NON LINEAR PROGRAMMING
Prof. Stephen Graves

In a linear program, the constraints are linear in the decision variables, and so is the objective function. In a **non linear program**, the constraints and/or the objective function can also be non linear function of the decision variables.

Example: Gasoline Blending: The qualities of a blend are determined by the qualities of the stocks used in the blend. An optimization is to determine the volume of each input stock in each blend so that the objective function is optimized subject to the output blends satisfying their quality specifications, stock availability constraints, and blend demand constraints.

The decision variables are x_{ij} denoting the amount of stock i in blend j.

For the most part the constraints can be written as linear functions; but some of the quality constraints are non linear:

Distillation Blending:

$$D_{jk} = b_k + c_k * ln [\Sigma_i (S_{ik} * VF_{ij})]$$

where

D_{jk} is the k th distillation point for blend j.

S_{ik} is the k th distillation point for stock i.

VF_{ij} is the volume fraction of stock i in blend j and is equal to $x_{ij}/ (\Sigma_i x_{ij})$

and b_k and c_k are constants.

Octane Blending:

$$OCT_{jk} = a_k\{ \Sigma_i \ (b_i \ * b_i * VF_{ij}) - \ \Sigma_i \ (c_i * VF_{ij})^2 \}$$

$$+ \ d_k\{ \ \Sigma_i \ (e_i * VF_{ij}) \ - \ \Sigma_i \ (f_i * VF_{ij})^2 \}^2$$

$$+ \ g_k \ \Sigma_i \ \{ (h_i * VF_{ij}) - \ (j_i \ * k_i * VF_{ij} * VF_{ij})\}$$

where OCT_{jk} are the various octane indices for blend j.

For both D_{jk} and OCT_{jk} the optimization problem would have simple upper bounds and lower bounds for each blend and for each quality index.

Thus the constraints for the formulation would include:

for each stock i: $\sum_j x_{ij} \leq A_i$ where A_i is the availability of stock i

for each blend j: $\sum_i \mathrm{x}_{ij} \geq R_j$ where R_j is the requirement for blend j

for each combination i, j, we define : $\mathrm{V}_{ij} = \frac{x_{ij}}{\sum_i x_{ij}}$

plus upper bounds and lower bounds on the distillation points and octane levels for each blend

Example: Site Location; given customer locations (x_i, y_i), find the location (X, Y) that minimizes the weighted distances from the customer to the central warehouse (or minimizes the maximum distance to an emergency vehicle location).

The distance from customer i to the warehouse is d_i and is typically a non-linear function of the decision variables (x_i, y_i), and (X, Y). To wit, we might have

$$d_i = \sqrt{(x_i - X)^2 + (y_i - Y)^2}$$

or

$$d_i = |x_i - X| + |y_i - Y|$$

We then have an objective: $Min \sum_{i=1}^{N} w_i d_i$ subject to constraints on the decision variables.

Example: Determine the production quantities for each family of car (luxury, intermediate, mid size, compact, subcompact) that maximizes net revenue subject to production capacity constraints, fleet fuel mileage constraints. (Haas, SM thesis, 1977)

Decision variables are q_i and p_i, which denote the quantity and price for each car family.

We then need to assume a relationship between price and quantity, e.g., linear supply-demand function: $q_i = a_i - b_i p_i$ where a_i and b_i are positive constants.

The objective of the model is non-linear, to maximize profits:
$Max \sum_i q_i \times (p_i - C_i) = \sum_i q_i \times (p_i - C_i)$ where C_i equals the cost per unit for car from family i. We would have linear capacity constraints:

for each resource type j: $\sum_i R_{ij} q_i \le K_j$ where K_j is the amount of available respource of type j, and R_{ij} is the per unit consumption of resource j to produce a unit of car i.

We also have a non-linear fleet fuel mileage constraint; the fleet fuel mileage is computed as the harmonic average, and needs to exceed some target, say 30 mpg:
$\dfrac{q_1 + q_2 + \ldots + q_n}{q_1/mpg_1 + q_2/mpg_2 + \ldots + q_n/mpg_n} \ge 30 mpg$ where mpg_i is the miles per gallon for car family i.

Example: Flow in Pipes - In designing a network of pipes, say, for a chemical processing facility, you might be given the network topology (nodes and edges), the desired flow inputs at supply points, desired flow outputs at consumption points, and inlet pressures at supply points. The decision variables are the size of pipes (diameter) needed to connect the nodes of the network.

The problem is to determine for each edge of the network, the diameter of the pipe and the flow rate on that edge. We define the variables:

q_j is the flow rate on edge j,
dp_j is the pressure drop across edge j, and
d_j is the diameter of edge j.

There are flow balance constraints at each node of the network (i.e., flow into the node = flow out of the node), and constraints on the external flow inputs and outputs.

For each edge, the flow rate is a non-linear function of the diameter of the pipe and the drop in pressure across the edge, e.g., :

$$q_j^2 = c \ dp_j \ d_j^5$$

where c is a constant, and the drop in pressure across an edge equals the difference in the node potentials. The objective would be to minimize the cost of the pipes, which depends on the diameters chosen.

Example: Robot Motion Planning (taken from LFM thesis, *Evaluation of a New Robotic Assembly Workcell Using Statistical Experimental Techniques and Scheduling Procedures* by Erol Erturk, 1991)

The problem is to determine the velocity and acceleration for a new robot assembly system for a given displacement of length d. The objective is to minimize time, subject to a constraint on placement accuracy. We assume the robot accelerates at a constant rate of acceleration until it reaches its peak velocity, then will travel at its peak velocity, until it must decelerate also at a linear rate. Then the time (in seconds) to travel a distance d is:

Travel time = T = v/a + d/v

where a is the acceleration and deceleration rate (inch/sec^2), and v is the peak velocity (inch/sec) .

The accuracy (in mils) of the placement depends upon the acceleration rate and the peak velocity and has been found empirically to be given by:

$$A = accuracy = \left| 0.022v \ + \ 0.0079a \ - \ 0.0002v \times a \right|$$

The optimization is then to minimize T, subject to a constraint on accuracy A, as well as upper bounds on acceleration (a) and velocity (v).

Example: Design parameters for coil spring (from Rajan Ramaswamy's thesis, *Computer Tools for Preliminary Parametric Design*, Ph. D., LFM 1993) The coil spring is used to provide a clamping force in an indexing mechanism. Hence, it must deliver a specified force while satisfying constraints on compressed length, geometry and material. The objective is to find the lightest feasible design, i. e., minimize mass. The following equations come from Mark's handbook for mechanical engineers:

(1) C = Dspring/Dwire

C is spring coefficient; Dspring is spring diameter (in); and Dwire is the wire diameter (in).

(2) Kw = (4*C -1)/(4*C- 4) + 0.615/C

Kw is the Wahl curvature correction factor.

(3) Kspring = (Dwire^4 * G)/(8 * Dspring^3 * Ncoils)

Kspring is the spring stiffness (lb/in); G is torsional modulus (Mpsi); and Ncoils is the number of coils.

(4) L_Free = (Ncoils * Dwire +Tau_Max * 3.14*Dwire^3)/(8 * Dspring * Kw * Kspring)

L_Free is free length (in); Tau_Max is the peak allowable shear stress (Mpsi).

(5) F_Act = Kspring * (L_Free - L_C)

F_Act is the force acting (lb); L_C is the compressed length (in).

(6) Tau = (Kw * 8 * F_Act * Dspring) / (3.14 * Dwire^3)

Tau is the shear stress (Mpsi).

(7) Mass = Rho * Ncoils * 3.14 * 3.14 * Dwire^2 * Dspring/4

Mass is the mass of the spring (oz); and Rho is the density of the spring material (lb/in^3).

For this problem, L_C, G, Rho, Tau_Max are given constants.

We are given a lower bound on F_Act, namely the desired force; an upper bound on Tau, namely Tau-Max, a lower bound on L_Free, namely L_C, and upper and lower bounds on C, Dspring and Dwire. The number of coils (Ncoils) has to be at least 3.

ISSUES WITH NON LINEAR PROGRAMS

- Optimal solution may occur at extreme point; may occur on the boundary of feasible region; may occur in the interior of feasible region.

- Many problems will have both a global optimum and several local optima; it is very hard to distinguish between local and global optima.

APPROACHES TO NON LINEAR PROGRAMS

- Approximate non linear functions with linear or piece-wise linear functions, possibly by using binary integer variables. This works well if the non linear functions "separate" by decision variable.

- Use monotonic or algebraic transformation (e. g., log's) to make non linear functions linear.

- Solve for necessary conditions given by Lagrangean Function (if constraints are equalities) or given by Kuhn Tucker conditions (if constraints are inequalities). This works well if there are very few constraints or if the objective is quadratic and all of the constraints are linear (<u>a quadratic program</u>).

- Use a search algorithm:

  Find a feasible point (vector) to start **X**

  Find a feasible direction of improvement **D**

  Move to new point along direction of improvement, step size τ:
  $\mathbf{X} := \mathbf{X} + \tau \mathbf{D}$

  Continue until some convergence criterion satisfied.