, c_2, c_3, c_4)^T$ and $\mathbf{a} = (a_1, a_2, a_3, a_4)^T$). The efficiency of x_i represents the value of one kilo of the item corresponding to x_i.

variable	x_1	x_2	x_3	x_4
efficiency	1.8	2	1.6	1.86

We next construct a branch and bound tree, choosing our branching variables in the order of decreasing efficiency i.e. x_2, then x_4, then x_1, then x_3.

We construct the tree given in Figure 4 as follows. We first make x_2 as large as possible by setting $x_2 = \lfloor 92/8 \rfloor = 11$. This leaves a 'slack' of only $92 - 8(11) = 4$ in the constraint so forces $x_4 = x_1 = x_3 = 0$ in all integer solutions on this branch and gives a z-value of 176. This gives us our first integer feasible solution and we set $M := 176$.

We next consider the branch $x_2 = 10$, which leaves a slack of $92 - 8(10) = 12$ in the constraint. Since the efficiency of the next most efficient variable x_4 is 1.86 it follows that any feasible solution on the branch $x_2 = 10$ has a z-value of at most $16(10) + 1.86(12) \approx 182$. Since $M + 1 = 177 \leq 182$ it is possible that the branch contains an integer solution giving a z-value greater than 176. Thus we need to explore this branch. We set $x_4 = \lfloor 12/7 \rfloor = 1$, leaving a slack of $12 - 7 = 5$ in the constraint. Clearly the best integer solution down this branch is $x_1 = 0$ and $x_3 = 1$, which gives a z-value of 181. This is better than the previous solution so we set $M := 181$.

We next backtrack along the current branch until we find the first variable which has been given a non-zero value, i.e. $x_4 = 1$. We then consider the branch we obtain by decreasing the value of this variable by one, i.e. $x_2 := 10$ and $x_4 = 1 - 1 = 0$. Since the efficiency of the next most efficient variable x_1 is 1.8 it follows that any feasible solution on this branch has a z-value at most $18(10) + 1.8(12) = 181.6 < M + 1$. Since we already have a solution giving a z-value of 181, this branch may be terminated.

Next we backtrack along the current branch $x_2 = 10, x_4 = 0$ to the first non-zero variable i.e. $x_2 = 10$, and consider the branch $x_2 = 9$. Since the efficiency of x_4 is 1.86 we may deduce that all feasible solutions on this branch have a z-value of at most $9(16)+1.86(20) = 181.2 < M+1$. Thus the branch $x_2 = 9$ can be terminated. Similarly we terminate the branches $x_2 = 8, 7, ..., 0$ and deduce that the optimal solution is $x_2 = 10, x_4 = 1, x_1 = 0, x_3 = 1$ and $z = 181$. The branch and bound tree for this solution is shown below.

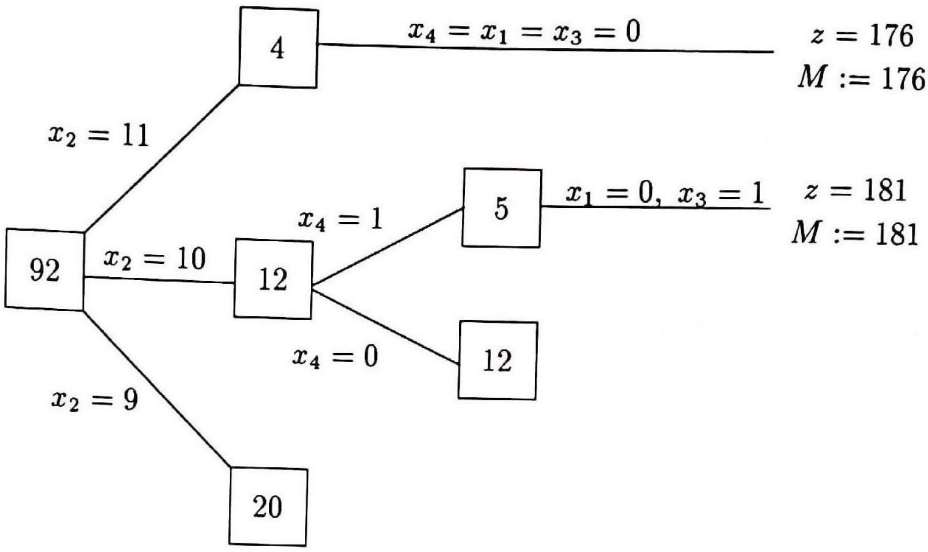


Figure 4:A branch and bound tree for the knapsack problem in Example 5.4.1.