Linear Programming (LP): Formulating Models for LP

Benoît Chachuat <benoit@mcmaster.ca>

McMaster University
Department of Chemical Engineering

ChE 4G03: Optimization in Chemical Engineering

Outline

1. "Straightforward" Models for LP
2. Approximations and Reformulations as LP Models

"Straightforward" LP Models Formulation

Everything in life is **not linear and continuous**! But an enormous variety of applications can be modeled validly as LPs:

- Allocation Models
- Blending Models
- Operations Planning
- Operations Scheduling

For additional examples, see Rardin (1998), Chapter 4

Allocation Models

The main issue in allocation models is to divide or allocate a valuable resource among competing needs.

- The resource may be land, capital, time, fuel, or anything else of limited availability
- Principal decision variables in allocation models specify how much of the critical resource is allocated to each use

Example:

The Ontario Forest Service must trade-off timber, grazing, recreational, environmental, regional preservation and other demands on forestland.

The optimization seeks the best possible allocation of land to particular prescriptions (e.g., in terms of the *net present value*), subject to forest-wide restrictions on land use.

$x_{i,j} \triangleq$ number of acres in area i managed by prescription j

Blending Models

The main issue in blending models is to decide what mix of ingredients best fulfills specified output requirements.

- The blend can be from chemicals, diets, metals, animal foods, etc.
- Principal decision variables in blending models specify how much of the available ingredients to include in the mix
- Composition constraints typically enforce lower and/or upper limits on the properties of the blend

Example: Gasoline Blending Process

Maximize: Profit

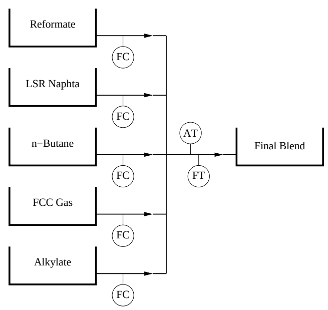
Subject to: Product Flow = •

• ≤ Octane No. ≤ •

• ≤ RVP ≤ •

• ≤ Rel. Vol. ≤ •

• ≤ Component Flows ≤ •

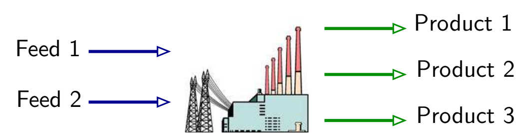


Operations Planning Models

The main issue in operations planning models is to help a decision maker decide what to do and where to do it.

- The decision making can be in manufacturing, distribution, government, volunteer, etc.
- Principal decision variables in operations planning always resolve around what operations to undertake — recall that, to gain tractability, decision variables of relatively large magnitude are best modeled as continuous
- Balance constraints assure that in-flows equal or exceed out-flows for materials and products created by one stage of production and consumed by others

Example: Which amounts of Feed 1 and Feed 2 should we use?



Operations Planning Models

Class Exercise: What is the Optimization Model?

	Prod. 1	Prod. 2	Prod. 3	Feed Min.	Feed Max.	Feed Cost
Feed 1	0.7	0.2	0.1	0	1000	5
Feed 2	0.2	0.2	0.6	0	1000	6
Prod. Min.	0	0	0			
Prod. Max.	100	70	90			
Prod. Value	10	11	12			

$$\begin{aligned}
\max\quad & [10P_1 + 11P_2 + 12P_3] - [5F_1 + 6F_2] \\
\text{s.t.}\quad & P_1 = 0.7F_1 + 0.2F_2 \\
& P_2 = 0.2F_1 + 0.2F_2 \\
& P_2 = 0.1F_1 + 0.6F_2 \\
& 0 \leq P_1 \leq 100 \\
& 0 \leq P_2 \leq 70 \\
& 0 \leq P_3 \leq 90 \\
& 0 \leq F_1, F_2 \leq 1000
\end{aligned}$$

Are there important issues not included?

-
-
-

Operations Scheduling Models

In operations scheduling models, the work is already fixed and the resources must be planned for meeting varying-time demands.

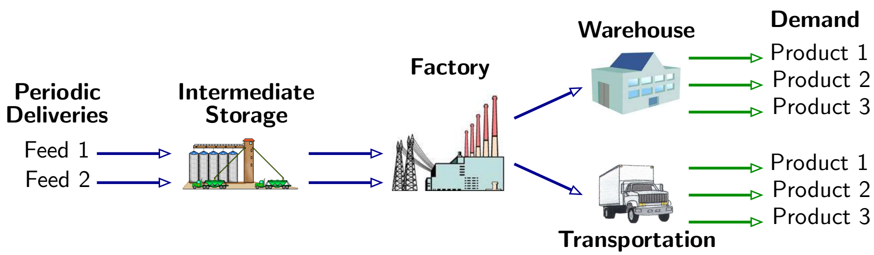
- Principal decision variables in operations scheduling are time-phased — time is an index and decisions may be repeated in each time period
  - Number of ball-bearings produced during period t
  - Inventory level at the beginning of period t
  - Number of employees beginning a shift at time t
- Scheduling models typically link decisions in successive time periods with balance constraints of the form:

$$\begin{pmatrix}\text{starting}\\ \text{level in}\\ \text{period } t\end{pmatrix} + \begin{pmatrix}\text{impacts of}\\ \text{period } t\\ \text{decisions}\end{pmatrix} = \begin{pmatrix}\text{starting}\\ \text{level in}\\ \text{period } t+1\end{pmatrix}$$

- Covering constraints assure that the requirements over each time periods are met

Operations Scheduling Models

Example: Raw material deliveries are now periodic: We must decide when and how much raw materials to purchase in order to maximize profit while satisfying the demand



- The model must consider the inventories in tanks, factory, warehouses
- Delays in transportation can be important
- What type of balances are needed?

"Straightforward" Models for LP

Fundamental Balances:

- **Material:** ball bearings, fluid, people, etc.
- **Energy:** vehicle travel, processing, etc.
- **Space:** volume, area
- **Lumped quantity:** pollution, economic activity, etc.
- **Time:** utilization of equipment, people work, etc.

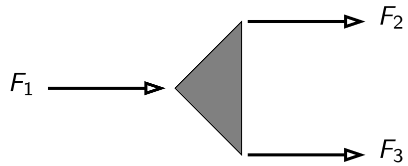
Only retain **key decisions** as variables:

- Production rates
- Flows
- Investment
- Inventories

Combine other factors in **parameters** (constants)

Formulating "Straightforward" Models for LP

Example: Flow Splitting



Feed and product mole fractions $\alpha_1, \ldots, \alpha_n$

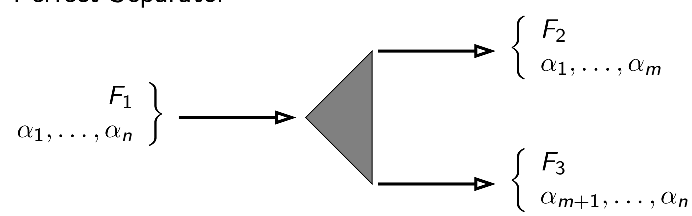
- **Material Balance:**

$$F_1 = F_2 + F_3$$

- Feed and product composition (mole fractions) cannot change — **Why?**

Formulating "Straightforward" Models for LP

Example: Perfect Separator



- **Material Balance:**

$$F_1 = F_2 + F_3$$

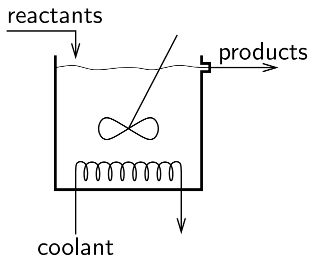
- **Component Balance:**

$$F_1 \sum_{k=1}^{m} \alpha_m = F_2$$

  - Can we make the product mole fractions $\alpha_1, \ldots, \alpha_n$ variables?
  - Could we model it differently?

Formulating "Straightforward" Models for LP

Example: CSTR Reactor



- Reaction system: $\begin{cases} \mathrm{A} + \mathrm{B} \rightarrow \mathrm{C} \\ \mathrm{B} + \mathrm{C} \rightarrow \mathrm{D} \end{cases}$
- Feed flow rate: F_{f}
- Product flow rates: $F_{\mathrm{A}}, F_{\mathrm{B}}, F_{\mathrm{C}}, F_{\mathrm{D}}$

Material Balances:

$$F_{\mathrm{A}} = \alpha_{\mathrm{A}} F_{\mathrm{f}}$$
$$F_{\mathrm{B}} = \alpha_{\mathrm{B}} F_{\mathrm{f}}$$
$$F_{\mathrm{C}} = \alpha_{\mathrm{C}} F_{\mathrm{f}}$$
$$F_{\mathrm{D}} = \alpha_{\mathrm{D}} F_{\mathrm{f}}$$

Remarks:

- The α's are for specific reactions, reactor temperature, level, mixing pattern, etc.
- If A, B, C, D are the only components,

$$\alpha_{\mathrm{A}} + \alpha_{\mathrm{B}} + \alpha_{\mathrm{C}} + \alpha_{\mathrm{D}} = 1$$

- The F's are mass units!

Approximate Models for LP

Besides "straightforward" LP models, certain classes of nonlinear or multiobjective optimization problems can be **reformulated** or **approximated** as LP models:

- Base-Delta Models
- Separable Programming
- Minimax and Maximin (Linear) Objectives
- Goal Programming

These approaches are usually reasonable when the uncertainties in the problem do not justify further model accuracy — Otherwise, solve the nonlinear model using NLP!

Base-Delta LP Models

- **Goal:** Extend "straightforward" LP models to include nonlinear, secondary decision variables:

$$\mathbf{0} = \mathbf{f}(\mathbf{x},\mathbf{y}) \approx \mathbf{f}(\mathbf{x},\mathbf{y}^\circ) + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{y}}\right|_{\mathbf{x}^\circ,\mathbf{y}^\circ} (\mathbf{y}-\mathbf{y}^\circ)^{\mathsf{T}}$$

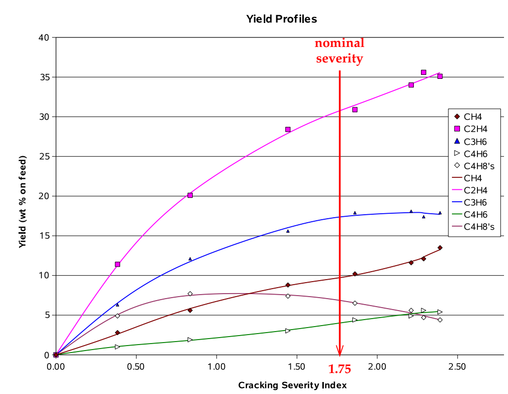
  - The base model, $\mathbf{f}(\mathbf{x},\mathbf{y}^\circ)$, describes the (linear) effect of the primary decision variables, while the secondary variables are kept constant at their nominal value $\mathbf{y} = \mathbf{y}^\circ$
  - The delta model provides small corrections for deviations ("deltas") in the secondary variables $\mathbf{y}$ around $(\mathbf{x}^\circ, \mathbf{y}^\circ)$

- The accuracy of the solution depends on how well the approximation applies at $(\mathbf{x}^*, \mathbf{y}^*) \neq (\mathbf{x}^\circ, \mathbf{y}^\circ)$

- To improve the accuracy, the primary and secondary decision variables should be limited by upper and lower bounds

Formulating Base-Delta LP Models

Class Exercise: Pyrolysis of n-heptane



1. Develop a "straightforward" model that predicts the flow rate of methane from the reactor
2. Enhance this model by adding a delta due to changes in severity. Recommend the allowable range for the severity variable
3. Is the material balance closed in the base-delta approach?

Separable Programming

- Consider the following mathematical program:

$$\min_{\mathbf{x}} \quad z \triangleq \sum_{j=1}^{n} f_j(x_j) = f_1(x_1) + \cdots + f_n(x_n)$$

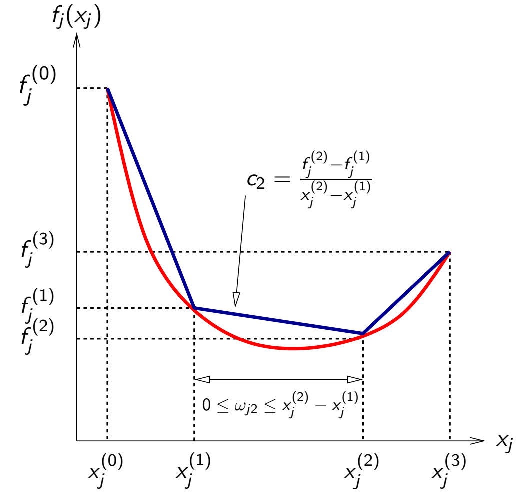
$$\text{s.t.} \quad \sum_{j=1}^{n} a_{ij} x_j \leq b_i, \quad i = 1, \ldots, m$$

$$x_j \geq 0, \quad j = 1, \ldots, n$$

  - The objective consists of n nonlinear, separable terms $f_j(x_j)$, each a function of a single variable only — **Examples?**
  - The m constraints are linear
- When *each* f_j is convex on the feasible region, the separable program can be approximated with an LP
- An analogous situation exists when the objective is to *maximize* a separable *concave* function — **Why?**

Approximating Separable Programs as LPs

Piecewise affine approximation:
(N_j intervals)



- Define:

$$c_j^{(k)} = \frac{f_j^{(k)} - f_j^{(k-1)}}{x_j^{(k)} - x_j^{(k-1)}}$$

$$0 \leq \omega_{jk} \leq x_j^{(k)} - x_j^{(k-1)}$$

- Substitute *each* variable x_j and function f_j by:

$$x_j \leftarrow x_j^{(0)} + \sum_{k=1}^{N_j} \omega_{jk}$$

$$f_j(x_j) \leftarrow f_j^{(0)} + \sum_{k=1}^{N_j} c_j^{(k)} \omega_{jk}$$

Approximating Separable Programs as LPs (cont'd)

Approximate Linear Program (on N_j Intervals):

$$\min_{\boldsymbol{\omega}} \quad \sum_{j=1}^{n} f_j^{(0)} + \sum_{j=1}^{n} \sum_{k=1}^{N_j} c_j^{(k)} \omega_{jk}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} \sum_{k=1}^{N_j} a_{ij} \omega_{jk} \leq b_i - \sum_{j=1}^{n} a_{ij} x_j^{(0)}, \quad i = 1, \ldots, m$$

$$0 \leq \omega_{jk} \leq x_j^{(k)} - x_j^{(k-1)}, \quad j = 1, \ldots, n, \ k = 1, \ldots, N_j$$

Important Remarks:

- Convexity guarantees that the pieces in the solutions will be included in the right order — This approach does not work if not *all* functions are convex!
- The solution can be made as accurate as desired by using enough intervals — One pays the price in terms of increased problem size!

Minimax and Maximin Problems

- Consider the case of multiple, competing *linear* objectives:

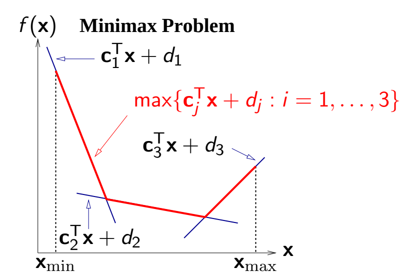
$$f_1(\mathbf{x}) \triangleq \mathbf{c}_1^{\mathsf{T}} \mathbf{x} + d_1, \quad \ldots, \quad f_N(\mathbf{x}) \triangleq \mathbf{c}_N^{\mathsf{T}} \mathbf{x} + d_N$$

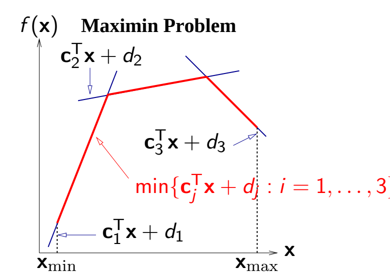
- Minimax problem: minimize the worst (greatest) objective:

$$\min_{\mathbf{x}} \ \max\{\mathbf{c}_i^{\mathsf{T}} \mathbf{x} + d_i : i = 1, \ldots, N\}$$

- Maximin problem: maximize the worst (least) objective:

$$\max_{\mathbf{x}} \ \min\{\mathbf{c}_i^{\mathsf{T}} \mathbf{x} + d_i : i = 1, \ldots, N\}$$





Formulating Minimax Problems

Class Exercise: Two groups of employees in a company are asked to work on Sundays, depending on the actual plant production x,

$$\text{Group 1:} \quad f_1(x) = 5x \qquad \text{Group 2:} \quad f_2(x) = 3x + 2$$

The CEO wants to minimize the maximum number of employees working on Sundays in all groups. Formulate a model for this optimization problem.

Reformulating Minimax and Maximin Problems

- **Idea:** Introduce a slack variable, along with N inequality constraints

Minimax Problem:

$$\min_{\mathbf{x}} \; \max\{\mathbf{c}_i^\mathsf{T}\mathbf{x} + d_i : i = 1, \ldots, N\}$$

$$\Downarrow$$

LP Model:

$$\begin{aligned} \min_{\mathbf{x},z} \quad & z \\ \text{s.t.} \quad & z \geq \mathbf{c}_1^\mathsf{T}\mathbf{x} + d_1 \\ & \vdots \\ & z \geq \mathbf{c}_N^\mathsf{T}\mathbf{x} + d_N \end{aligned}$$

Maximin Problem:

$$\max_{\mathbf{x}} \; \min\{\mathbf{c}_i^\mathsf{T}\mathbf{x} + d_i : i = 1, \ldots, N\}$$

$$\Downarrow$$

LP Model:

$$\begin{aligned} \max_{\mathbf{x},z} \quad & z \\ \text{s.t.} \quad & z \leq \mathbf{c}_1^\mathsf{T}\mathbf{x} + d_1 \\ & \vdots \\ & z \leq \mathbf{c}_N^\mathsf{T}\mathbf{x} + d_N \end{aligned}$$

- Additional linear constraints can be included in the formulations

Goal Programming

- Consider the case of multiple, competing *linear* objectives:

$$f_1(\mathbf{x}) \triangleq \mathbf{c}_1^\mathsf{T}\mathbf{x}, \quad \ldots, \quad f_N(\mathbf{x}) \triangleq \mathbf{c}_N^\mathsf{T}\mathbf{x},$$

  and corresponding target levels $\ell_1, \ldots, \ell_N$

- Find a **compromise** between the various goals in such a way that most are to some extent satisfied
  - Lower One-Sided Goal: Achieve a value of at least ℓ_k for the kth goal,

$$f_k(\mathbf{x}) = \mathbf{c}_k^\mathsf{T}\mathbf{x} \geq \ell_k$$

  - Upper One-Sided Goal: Achieve a value of at most ℓ_k for the kth goal,

$$f_k(\mathbf{x}) = \mathbf{c}_k^\mathsf{T}\mathbf{x} \leq \ell_k$$

  - Two-Sided Goal: Achieve a value of exactly ℓ_k for the kth goal,

$$f_k(\mathbf{x}) = \mathbf{c}_k^\mathsf{T}\mathbf{x} = \ell_k$$

Formulating Goal Programming as LP Models

Soft Constraints

Target levels specify requirements that are desirable to satisfy, but which may be violated in feasible solutions

- **Idea:** Introduce deficiency variables, $d_k^\pm$, representing the amount by which the kth goal is over- or under-achieved
  - Lower One-Sided Goal: $\mathbf{c}_k^\mathsf{T}\mathbf{x} + d_k^- = \ell_k, \quad d_k^- \geq 0$
  - Upper One-Sided Goal: $\mathbf{c}_k^\mathsf{T}\mathbf{x} - d_k^+ = \ell_k, \quad d_k^+ \geq 0$
  - Two-Sided Goal: $\mathbf{c}_k^\mathsf{T}\mathbf{x} + d_k^- - d_k^+ = \ell_k, \quad d_k^+, d_k^- \geq 0$
- Fundamental balances (such as material and energy balances) should **never** be softened! These must always be strictly observed
- Softening constraints may help debugging models in case infeasibility is reported

Formulating Goal Programming as LP Models (cont'd)

- **Non-Preemptive LP Model Formulation:** All the goals are considered simultaneously in the objective function,

$$\begin{aligned}\min_{\mathbf{x},\mathbf{d}^\pm} \quad & \boldsymbol{\omega}^\mathsf{T}\mathbf{d}^\pm \\ \text{s.t.} \quad & \mathbf{c}_k^\mathsf{T}\mathbf{x} \pm d_k^\mp = \ell_k, \quad k = 1, \ldots, N \\ & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}, \quad \mathbf{d}^\pm \geq \mathbf{0}\end{aligned}$$

  Determining the weights $\boldsymbol{\omega}$ is a subjective step... Different weights often yield very different solutions!

- **Preemptive LP Model Formulation:** The goals are subdivided into sets, and each set is given a priority
  - The solution proceeds by solving a sequence of subproblems, from highest to lowest priority goals
  - Goals of *lower* priority are ignored in a given subproblem
  - Goals of *higher* priority are enforced as hard equality constraints