**Joe Wilck, Integer Programming:** *Formulation*Basic Types of Integer Programs (IPs):  
1. Direct - there is a physical meaning to the variables (e.g., # of workers).  
2. Coded - yes/no decisions  
3. Transformed - *Either/Or* and *Choose k from m* are two examples.

Important coded IPs: knapsack, fixed charge, and set covering; Traveling Salesman Problem.

**1.** knapsack: given a fixed amount of space, select items to max utility (return)

Index sets: i = items (i = 1, .., n)

Data:  
ci = utility of item i  
ai = space consumed by item i  
b = space available

Variables: xi = 1 if item i is selected; 0 otherwise.  
  
Objective: Maximize Utility  
Max   
Constraints:  
 (Space constraint.)  
xi = 0 or 1 (xi binary)

**2.** fixed charge problem (i.e., potential facility sites); setup costs and fixed costs associated with customers and facilities. Minimize total cost of opening facilities and meeting customer demand.

Index Sets:  
i = facility number (i = 1, …, m)  
j = customer number (j = 1, …, n)

Data:  
cij = unit shipping cost from facility i to customer j  
dj = demand for customer j  
ki = capacity of facility i  
fi = fixed cost (charge) of opening (or constructing) facility i

Variables:  
xi = 1 if facility i is open; 0 otherwise  
yij = amount shipped from facility i to customer j

Objective Minimize Cost  
Min 

Constraints:  
 (Facility capacity.)  
 (Demand. Could be ≥ since minimum cost.)  
xi is binary and yij ≥0 for all i,j [yij is integer, but will solve as integer due to problem constraints/objective.]

**3.** set covering problem - find the minimum cost subset of facilities so that each region is covered by at least 1 facility (i.e., fire stations)

Indexed Sets:  
i = region (i = 1, …, m)  
j = facility (j = 1, …, n)

Data:  
cj = cost of building/opening facility j  
aij = 1 if region i can be covered by facility j; 0 otherwise.

Variables:  
xj = 1 if facility j is opened; 0 otherwise.

Objective: Minimize Cost  
Min 

Constraints:  
  
xj ~ binary

Closely related to Set Partitioning. Where each region is covered by EXACTLY one facility. So constraint is =.

**4.** TSP: Traveling Salesman Problem - given n cities, find the least cost (distance) Hamiltonian circuit. Hamiltonian circuit travels through each city exactly once.

Index sets:  
i: city number (i = 1, …, n)  
j: city number (j = 1, …, n)

Data:  
cij: distance between cities i and j

Variables:  
xij = 1 if we travel directly from i to j; 0 otherwise. ti ~subtour variables

Objective Minimize Cost:  
Min   
Constraints:  
 (Leave a city exactly once.)  
 (Enter a city exactly once.)  
ti - tj + n xij ≤ n-1 (for i≠j; i=2, …, n; j=2, …, n)  
(Subtour Elimination Constraints: eliminates all subtours that do not include Node 1)  
xij ~ binary ti ≥ 0 for all i

The subtour elimination constraints above are sufficient since i cannot equal j. However, there are better (allow for easier solving time) Subtour Elimination Constraints.

The first two constraint sets are the assignment problem.

**"*At Least k Out of m*" And "*Either - Or Constraints*" Examples:**  
*Example*, consider the following three constraints: 2x1 + x2 ≤ 12 x1 + x2 ≤ 7 x1 +2x2 ≤ 12  
What does the feasible region look like if all must be satisfied? Only 1 must be satisfied? Only 2 must be satisfied?

**Solution:**

Can draw pictures.

All feasible. Convex set.  
One or two feasible. Non-convex

Basic strategy: relax the constraints, and then force a subset (of the constraints) to be feasible.

2x1 + x2 ≤ 12 + M(1-y1)  
x1 + x2 ≤ 7+ M(1-y2)  
x1 +2x2 ≤ 12+ M(1-y3)  
yi = 1 if constraint i is enforced; 0 otherwise.

At least 1 is satisfied:  and yi ~ binary.  
At least 2 are satisfied:  and yi ~ binary.

*Example*, suppose we are scheduling jobs on a single machine.  
Let ti equal machine time required for job i.  
Let xi equal start time for job i.  
Thus, completion time of job i is equal to ti + xi.  
Consider two jobs, j and k, that both need to use the machine.

**Solution:**Completion time is less-than or equal to start time of next job.  
xj + tj ≤ xk OR xk + tk ≤ xj   
xj + tj ≤ xk + My When y=1 ~ this constraint is relaxed.  
xk + tk ≤ xj + M(1-y) When y=0 ~ this constraint is relaxed.  
y is binary.  
Choose "M" large enough to properly relax the constraint, but not too large to cause solution problems.

**Formulation Examples:**

**1.** Coach Fischer is picking his starting lineup for the W&M basketball team. The team consists of seven players who have been rated (on a scale of 1 poor to 3 = excellent) according to their ball-handling, shooting, rebounding, and defensive abilities. The positions that each player is allowed to play and the player's abilities are listed in the table.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Player | Position | Ball-Handling | Shooting | Rebounding | Defense |
| 1 | G | 3 | 3 | 1 | 3 |
| 2 | C | 2 | 1 | 3 | 2 |
| 3 | G, F | 2 | 3 | 2 | 2 |
| 4 | F, C | 1 | 3 | 3 | 1 |
| 5 | G, F | 3 | 3 | 3 | 3 |
| 6 | F, C | 3 | 1 | 2 | 3 |
| 7 | G, F | 3 | 2 | 2 | 1 |

The five-player starting lineup must satisfy the following restrictions:

1. At least 4 members must be able to play guard (G), at least 2 members must be able to play forward (F), and at least 1 member must be able to play center (C).

2. The average ball-handling, shooting, and rebounding level of the starting lineup must be at least 2.

3. If player 3 starts, then player 6 cannot start.

4. If player 1 starts, then players 4 and 5 must start.

5. Either player 2 or player 3 must start.

Given these constraints, Coach Shaver wants to maximize the total defensive ability of the starting team. Formulate an integer program (IP) that will help him choose his starting lineup. DO NOT SOLVE!

**Nonlinear Examples:**

**2.** A closed cylindrical tank is being designed to carry at least 20 cubic feet of chemicals. Metal for the top and sides costs $2 per square foot, but the heavier metal of the base costs $8 per square foot. Also, the height of the tank can be no more than twice its diameter to keep it from being top heavy. Formulate a constrained nonlinear program to find a design of minimum cost.

**3.** Two airstrips are to be constructed in the jungle to service 6 remote oil fields. The first oil field requires 25 tons of supplies per month. Oil Field 2, which is 75 miles east and 330 miles north of the first, requires 14 tons. Oil Field 3 requires 30 tons per month and is 100 miles east and 280 miles south of the first. Oil Field 4 requires 18 tons per month and is 150 miles west and 160 miles north of the first. Oil Field 5 requires 22 tons per month and is 200 miles east and 100 miles north of the first. Oil Field 6 requires 34 tons per month and is 225 miles west and 40 miles south of the first. Formulate a model to locate the airstrips and minimize the ton-miles flown per month.

***Additional Example (Supplemental):***Governor Blue of the state of Berry is attempting to get the state legislature to gerrymander Berry's congressional districts. The state consists of ten cities, and the numbers of registered Republicans and Democrats (in thousands) in each city are shown in the table below. Berry has five congressional representatives. To form congressional districts, cities must be grouped according to the following restrictions:  
i) All voters in a city must be in the same district.  
ii) Each district must contain between 150,000 and 250,000 voters (there are no independent voters).  
Governor Blue is a Democrat. Assume that each voter always votes a straight party ticket. Formulate an IP to help Governor Blue maximize the number of Democrats who will win congressional seats. Assume districts with an equal amount of Republicans and Democrats will vote Republican (i.e., ties always go to Republicans).

|  |  |  |
| --- | --- | --- |
| City | Republications | Democrats |
| City 1 | 80,000 | 34,000 |
| City 2 | 60,000 | 44,000 |
| City 3 | 40,000 | 44,000 |
| City 4 | 20,000 | 24,000 |
| City 5 | 40,000 | 114,000 |
| City 6 | 40,000 | 64,000 |
| City 7 | 70,000 | 14,000 |
| City 8 | 50,000 | 44,000 |
| City 9 | 70,000 | 54,000 |
| City 10 | 70,000 | 64,000 |

Index Sets:  
i: city #  
j: district #

Data:  
ri: # of republications in city i  
di: # of democrats in city i  
l: lower population limit  
u: upper population limit  
M: large arbitrary number

Variables:  
xij: 1 if city I is assigned to district j; 0 otherwise  
yj: 1 if # of democrats > # of republications in district j; 0 otherwise

First, when does yj = 1 or 0.



Use either/or constraint to help with y.

Objective:  
Maximize Z =  (Maximize number of districts with democratic control)

Constraints:  
 (Each city must be assigned to one district)  
 (Each district must be within population limits)  
 (First 'on' if Dem > Rep; Second 'on' if Rep > Dem).

xij ~ binary all i,j  
yj ~ binary all j  
M = u would work.