AAE 526: Yet More Linear Programming

Thomas F. Rutherford

Fall Semester, 2019
Department of Agricultural and Applied Economics
University of Wisconsin, Madison

September 11, 2019

Outline

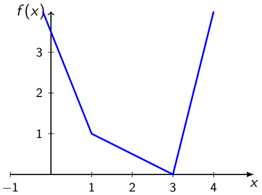
- Piecewise linear functions
- Minimax problems
- Canonical minmax model: Concert planning
- Canonical minmax model: Baker paint
- Blending models
- Canonical blending model: busing
- Canonical blending model: milk processing

Piecewise Linear Functions

- Some problems do not appear to be LPs but can be converted to LPs using a suitable transformation.
- An important case: *convex* piecewise linear functions.
- Consider the following **nonlinear** optimization:

$$\min_{x} f(x)$$

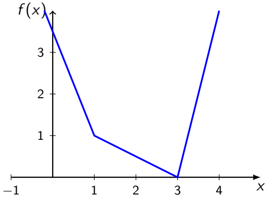
subject to: $x \geq 0$
Where $f(x)$ is the function:

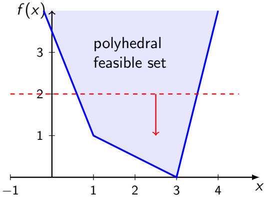


Piecewise linear functions

The trick is to convert the problem into epigraph form: add an extra decision variable t and turn the cost function into a constraint!

$$\min_x f(x) \quad \text{s.t.} \quad x \geq 0 \qquad \Rightarrow \qquad \min_{x,t} t \quad \text{s.t.} \quad t \geq f(x), \; x \geq 0$$



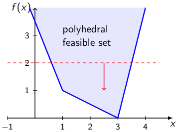


Piecewise linear functions

$$\min_{x,t}\ t$$

s.t.

$$t \geq f(x)$$
$$x \geq 0$$



This feasible set is **polyhedral**. It is the set of (x, t) satisfying:

$$\{t \geq -2x + 3, t \geq -\frac{1}{2}x + \frac{3}{2}, t \geq 3x - 9\}$$

The equivalent *linear program* is:

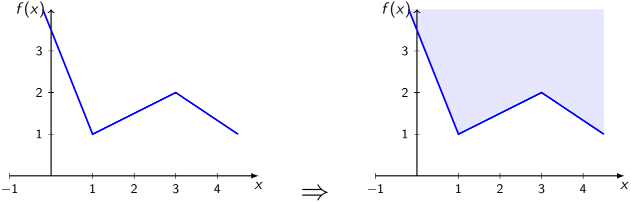
$$\min_{x,t}\ t$$

subject to:

$$t \geq -2x + 3$$
$$t \geq -\frac{1}{2}x + \frac{3}{2}$$
$$t \geq 3x - 9$$
$$x \geq 0$$

The epigraph "trick" only works for **convex** functions.

When convex, a piecewise linear function can be represented as a *system of affine inequalities*.



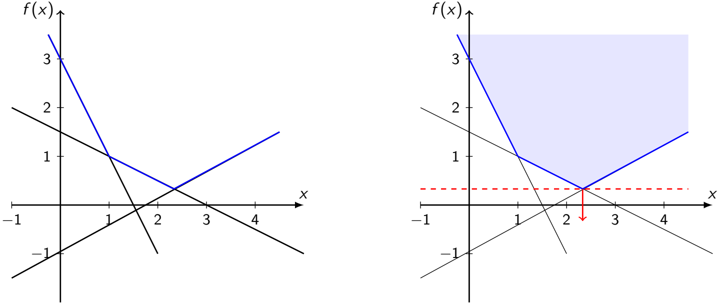
This epigraph is **not convex** so it cannot be the feasible set of a linear program.

Outline

Minimax problems

The maximum of several linear functions is *always* convex. So we can minimize it using the epigraph trick. Example:

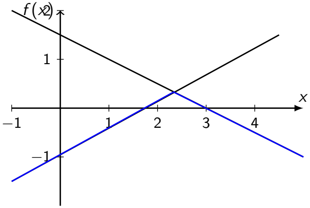
$$f(x) = \max_{i=1,\dots k} \left\{ a_i^T x + b_i \right\}$$

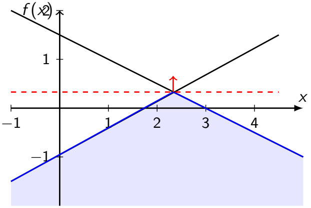


$$\min_x \max_{i=1,\dots,k} \left\{ a_i^T x + b_i \right\} \quad \Rightarrow \quad \begin{aligned} &\min_{x,t} \; t \\ &\text{s.t.} \quad t \ge a_i^T x + b_i \quad \forall i \end{aligned}$$

The minimum of several linear functions is *always* concave. So we can minimize it using the epigraph trick. Example:

$$f(x) = \min_{i=1,\dots k} \left\{ a_i^T x + b_i \right\}$$



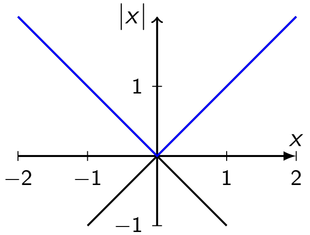


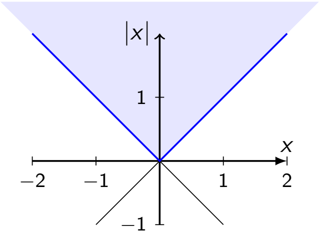
$$\max_x \min_{i=1,\dots,k} \left\{ a_i^T x + b_i \right\} \qquad \Rightarrow \qquad \begin{aligned} &\max_{x,t} \; t \\ \text{s.t.} \quad & t \le a_i^T x + b_i \quad \forall i \end{aligned}$$

Absolute Value

The absolute value function is piecewise linear and convex: the epigraph "trick" works here as well.

$$\min_{x \in \mathbb{R}^n} |x_j| \quad \text{s.t.} \quad Ax \leq b \qquad \Rightarrow \qquad \begin{aligned} \min_{x,t}\ & t \\ \text{s.t.}\quad & Ax \leq b \\ & t \geq x_j \\ & t \geq -x_j \end{aligned}$$





Outline

- Piecewise linear functions
- Minimax problems
- Canonical minmax model: Concert planning
- Canonical minmax model: Baker paint
- Blending models
- Canonical blending model: busing
- Canonical blending model: milk processing

Concert Planning

The promoters of a rock concert must perform the tasks shown in the GAMS code below before the concert can be held:

```
set activity /
        A        "Find Site",
        B        "Find Engineers",
        C        "Hire Opening Act",
        D        "Set Radio and TV Ads",
        E        "Set Up Ticket Agents",
        F        "Prepare Electronics",
        G        "Print Advertising",
        H        "Set up Transportation",
        I        "Rehearsals",
        J        "Last-Minute Details"/;
```

Concert Planning (cont.)

Each activity requires some time:

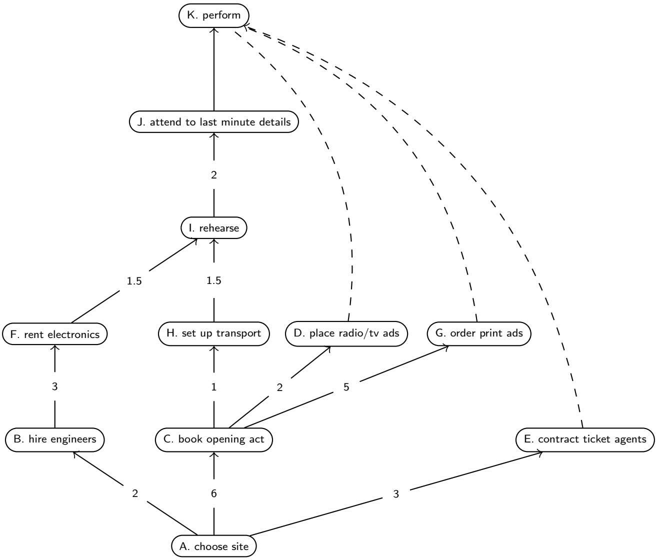
```
parameter duration(activity) "in days" /
        A 3, B 2, C 6, D 2, E 3, F 3, G 5, H 1, I 1.5, J 2 /;
```

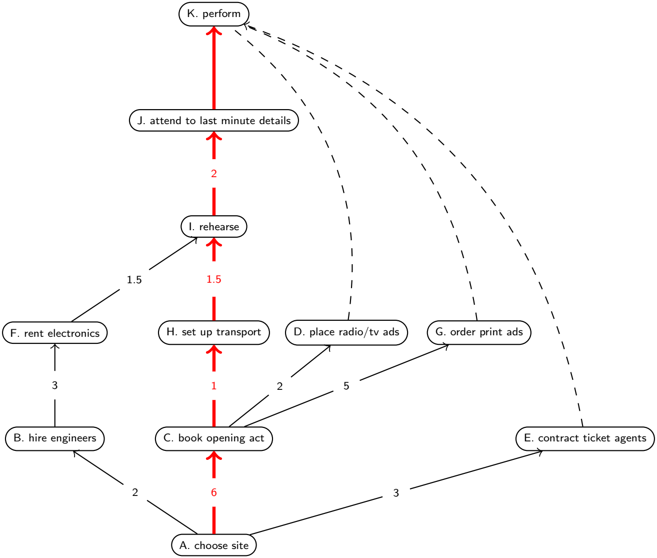
Certain activities must be completed before subsequent activities may be initiated. For instance, a concert site must be found before we find engineers, hire an opening act or set up ticket agents, and so forth.

```
alias (activity,i,j);

set prec(i,j) "Precedence order" /
        A.(B,C,E)
        B.F
        C.(D,G,H)
        (F,H).I
        I.J /;
```

Set up a linear program to find the project duration (that is the minimum number of days needed to prepare for the concert).


K. perform
J. attend to last minute details
2
I. rehearse
1.5
1.5
F. rent electronics
H. set up transport
D. place radio/tv ads
G. order print ads
3
1
2
5
B. hire engineers
C. book opening act
E. contract ticket agents
2
6
3
A. choose site


K. perform
J. attend to last minute details
2
I. rehearse
1.5
1.5
F. rent electronics
H. set up transport
D. place radio/tv ads
G. order print ads
3
1
2
5
B. hire engineers
C. book opening act
E. contract ticket agents
2
6
3
A. choose site

Concert Planning Model - Scalar

```
$title  Critical Path Model -- Scalar Format

FREE
VARIABLE
        T       Completion time;

NONNEGATIVE
VARIABLES
        A       "Start time: find site",
        B       "Start time: find engineers",
        C       "Start time: hire opening act",
        D       "Start time: set radio and TV ads",
        E       "Start time: set up ticket agents",
        F       "Start time: prepare electronics",
        G       "Start time: print advertising",
        H       "Start time: set up transportation",
        I       "Start time: rehearsals",
        J       "Start time: last-minute details";
```

Concert Planning Model - Scalar

```
equations

*       Completion times no sooner than any of the tasks:

t_A, t_B, t_C, t_D, t_E, t_F, t_G, t_H, t_I, t_J

*       Task which must be completed before subsequent task:

s_AB, s_AC, s_AE, s_BF, s_CD, s_CG, s_CH, s_FI, s_HI, s_IJ;
```

Concert Planning Model - Scalar

```
t_A.. T =G= A+3; t_B.. T =G= B+2; t_C.. T =G= C+6;
t_D.. T =G= D+2; t_E.. T =G= E+3; t_F.. T =G= F+3;
t_G.. T =G= G+5; t_H.. T =G= H+1; t_I.. T =G= I+1.5;
t_J.. T =G= J+2;

s_AB.. A+3 =L= B; s_AC.. A+3 =L= C; s_AE.. A+3 =L= E;
s_BF.. B+2 =L= F; s_CD.. C+6 =L= D; s_CG.. C+6 =L= G;
s_CH.. C+6 =L= H; s_FI.. F+3 =L= I; s_HI.. H+1 =L= I;
s_IJ.. I+1.5 =L= J;

model cpm /all/;
solve cpm using lp minimizing T;
```

Use of Tuples in GAMS

A *tuple* in GAMS is a multi-dimensional set. It corresponds to a logical (yes/no) data structure. Consider the following code fragment:

```
Set     i /i1*i4/,
        j /j1*j5/,
        ij(i,j) /i1.j1, i2.(j1,j2), (i3,i4).(j3*j5)/;

option ij:0:0:8;

display i, j, ij;
```

which produces the following listing:

```
----      5 SET i
i1,    i2,    i3,    i4

----      5 SET j
j1,    j2,    j3,    j4,    j5

----      5 SET ij
i1.j1,    i2.j1,    i2.j2,    i3.j3,    i3.j4,    i3.j5,    i4.j3,    i4.j4
i4.j5
```

Use of Tuples in GAMS

A *tuple* can be used to restrict assignments, either as a mask over the set domain:

```
parameter a(i,j), b(i,j);

a(i,j) = uniform(0,1);

b(ij(i,j)) = a(i,j);

display a,b;
```

```
----      14 PARAMETER a

             j1          j2          j3          j4          j5
i1        0.172       0.843       0.550       0.301       0.292
i2        0.224       0.350       0.856       0.067       0.500
i3        0.998       0.579       0.991       0.762       0.131
i4        0.640       0.160       0.250       0.669       0.435


----      14 PARAMETER b

             j1          j2          j3          j4          j5
i1        0.172
i2        0.224       0.350
i3                                0.991       0.762       0.131
i4                                0.250       0.669       0.435
```

Declaration of Multi-Connected Tuples

Notice that:

```
set prec(i,j) "Precedence order" /
A.(B,C,E), B.F, C.(D,G,H), (F,H).I, I.J /;
```

is equivalent to

```
set prec(i,j) "Precedence order" /
A.B, A.C, A.E, B.F, C.D,C.G,C.H, F.I,H.I, I.J /;
```

Likewise

```
set ij(i,j) /(A,B).(C,D,E)/;
```

is equivalent to:

```
set ij(i,j) /A.C,A.D,A.E, B.C,B.D,B.E)/;
```

Concert Planning: An Indexed Formulation

```
set activity / A*J/;    alias (activity,i,j);

parameter duration(activity) "in days" /
  A 3, B 2, C 6, D 2, E 3, F 3, G 5, H 1, I 1.5, J 2 /;

set prec(i,j) "Precedence order" /
  A.(B,C,E), B.F, C.(D,G,H), (F,H).I, I.J /;

FREE VARIABLE        T       Time to completion;

NONNEGATIVE VARIABLE S(i)    Starting time for activity i;

EQUATIONS   ctime(i) Completion time,  ptime(i,j) Sequence;

ctime(i)..          T =g= S(i) + duration(i);

ptime(prec(i,j)).. S(i) + duration(i) =L= S(j);

model cpm /all/;   solve cpm using lp minimizing T;
```

Outline

- Piecewise linear functions
- Minimax problems
- Canonical minmax model: Concert planning
- Canonical minmax model: Baker paint
- Blending models
- Canonical blending model: busing
- Canonical blending model: milk processing

Canonical application: Baker Paint Company

The Baker Paint Company produces and markets a large line of inside and outside paints through hardware stores and lumber companies and its own retail outlets. In order to guarantee rapid delivery to customers, Baker operates a number of warehouses and a fleet of trucks.

In New York, sales had reached the point where management believed a new warehouse was necessary. There were a number of factors which had to be considered in locating a new warehouse. The most important one, however, was the total gallon-miles traveled from warehouse to customers since total costs correlated quite closely to this figure.

Baker (cont.)

The company figured that it cost an average of \$0.15 to ship 1000 gallons of paint one mile. The cost included gasoline, oil, maintenance for the outward trip and the return trip from the customer when the truck would be empty.

For each customer in the area, the present annual sales volume and growth rate were computed and yearly sales estimates were made for each of the next five years. Denote the demand at node $n \in \{a, \ldots, z\}$ as Q_n and the location as $(\bar{x}_n, \bar{y}_n)$.

An LP Model

Formulate a linear programming model which determines the optimal location of the warehouse minimizing gallon-miles, assuming that roads only run east-west and north-south.

Given the orientation of roads, the distance between point (X, Y) and node n is given by:

$$\mathcal{D}_n(X, Y) = |X - \bar{x}_n| + |Y - \bar{y}_n|$$

We thus wish to solve:

$$\min Z = \sum_n Q_n \times \mathcal{D}_n(X, Y) = \sum_n Q_n \times (|X - \bar{x}_n| + |Y - \bar{y}_n|)$$

Absolute Value Formulation

This can be converted to a linear program by the introduction of two sets of variables Δ_n^x and Δ_n^y:

$$\min_{\Delta_n^x, \Delta_n^y, X, Y} Z = \sum_n Q_n \times (\Delta_n^x + \Delta_n^y)$$

subject to:

$$\Delta_n^x \geq X - \bar{x}_n$$

$$\Delta_n^x \geq \bar{x}_n - X$$

$$\Delta_n^y \geq Y - \bar{y}_n$$

$$\Delta_n^y \geq \bar{y}_n - Y$$

GAMS Model: Baker Paint

```
set     n /a*z/;

table   data(*,*)
        x       y       demand
a       2.16    2.44    10
b       2.32    2.43    2
c       3.45    2.08    5
d       4.25    2.47    2
e       4.37    2.83    1
f       4.09    2.87    4
g       5.09    2.59    3
h       3.21    2.88    6
i       2.25    3.13    7
j       2.03    4.13    4
k       2.54    3.71    2
l       3.27    3.37    1
m       4.35    3.46    9
n       5.12    3.27    4
o       4.84    4.05    5
p       3.81    3.83    3
q       2.99    4.19    4
r       2.47    4.6     1
s       3.07    4.47    5
t       4.14    4.53    2
u       3.77    4.7     6
v       2.63    4.81    1
w       3.45    5.19    10
x       4.65    4.45    2
y       4.85    4.63    8
z       4.53    5.22    10;
```

GAMS Model: Baker Paint Logic

```
parameter
       d(n)    Demand by node
       xl(n)   Customer location by node (east-west dimension X)
       yl(n)   Customer location by node (north-south dimension Y);

d(n) = data(n,"demand");
xl(n) = data(n,"x");
yl(n) = data(n,"y");

variables       C       Total Shipping cost,
                X       X location of optimal location,
                Y       Y location of optimal location;

nonnegative
variables       XD(n)   X dimension distance,
                YD(n)   Y dimension distance;

equations nXD, pXD, nYD, pYD, costdef;

nXD(n)..        XD(n) =g= X - xl(n);
pXD(n)..        XD(n) =g= xl(n) - X;
nYD(n)..        YD(n) =g= Y - yl(n);
pYD(n)..        YD(n) =g= yl(n) - Y;

*       Measure cost as a demand-weighted average distance:

costdef..       C =e= sum(n, d(n)*(XD(n)+YD(n)));

model grid /nXD, pXD, nYD, pYD, costdef/;

solve grid using LP minimizing C;
```

Alternative Model: The Box Norm

While unit transport cost is a meaningful metric, an alternative measure of proximity is the *box-norm*, i.e.

$$Z = \max_n |X - \bar{x}_n| + |Y - \bar{y}_n|$$

This provides a measure of the worst-case performance of delivery service to customers serviced by the warehouse. The warehouse location problem can then be formulated using the mini-max paradigm:

```
equation        costLinf        L-infinity cost metric;

costLinf(n)..   C =g= XD(n) + YD(n);

model gridLinf /nXD, pXD, nYD, pYD, costLinf/;

solve gridLinf using LP minimizing C;
```

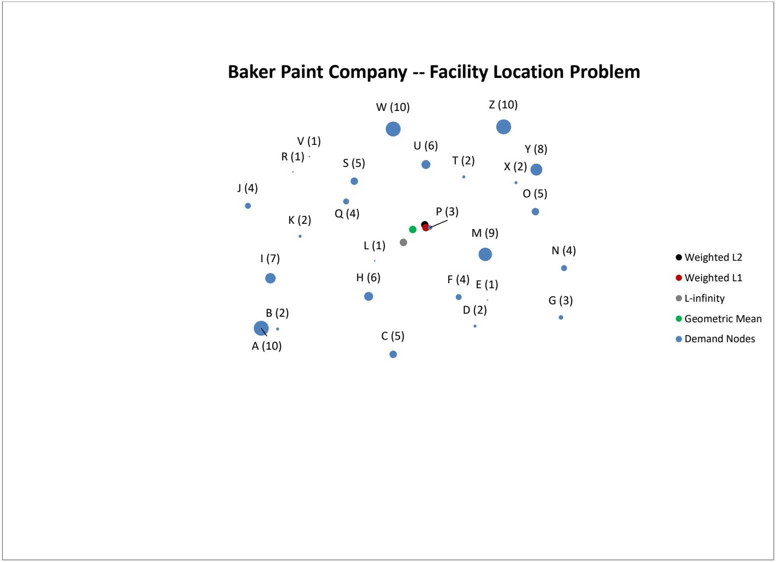
Yet Another Model: The Euclidian Norm

In some circumstances, the conventional Euclidian (L2) distance norm may be preferred. In this case, we might then solve:

$$\min Z = \sum_{n} Q_n \times \ell_n(X, Y) = \sum_{n} Q_n \times \sqrt{(X - \bar{x}_n)^2 + (Y - \bar{y}_i)^2}$$

With this objective function, the facility problem becomes a quadratic program, one for which there are numerous suitable solvers.

It depends?


Baker Paint Company -- Facility Location Problem
W (10)
Z (10)
V (1)
R (1)
U (6)
S (5)
T (2)
Y (8)
X (2)
J (4)
O (5)
P (3)
Q (4)
K (2)
M (9)
L (1)
N (4)
I (7)
H (6)
F (4)
E (1)
G (3)
D (2)
B (2)
C (5)
A (10)
Weighted L2
Weighted L1
L-infinity
Geometric Mean
Demand Nodes

Outline

- Piecewise linear functions
- Minimax problems
- Canonical minmax model: Concert planning
- Canonical minmax model: Baker paint
- Blending models
- Canonical blending model: busing
- Canonical blending model: milk processing

Blending

Interpreted literally, a *blending constraint* is nonlinear. Consider a constraint "the ith input share can be no greater than θ_i". If the ith input is x_i, we could write this as:

$$\frac{x_i}{\sum_{i'} x_{i'}} \leq \theta_i$$

But, if we multiply through by the total, the constraint is linear:

$$x_i \leq \theta_i \left(\sum_{i'} x_{i'} \right)$$

Note that in GAMS syntax, symbol i' in this equation is an `alias` of i

Outline

- Piecewise linear functions
- Minimax problems
- Canonical minmax model: Concert planning
- Canonical minmax model: Baker paint
- Blending models
- Canonical blending model: busing
- Canonical blending model: milk processing

Madison contains three school districts. the number of minority and non-minority students in each district is given below:

District	Minority	Non-Minority
1	50	200
2	50	250
3	100	150

The local court has decided that East and West High Schools must have approximately the same percentage of minorities (with $\pm 5\%$) as the whole city. The distances (in miles) between the school districts and the high schools are given below:

District	East	West
1	1	2
2	2	1
3	1	1

Each high school must have an enrollment of 300-500 students. Determine the percentage of students from each district assigned to schools that minimizes the total distance students must travel to school.

GAMS Code – Data

```
set     i District      /1*3/,
        r Racial type   /M      Minority,
                         N      Non-minority/,
        s Schools       /E      East High School,
                         W      West High School/;

table pop(i,r)  Population
                M       N
        1       50      200
        2       50      250
        3       100     150

table dist(i,s) Distances

                E       W
        1       1       2
        2       2       1
        3       1       1;
```

GAMS Code – Assignments

```
*       Extract some statistics from the problem data
*       using parameter assignments:

parameter       popshr(r)       Share of population of race r
                totpop          Total population of students;

totpop    = sum((i,r), pop(i,r));
popshr(r) = sum(i, pop(i,r)) / totpop;
```

GAMS Code – Model

```
*       Declare decision variables (conforming to the
*       capitalization convention for variables):

NONNEGATIVE
VARIABLES       E(s)            Enrollment,
                B(i,r,s)        Number of students bussed,
                X(r,s)          Racial populations;

VARIABLE        D               Total distance travelled;

EQUATIONS  edef, allstudents, travel, fraction, lower, upper;
```

GAMS Code – Model Logic

```
*       The enrollment at school s includes the students
*       from all districts of both racial types:

edef(s)..        E(s) =e= sum((i,r), B(i,r,s));

*       All students need to go to school:

allstudents..    sum(s, E(s)) =e= totpop;

*       The total distance travelled depends on the
*       school assignments and the distances:

travel..         D =e= sum((i,r,s), dist(i,s)*B(i,r,s));
```

GAMS Code – Model

```
*       Racial composition of each school:

fraction(r,s).. X(r,s) =e= sum(i, B(i,r,s));

*       Blending constraints: lower bound on
*       racial composition:

lower(r,s)..    X(r,s) =g= 0.95 * popshr(r) * E(s);

*       Blending constraints: upper bound on
*       racial composition:

upper(r,s)..    X(r,s) =l= 1.05 * popshr(r) * E(s);
```

GAMS Code – Bounds and Solution

```
*       Use the simple declaration -- all equations enter
*       the model:

model busing /all/;

*       Bounds on school size:

E.LO(s) = 300;
E.UP(s) = 500;

*       Solve the model using linear programming:

solve busing using lp minimizing D;
```

Outline

- Piecewise linear functions
- Minimax problems
- Canonical minmax model: Concert planning
- Canonical minmax model: Baker paint
- Blending models
- Canonical blending model: busing
- Canonical blending model: milk processing

Milk Processing

Happy Milk Distributors (HMD) purchases raw milk from farmers in regions A and B. Prices, butterfat content and separation properties of the raw milk differ between the two regions. HMD processes the raw milk to produce cream and milk to desired specifications for distribution to the consumers.

The Model (cont.)

Milk from region A is 54 cents per gallon up to 500 gallons and 58 cents per gallon in excess of 500 gallons. There is no upper bound on the amount that can be purchased. Raw milk from Region A has 25% butterfat and when separated (at 5 cents per gallon) it yields two "milks", one with 41% butterfat and another with 12% butter fat.
The volume of milk is conserved in all separation processing.

The Model (cont.)

The purchase price for milk from region B is 38 cents per gallon up to 700 gallons and 42 cents per gallon thereafter. Raw milk from Region B has 15% butterfat and when separated (at 7 cents per gallon) yields two “milks”, one with 43% butterfat and another with 5% butterfat.

The Model (cont.)

After the milk is purchased and collected at the plant, it is either mixed directly or separated and then mixed. The mixing is done at no cost, and its purpose is to produce cream and milk to specifications. For example, some of the raw milk from Region A may be separated and then mixed, and some of it may be mixed directly (i.e., without having been separated).

Sets

```
SET     r       Regions         /A,B/
        c       Cost            /lowcost,highcost/,
        i       Products        /cream, milk/,

        t       Types of raw milk inputs /
                        raw     Raw (unseparated) milk,
                        highfat High fat (separated),
                        lowfat  Low fat (separated)/,

        s(t)    Separated milk /highfat,lowfat/;
```

Tables

```
TABLE    costr(r,c)      Cost of raw milk
         lowcost highcost
A        54      58
B        38      42;

TABLE    supply(r,c)     Supplies of raw milk
         lowcost highcost
A        500     +inf
B        700     +inf;

TABLE     bf(r,*)        Butterfat content
         RAW     highfat lowfat
A        25      41      12
B        15      43       5;
```

Parameters (vectors)

```
PARAMETER

cs(r)      Cost of separation per gallon    / A 5, B 7/,
p(i)       Sales price by product           /cream 150, milk 70/,
bfat(i)    Butterfat requirements           /cream 40, milk 20/,
maxd(i)    Maximum demand for products      /cream 250, milk 2000/;
```

Variable and Equation Declarations

```
*       Declare and reference variables in upper case so that they
*       are easily distinguished from sets and data parameters:

VARIABLES
        PI                Profit,
        Z(i)              Output of milk products
        X(r,c)            Supply of raw milk by region and cost,
        Y(i,r,t)          Supply of raw or separated milk by region;

NONNEGATIVE VARIABLES X, Y, Z;

EQUATIONS
        profit            Value of sales less costs,
        bfreq(i)          Butterfat requirments constraint,
        bfcont(r)         Butterfat content of separated milk,
        product(i)        Product supply,
        msupply(r)        Milk supply;
```

Equation Definitions

```
profit..        PI =e=    sum(i,

*       Value of output products:

                p(i) * Z(i))

*       Cost of milk purchases:

                - sum((r,c), costr(r,c) * X(r,c))

*       The volume of separated milk equals the total volume of
*       raw milk purchased less the volume of milk which is used
*       in unseparated form:

             + sum(r, cs(r)*(sum(c,X(r,c))-sum(i,Y(i,r,"raw"))));
```

Equation Definitions (cont)

```
*       This is the butterfat blending constraint.  The butter fat
*       contained in the constituent products must be greater than
*       or equal to the butter fat requirement:

bfreq(i)..
        sum((r,t), Y(i,r,t) * bf(r,t)) =g= bfat(i) * Z(i);

*       The butterfat content of separated milk has the same
*       butterfat content of the constitute raw milk input:

bfcont(r)..
        bf(r,"raw") * sum((i,s), Y(i,r,s))

                        =g= sum((i,s), bf(r,s)*Y(i,r,s));
```

Equation Definitions (cont)

```
*       Product output equals the volume of all raw milk purchased
*       for use in this product:

product(i)..    Z(i) =L= sum((r,t), Y(i,r,t));

*       The supply of milk from region r equals the utilization
*       of milk from region r:

msupply(r)..    sum(c, X(r,c)) =G= sum((i,t), Y(i,r,t));
```

Simple Bounds (cont)

```
*       There are upper bounds on the volume of milk which
*       can be purchased by region and by price category:

X.UP(r,c) = supply(r,c);

*  There are upper bounds on the milk which can be produced:

Z.UP(i) = maxd(i);
```

Model Declaration, Solution and Report

```
*       The model consists of all the declared equations:

MODEL hmd /all/;

SOLVE hmd USING LP MAXIMIZING pi;

*       Generate a summary report of the optimal
*       program:

parameter       solution        Model solution;
solution(i,r) = sum(t,Y.L(i,r,t));
solution(c,r) = X.L(r,c);
solution(t,r) = sum(i,Y.L(i,r,t));
display solution;
```

Listing File

```
----     147 PARAMETER solution  Model solution

                   A           B

lowcost      500.000     700.000
highcost    1080.000
highfat      708.276     184.211
lowfat       871.724     515.789
cream         48.611     201.389
milk        1531.389     498.611
```