Extensions to Mathematical Programming

G. Cornelis van Kooten
REPA Group
University of Victoria

Topics

1. Integer programming in forestry
2. Stochastic Dynamic Programming
3. Calibrating models
   – Positive mathematical programming
   – Mixes

- Many examples from Joseph Buongiorno & J. Keith Gilless, 2003. *Decision Methods in Forest Resource Management* (Academic Press Elsevier)

1. Integer Programming

- Many resource allocation problems involve lumpy decisions
  - Harvest entire site or not (forestry)
  - Plant entire field to same crop (agriculture)
  - Purchase new truck or tractor-trailer (transportation)
  - Purchase new combine/tractor (agriculture)
  - Build new warehouse (warehousing/manufacturing)
- Fixed investments are integer in nature

Example (Buongiorno & Gilless Chap 11)

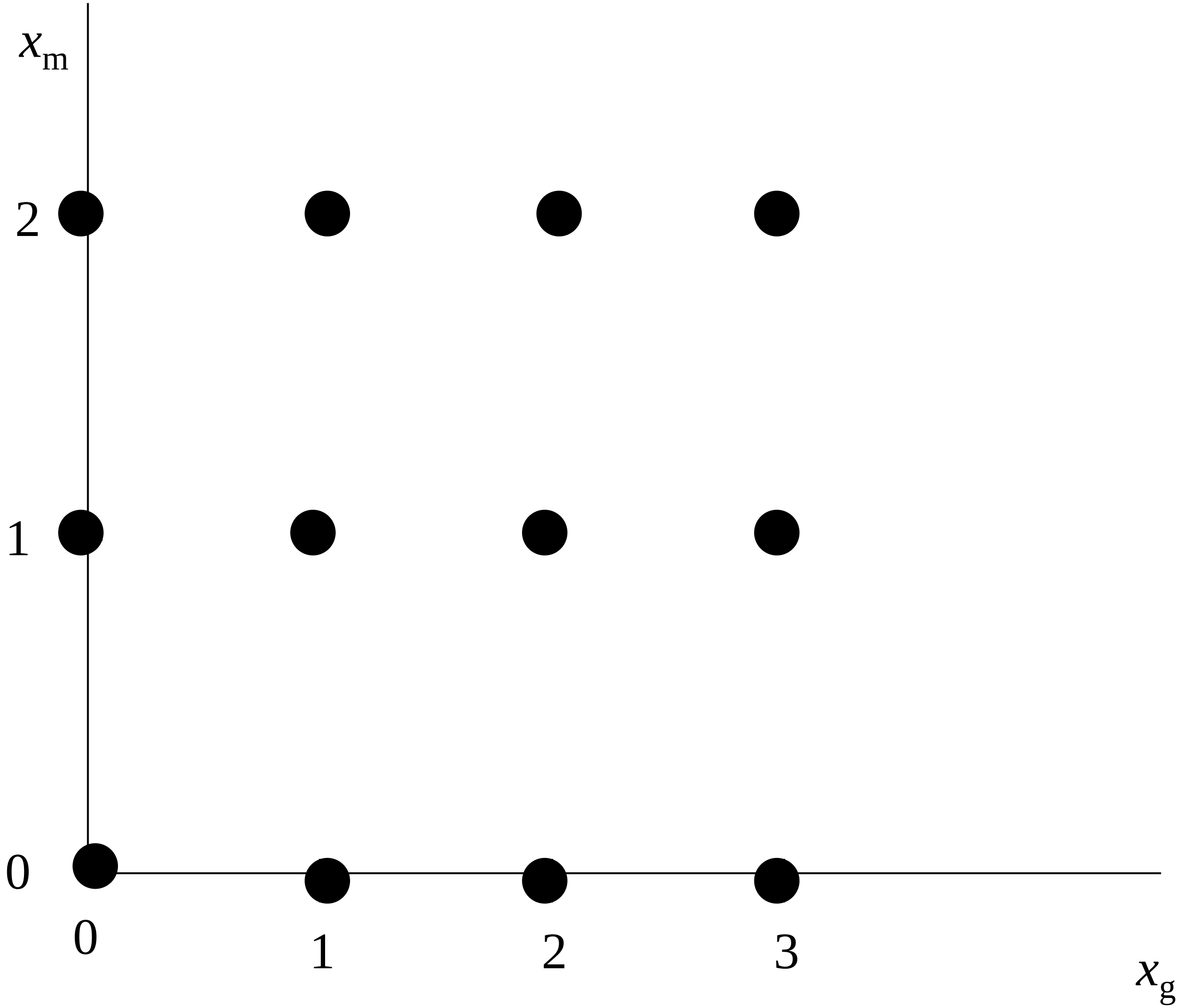
- Forest consulting firm can contract for five projects, three in Georgia and two in Michigan. Each of the Georgia projects requires 1-person year and returns $10,000 profit; each Michigan project requires 10-person years and returns $50,000.
- Firm has a staff of 20
- Let $x_g = \{0, 1, 2, 3\}$ and $x_m = \{0, 1, 2\}$.

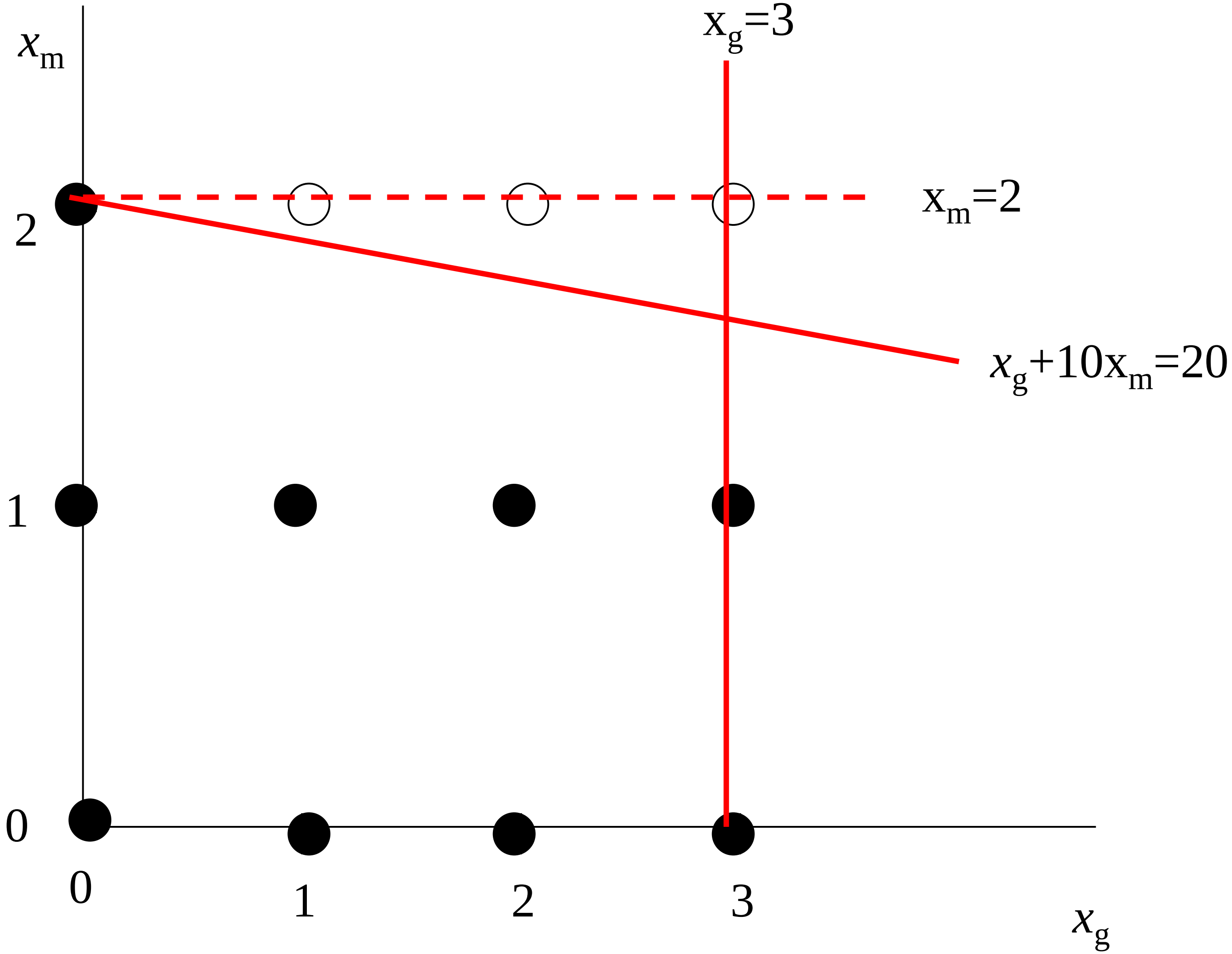
Integer Programming Example

- **Problem**:

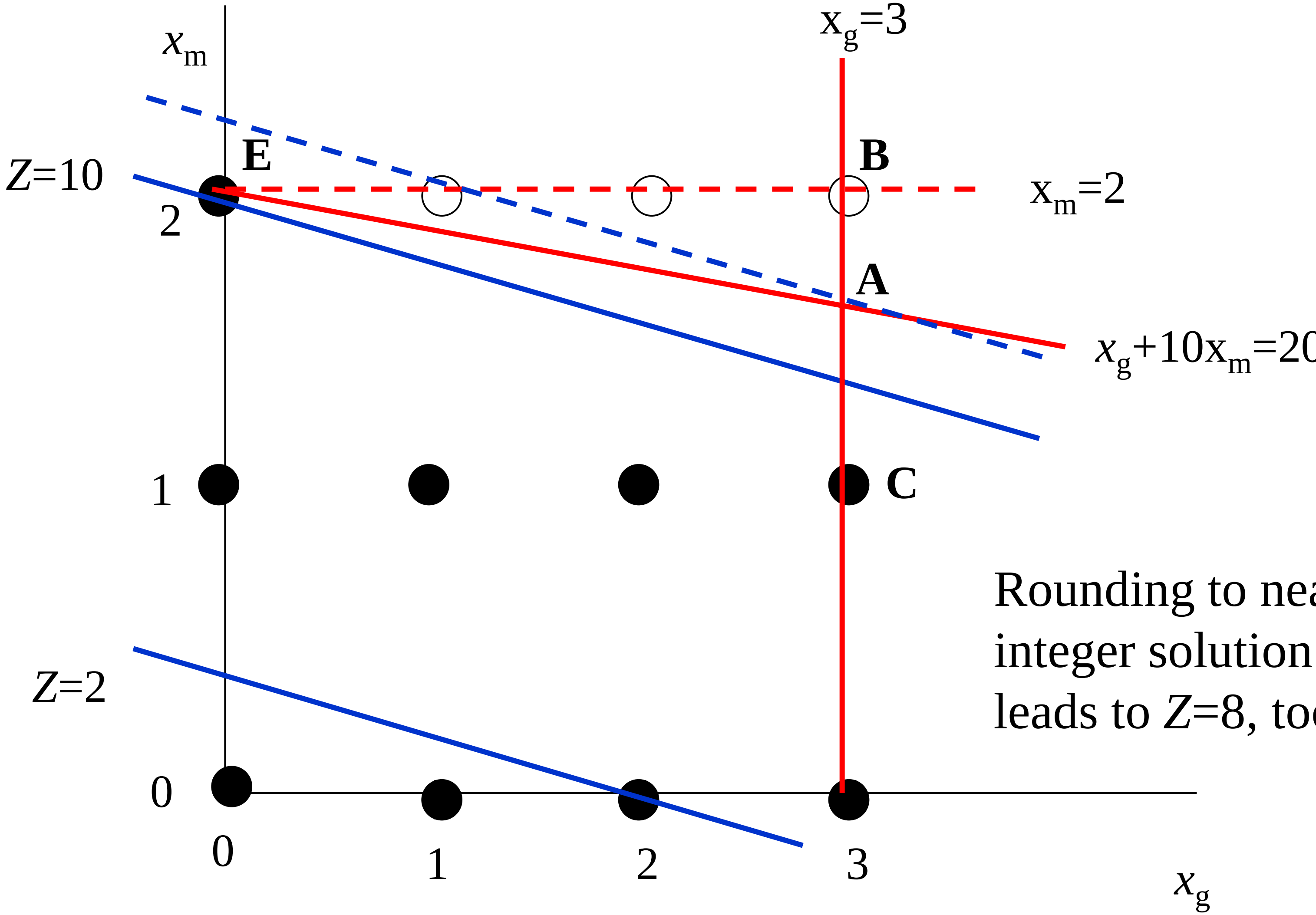
$$
\begin{aligned}
\text{Max} \quad & Z = x_g + 5\, x_m \\
\text{s.t.} \quad & x_g \le 3 && \text{(Georgia projects)} \\
& x_m \le 2 && \text{(Michigan projects)} \\
& x_g + 10\, x_m \le 20 && \text{(person-years)} \\
& x_g, x_m \ge 0
\end{aligned}
$$

Possible solutions to the problem





Constraints: Notice one is redundant. Some solutions are subsequently ruled out.



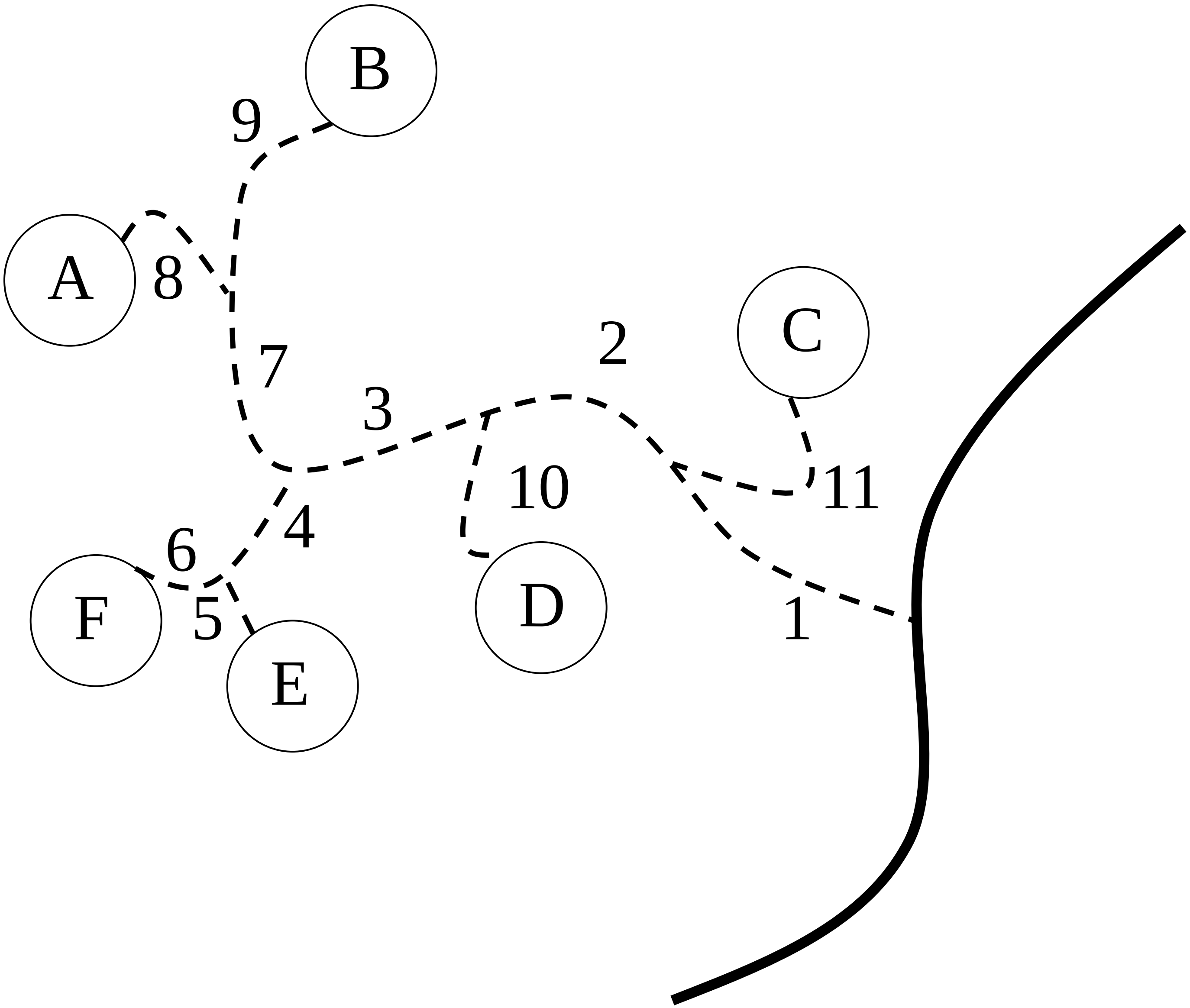
Rounding to nearest integer solution **C** leads to *Z*=8, too low.

IP solution occurs at point **E**, while LP would give the solution at point **A**, which is infeasible.

Second integer programming example

The Ministry of Forests is considering six multiple-use projects to provide timber and hunting opportunities. Each is represented in the figure by letters. Each project must be connected by a road to the existing road shown as a solid line. The dashed lines are the road sections that *might* be built. Each road section is identified by a number. The Ministry's objectives are to:

1. minimize road construction costs
2. provide at least 400 hunting days per year
3. harvest at least 30,000 m^3 per year of timber
4. get enough revenues from projects to cover total costs of road construction.



Forest sites and road network diagram

Three questions

1. What is timber production and number of hunting days? What is total net present value? Each project must be done completely or not at all.
2. Assume projects are perfectly divisible and that outputs and revenues of projects are directly proportional to the scale at which they are undertaken. How does this solution compare to 1?
3. Road section 1 will clearly bear more traffic than other sections, and needs to be built to a higher standard. How is that taken into account?

Constructing the IP in GAMS

1. Set up the model analytically
2. Use the analytical representation of the objective function and constraints to develop the GAMS model

Let s refer to road segments, p to projects/sites, c to construction costs, h to hunting days, t to timber and r to revenues

Min $\sum_i c_i s_i$ i=1, …, 11 (road building costs)

s.t. $\sum_j h_j p_j \geq 400$ j=1, …, 6 (total hunting days)

$\sum_j t_j p_j \geq 30$ j=1, …, 6 (total timber harvest)

$\sum_j r_j p_j \geq \sum_i c_i s_i$ (revenue exceeds costs)

Constraints that associate projects with road segments

$p_A \leq s_8$ $p_B \leq s_9$ $p_C \leq s_{11}$

$p_D \leq s_{10}$ $p_E \leq s_5$ $p_F \leq s_6$

Constraints to ensure a collector road is built if branches are built

$s_{11} + s_2 \leq 2\ s_1$ $s_{10} + s_3 \leq 2\ s_2$ $s_5 + s_6 \leq 2\ s_4$

$s_4 + s_7 \leq 2\ s_3$ $s_8 + s_9 \leq 2\ s_7$

p_j, s_i binary

p refers to projects, *s* to road segments

Road building costs

	Road Section										
	1	2	3	4	5	6	7	8	9	10	11
Cost ($\$10^3$)	75	50	65	40	45	70	50	40	20	50	25

Benefits associated with each project

	Project					
	A	B	C	D	E	F
Hunting days (per yr)	200	300	100	100	200	300
Timber (10^3 m^3/yr)	6	9	13	10	8	3
Revenues ($\$10^3$)	70	130	140	130	110	100

GAMS program found here

gamside: C:\Documents and Settings\kooten.UVIC\My Documents\Classes\Programming\GAMS\Programs\class_examples.g...

File Edit Search Windows Utilities Help

HW#3_2006.gms | milk.gms | shipping.gms

```
* This solves the road building integer program of
* Joseph Buongiorno and J. Keith Gilless (2003, problem 11.8, pp.229-31)

sets r road segments and projects /s1*s11/
     p projects /A, B, C, D, E, F/
     i site uses /hunting, timber, revenue, traffic/
     k additional constraints /1*5/;

Parameter cost(r) cost of road segments in $000s

/ s1  75
  s2  50
  s3  65
  s4  40
  s5  45
  s6  70
  s7  50
  s8  40
  s9  20
  s10 50
  s11 25/ ;

Parameter b(i) RHS requirements /hunting 400, timber 30, revenue 0, traffic 0/;

Table a(p,i) table of technical coefficients for equations
```

1: 52 Modified Insert

Start C:\Document... LP_extension... gamside 80% 9:16 AM

gamside: C:\Documents and Settings\kooten.UVIC\My Documents\Classes\Programming\GAMS\Programs\class_examples.g...

File Edit Search Windows Utilities Help

HW#3_2006.gms | milk.gms | shipping.gms

```
Table a(p,i) table of technical coefficients for equations
        hunting  timber  revenue  traffic
    A     200       6       70        8
    B     300       9      130        4
    C     100      13      140       14
    D     100      10      130       13
    E     200       8      110       10
    F     300       3      100        7
;

Table segproj(k,r) table relating road segments to projects
        s1  s2  s3  s4  s5  s6  s7  s8  s9  s10  s11
    1   -2  1   0   0   0   0   0   0   0   0    1
    2   0   -2  1   0   0   0   0   0   0   1    0
    3   0   0   0   -2  1   1   0   0   0   0    0
    4   0   0   -2  1   0   0   1   0   0   0    0
    5   0   0   0   0   0   0   -2  1   1   0    0
;

Variables
zroad           road costs
segment(r)      whether to build road segment
project(p)      whether to build project
;
```

1: 52 Modified Insert

Start | C:\Document... | LP_extension... | gamside | 79% | 9:18 AM

gamside: C:\Documents and Settings\kooten.UVIC\My Documents\Classes\Programming\GAMS\Programs\class_examples.g...

File Edit Search Windows Utilities Help

HW#3_2006.gms | milk.gms | shipping.gms

```
Binary variables segment, project;
*Positive variables project;

Equations
obj        objective function is minimize road building cost
hunt       minimum hunting days constraint
harvest    minimum timber harvest constraint
revenue    revenue must exceed costs of road building
totseg(k)  segments per project must balance
psbal1     Following equations balance projects and segments
psbal2
psbal3
psbal4
psbal5
psbal6
contproj(p)
;

obj..   zroad =E= sum(r, cost(r)*segment(r));

hunt..  sum(p, a(p,'hunting')*project(p)) =G= b('hunting');

harvest..  sum(p, a(p,'timber')*project(p)) =G= b('timber');
```

1: 52 Modified Insert

Start | C:\Document... | LP_extension... | gamside | 79% | 9:19 AM

gamside: C:\Documents and Settings\kooten.UVIC\My Documents\Classes\Programming\GAMS\Programs\class_examples.g...

File Edit Search Windows Utilities Help

HW#3_2006.gms | milk.gms | shipping.gms

```
obj..    zroad =E= sum(r, cost(r)*segment(r));

hunt..   sum(p, a(p,'hunting')*project(p)) =G= b('hunting');

harvest..   sum(p, a(p,'timber')*project(p)) =G= b('timber');

revenue..   sum(p, a(p,'revenue')*project(p)) - sum(r, cost(r)*segment(r))
                       =G= b('revenue');

totseg(k).. sum(r, segproj(k,r)*segment(r)) =L= 0;

psbal1..  project('A') =L= segment('s8');
psbal2..  project('B') =L= segment('s9');
psbal3..  project('C') =L= segment('s11');
psbal4..  project('D') =L= segment('s10');
psbal5..  project('E') =L= segment('s5');
psbal6..  project('F') =L= segment('s6');
contproj(p).. project(p) =L= 1;

Model road /all/;
option mip = osl;
Solve road using MIP minimizing zroad;
```

1: 52 Modified Insert

Start | C:\Document... | LP_extension... | gamside | 78% | 9:19 AM

Solution

Item	Integer	Non-integer
Objective ($'000s)	335	335
Hunting days	500.0	484.6
Harvest ('000 m^3)	32	30
Net returns ($'000s)	65.00	43.46
Projects	B, C, D	B, 0.846C, D
Road segments	s_1, s_2, s_3, s_7, s_9, s_{10}, s_{11}	s_1, s_2, s_3, s_7, s_9, s_{10}, s_{11}

- Note that when we change the ‘project’ variable from being binary to a positive variable the answer depends on the solver that is used. Using MIP, the BDMLP solver gives a value of 1 for project C, while the OSL solver (from IBM) gives 0.846.
- Now consider the third question:

Road section 1 will clearly have to bear more traffic than other sections, so it is necessary to build it to a higher standard. The carrying and construction costs for three different standards are:

		Standard	
	Low	Medium	High
Carrying capacity (10^3 tons/yr)	40	50	70
Cost (\$$10^3$)	25	50	75

Traffic from each project	Project					
	A	B	C	D	E	F
Traffic (10^3 tons/yr)	8	4	14	13	10	7

Need several additional constraints:

$\sum_j f_j\, p_j \le \sum_n k_n\, s_n$, n = 1Low, 1Med, 1Hi   (traffic load)

$s_{11} + s_2 \le 2\,(s_{1\mathrm{Low}} + s_{1\mathrm{Med}} + s_{1\mathrm{Hi}})$

$s_{1\mathrm{Low}} + s_{1\mathrm{Med}} + s_{1\mathrm{Hi}} = 1$

Plus objective function needs to be modified.

The modified GAMS code is as follows:

gamside: C:\Documents and Settings\kooten.UVIC\My Documents\Classes\Programming\GAMS\Programs\class_examples.g...

File Edit Search Windows Utilities Help

HW#3_2006.gms | HW#3_Part2_2006.gms | milk.gms | shipping.gms

```
* This solves the road building integer program of
* Joseph Buongiorno and J. Keith Gilless (2003, problem 11.8, pp.229-31)

sets r road segments and projects /s1L, s1M, s1H, s2*s11/
     p projects /A, B, C, D, E, F/
     i site uses /hunting, timber, revenue, traffic/
     k additional constraints /1*5/;

Parameter cost(r) cost of road segments in $000s

/ s1L 25
  s1M 50
  s1H 75
  s2  50
  s3  65
  s4  40
  s5  45
  s6  70
  s7  50
  s8  40
  s9  20
  s10 50
  s11 25/ ;

Parameter b(i) RHS requirements /hunting 400, timber 30, revenue 0, traffic 0/;
```

Note 3 choices for s1

1: 52 Modified Insert

Start C:\Doc... LP_ext... gamside Math_P... 69% 9:52 AM

```
Table a(p,i) table of technical coefficients for equations
       hunting  timber  revenue  traffic
    A    200       6       70       8
    B    300       9      130       4
    C    100      13      140      14
    D    100      10      130      13
    E    200       8      110      10
    F    300       3      100       7
;

Table segproj(k,r) table relating road segments to projects
       s1L  s1M s1H s2  s3  s4  s5  s6  s7  s8  s9  s10  s11
    1  -2   -2  -2  1   0   0   0   0   0   0   0   0    1
    2  0     0   0  -2  1   0   0   0   0   0   0   1    0
    3  0     0   0  0   0   -2  1   1   0   0   0   0    0
    4  0     0   0  0   -2  1   0   0   1   0   0   0    0
    5  0     0   0  0   0   0   0   0   -2  1   1   0    0
;

Variables
zroad             road costs
segment(r)        whether to build road segment
project(p)        whether to build project
;
```

This table has been expanded

gamside: C:\Documents and Settings\kooten.UVIC\My Documents\Classes\Programming\GAMS\Programs\class_examples.g...

File Edit Search Windows Utilities Help

HW#3_2006.gms | HW#3_Part2_2006.gms | milk.gms | shipping.gms

```
Binary variables segment, project;
*Positive variables project;

Equations
obj        objective function is minimize road building cost
hunt       minimum hunting days constraint
harvest    minimum timber harvest constraint
revenue    revenue must exceed costs of road building
totseg(k)  segments per project must balance
traffic    traffic equation constraint
Sproject   only one of the A projects can be completed
psbal1     Following equations balance projects and segments
psbal2
psbal3
psbal4
psbal5
psbal6


;

obj..   zroad =E= sum(r, cost(r)*segment(r));

hunt..  sum(p, a(p,'hunting')*project(p)) =G= b('hunting');

harvest..  sum(p, a(p,'timber')*project(p)) =G= b('timber');
```

1: 52 Modified Insert

Start | C:\Doc... | LP_ext... | gamside | Math_P... | 69% | 9:54 AM

gamside: C:\Documents and Settings\kooten.UVIC\My Documents\Classes\Programming\GAMS\Programs\class_examples.g...

File Edit Search Windows Utilities Help

HW#3_2006.gms | HW#3_Part2_2006.gms | milk.gms | shipping.gms

```
hunt..  sum(p, a(p,'hunting')*project(p)) =G= b('hunting');

harvest..  sum(p, a(p,'timber')*project(p)) =G= b('timber');

revenue..  sum(p, a(p,'revenue')*project(p)) - sum(r, cost(r)*segment(r))
                       =G= b('revenue');

totseg(k).. sum(r, segproj(k,r)*segment(r)) =L= 0;

traffic.. sum(p, a(p, 'traffic')*project(p)) =L= 40*segment('s1L')
                      + 50*segment('s1M') + 70*segment('s1H');

Sproject.. segment('s1L') + segment('s1M') + segment('s1H') =E= 1;

psbal1..  project('A') =L= segment('s8');
psbal2..  project('B') =L= segment('s9');
psbal3..  project('C') =L= segment('s11');
psbal4..  project('D') =L= segment('s10');
psbal5..  project('E') =L= segment('s5');
psbal6..  project('F') =L= segment('s6');

Model road /all/;
*option mip = osl;
Solve road using MIP minimizing zroad;
```

Note added 'traffic' and 'Sproject' equations

1: 52 Modified Insert

Start | C:\Doc... | LP_ext... | gamside | Math_P... | 69% | 9:54 AM

	Solution		
Item	Integer	Non-integer	Integer/ heavier s1
Objective ($'000s)	335	335	285
Hunting days	500.0	484.6	500.0
Harvest ('000 m^3)	32	30	32.0
Net returns ($'000s)	65.00	43.46	115.0
Projects	B, C, D	B, 0.846C, D	B, C D
Road segments	s_1, s_2, s_3, s_7, s_9, s_{10}, s_{11}	s_1, s_2, s_3, s_7, s_9, s_{10}, s_{11}	s_{1Low}, s_2, s_3, s_7, s_9, s_{10}, s_{11}

Problems with IP

- Simplex algorithm does not work; most use a ‘branch-and-bound’ algorithm – choose integer values for variables and search along a branch for ‘better’ solutions than those found earlier (e.g., those associated with starting values). If none found, look at other paths. Optimal solution tedious to find
- No Kuhn-Tucker conditions, so there is no guarantee that any solution is optimal
- No dual solution, so there is no reduced cost/ contribution calculation and no shadow prices

Mixed Integer Programming

- Real-world problems feature both continuous and integer variables – hence, we have mixed integer programming (MIP) problems
- Linear versions of such models can be solved, but the scale of models is limited. The best solver for MIP problems in CPLEX
- Nonlinear models with integer variables are extremely tricky to solve and solvers are only now beginning to appear. Hence, heuristics such as ‘tabu search’ are often used instead.

General MIP Formulation

Min $\quad Z = \sum_j v_j x_j + \sum_k f_k Q_k \qquad j, k = 1, ..., n$

s.t. $\quad A x \leq b \qquad m$ constraints

$-d_j x_j + Q_j \leq 0 \qquad$ for all $j=k = 1, ..., n$

$\sum_k Q_k \geq C$

Where

v is variable cost per unit and f is fixed cost per unit

x are continuous variables and Q are integer

C is the desired total capacity made up of the individual Qs

d is the proportion of relevant Q used in production at any time

2. Stochastic Dynamic Programming (SDP)

In deterministic DP, the state variable evolves according to the difference equation:

$$x_{t+1} = g(t, x_t, u_t)$$

controlled by the appropriate choice of u_t

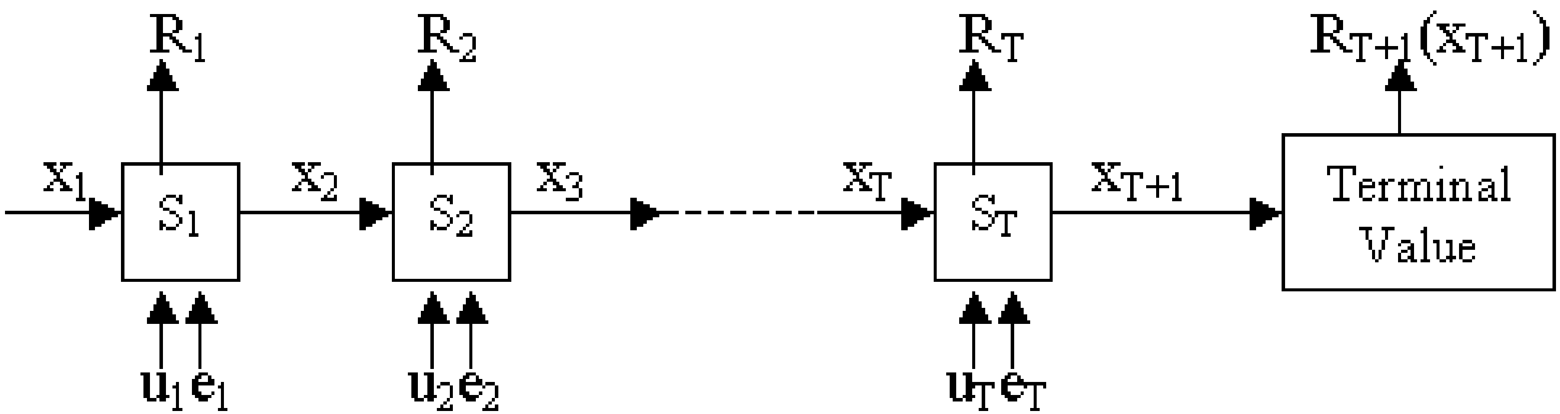
Now let x_t be random/stochastic:

$$x_{t+1} = g(t, x_t, u_t, \varepsilon_t)$$

where ε_t is a random variable.

In practice, this equation is often represented by a transition matrix for each control and **Markov chain programming** is employed.

Flowchart for Stochastic DP System:



e_i= random effect occurring at stage *i*

Motivation

- Example from Buongiorno & Gilless (Chaps 16 & 17; see also Chap 13)

State i	Volume (m^3/ha)
L (low)	<400
M (medium)	400-700
H (high)	>700

Forest Management Example

Transition probability matrix with NO management

	Next state *j*		
Begin state *i*	L	M	H
L	0.40	0.60	0
M	0	0.30	0.70
H	0.05	0.05	0.90

Assumed time step is 20 years

- Assume forest is initially in state L with probability 1. We want to know how it moves over time without management. Where will it end up?

$$p_0 = [1\ 0\ 0]$$

$$p_1 = p_0 P^{NM} = [1\ 0\ 0]\begin{bmatrix} 0.40 & 0.60 & 0 \\ 0 & 0.30 & 0.70 \\ 0.05 & 0.05 & 0.90 \end{bmatrix} = [0.40\ 0.60\ 0]$$

$$p_2 = p_1 P^{NM} = [0.40\ 0.60\ 0]\begin{bmatrix} 0.40 & 0.60 & 0 \\ 0 & 0.30 & 0.70 \\ 0.05 & 0.05 & 0.90 \end{bmatrix} = [0.16\ 0.42\ 0.42]$$

$$p_t = p_{t-1} P^{NM}, \text{ for } t = 1, \ldots, T$$

$$\Rightarrow p^{NM^*} = [\pi_L\ \ \pi_M\ \ \pi_H] = [0.07\ 0.12\ 0.82]$$

- In the long run, the stand of trees will have >700 m^3 timber with probability 0.82 and 400-700 m^3 with probability 0.12.

- How long can one expect the stand to remain in one of the three categories? The **mean residence time** is

$$m_i = SL/(1-p_{ii})$$

where SL is the stage length (20 years) and p_{ii} is the diagonal element on P^{NM}, or probability that a stand in state *i* at the beginning of the period is still in that state at the end of the period.

Mean residence time:

$$m_{\mathrm{L}} = 33.3 \text{ yrs}, \ m_{\mathrm{M}} = 28.6 \text{ yrs}, \ m_{\mathrm{H}} = 200 \text{ yrs}$$

Mean recurrence time is found as:

$$m_{\mathrm{ii}} = SL/\pi_{\mathrm{i}}$$

Recall the values of π_{i} come from p*=[.07 .12 .82], and m_{ii} is the time it takes for a stand in state *i* to return to that same state after exiting it.

$$m_{\mathrm{LL}} = 285.7 \text{ yrs}$$

$$m_{\mathrm{MM}} = 166.7 \text{ yrs}$$

$$m_{\mathrm{HH}} = 24.4 \text{ yrs}$$

Transition probability matrix
With Management

Begin state *i*	Next state *j*		
	L	M	H
L	0.40	0.60	0
M	0	0.30	0.70
H	0.40	0.60	0

Time step is 20 years

- Doing the same thing as before, we find:

	With Management		
State i	Mean residence time (m_i)	Steady-state probability (π_i)	Mean recurrence time (m_{ii})
L	33.3 yrs	0.22	90.9 yrs
M	28.6 yrs	0.46	43.5 yrs
H	20.0 yrs	0.32	62.5 yrs

Calculating long-run returns

- Suppose we have the following immediate return from harvest under management:

State *i*	Total volume (m^3/ha)	Average volume (m^3/ha)	Harvest and after harvest return to state L	
			m^3/ha	$/ha
L	<400	259	0	0
M	400-700	603	344	0
H	>700	817	558	7,254

Recursive Relationship

- Let V_{it} be the value of a stand in state i (= L, M, H) with t periods until the end of the time horizon.
- Let $\beta = 1/(1+r)^{20}$
- Present value of expected returns with t+1 periods to go to the end of the time horizon:

$$V_{i,t+1} = R_i + \beta\,(p_{iL}\,V_{Lt} + p_{iM}\,V_{Mt} + p_{iH}\,V_{Ht})$$

Begin with $V_{Lo} = V_{Mo} = V_{Ho} = 0$

Recursive relation: Stage 1

Assume discount rate of 5%

$$V_{L,1} = R_L + \beta\,(p_{LL}\,V_{L0} + p_{LM}\,V_{M0} + p_{LH}\,V_{H0})$$
$$= 0 + 0.377\,\{.4\,(0) + .6\,(0) + .0\,(0)\} = 0$$

$$V_{M,1} = R_M + \beta\,(p_{ML}\,V_{L0} + p_{MM}\,V_{M0} + p_{MH}\,V_{H0})$$
$$= 0 + 0.377\,\{.0\,(0) + .3\,(0) + .7\,(0)\} = 0$$

$$V_{H,1} = R_H + \beta\,(p_{HL}\,V_{L0} + p_{HM}\,V_{M0} + p_{HH}\,V_{H0})$$
$$= 7254 + 0.377\,\{.4\,(0) + .6\,(0) + .0\,(0)\}$$
$$= 7254$$

Recursive relation: Stage 2

$$V_{L,2} = R_L + \beta\,(p_{LL}\,V_{L1} + p_{LM}\,V_{M1} + p_{LH}\,V_{H1})$$

$$= 0 + 0.377\,\{.4\,(0) + .6\,(0) + .0\,(7254)\} = 0$$

$$V_{M,2} = R_M + \beta\,(p_{ML}\,V_{L1} + p_{MM}\,V_{M1} + p_{MH}\,V_{H1})$$

$$= 0 + 0.377\,\{.0\,(0) + .3\,(0) + .7\,(7254)\}$$

$$= 1914$$

$$V_{H,2} = R_H + \beta\,(p_{HL}\,V_{L1} + p_{HM}\,V_{M1} + p_{HH}\,V_{H1})$$

$$= 7254 + 0.377\,\{.4\,(0) + .6\,(0) + .0\,(7254)\}$$

$$= 7254$$

Recursive relation: Stage 3

$$V_{L,3} = R_L + \beta\,(p_{LL}\,V_{L2} + p_{LM}\,V_{M2} + p_{LH}\,V_{H2})$$
$$= 0 + 0.377\,\{.4\,(0) + .6\,(1914) + .0\,(7254)\}$$
$$= 433$$

$$V_{M,3} = R_M + \beta\,(p_{ML}\,V_{L2} + p_{MM}\,V_{M2} + p_{MH}\,V_{H2})$$
$$= 0 + 0.377\,\{.0\,(0) + .3\,(1914) + .7\,(7254)\}$$
$$= 2130$$

$$V_{H,3} = R_H + \beta\,(p_{HL}\,V_{L2} + p_{HM}\,V_{M2} + p_{HH}\,V_{H2})$$
$$= 7254 + 0.377\,\{.4\,(0) + .6\,(1914) + .0\,(7254)\}$$
$$= 7687$$

Recursive relation: Stage *n*

Since $\beta < 1$, convergence eventually occurs (in this case for $t > 10$). The result is that, for each potential starting state, we find the following value:

$$V_{\mathrm{L,n}} = \$624/\mathrm{ha}$$

$$V_{\mathrm{M,n}} = \$2343/\mathrm{ha}$$

$$V_{\mathrm{H,n}} = \$7878/\mathrm{ha}$$

Long-run expected return is found by multiplying the above values by $[\pi_L\ \pi_M\ \pi_H] = [.22\ .46\ .32]$

Expected return = \$3736/ha

Stochastic DP (cont)

So far we have had no decision to make.

Let $p(i,j,k)$ be the probability that, if system is in state i at time t, it will be in state j at $t+1$ if $u=k$.

Bellman's Equation:

$$V_t(x_t, u_t) = \max_{u_t} E\left[R(x_t, u_t) + \beta V_{t+1}(x_{t+1})\right]$$

$$= \max_{k}\left[ER(i,k) + \beta \sum_{j=1}^{n} p(i,j,k) V_{t+1}(j)\right]$$

$$V_t(i) = \max_{d(t)}\left[ER^d(i) + \beta \sum_{j=1}^{n} p^d(i,j) V_{t+1}(j)\right]$$

Transition probabilities replace state equation, or equation of motion

$$P^{u_1} = \begin{bmatrix} p_{11}^1 & p_{12}^1 & p_{13}^1 \\ p_{21}^1 & p_{22}^1 & p_{23}^1 \\ p_{31}^1 & p_{32}^1 & p_{33}^1 \end{bmatrix} \qquad P^{u_2} = \begin{bmatrix} p_{11}^2 & p_{12}^2 & p_{13}^2 \\ p_{21}^2 & p_{22}^2 & p_{23}^2 \\ p_{31}^2 & p_{32}^2 & p_{33}^2 \end{bmatrix}$$

One transition matrix for each decision

Sum of each row =1.0

Columns are **single-peaked**

Markov Assumption of DP: All the information about the past is contained in the present value of the state variable.

SDP Definitions

- **Policy iteration:** If any state is **reachable** from any other state, then there is convergence toward an optimal policy that holds for any t.
- **Optimal policy:** Optimal decision for any value of state variable at any t.
- **Value iteration:** The optimal policy depends not only on the value of the state variable, but also on t. Some states are not **reachable** from any other state (viz., soil erosion) – there can be an **absorbing state**

Forestry example: Transition matrices

	NO CUT			CUT		
Begin	Next state *j*			Next state *j*		
state *i*	L	M	H	L	M	H
L	0.40	0.60	0.00	0.40	0.60	0.00
M	0.00	0.30	0.70	0.40	0.60	0.00
H	0.05	0.05	0.90	0.40	0.60	0.00

Returns to each decision/state

State	Immediate Return R_{ik} (\$/ha)	
i	NO CUT	CUT
L	0	0
M	0	4,472
H	0	7,254

Recursive Relationship

Present value of expected returns with t+1 periods to go to the end of the time horizon:

$V_{i,t+1}$ =

Max { [R_{iN} + β (p_{iLN} V_{Lt} + p_{iMN} V_{Mt} + p_{iHN} V_{Ht})],

[R_{iC} + β (p_{iLC} V_{Lt} + p_{iMC} V_{Mt} + p_{iHC} V_{Ht})]}

where p_{ijk} is the probability that a stand moves from state *i* to state *j* when the decision is *k* (= N,C)

Recursive Relationship (cont)

Proceed as before, but now keep track of the best decision –

cut (C)

no cut (N)

Again, begin with $V_{\mathrm{Lo}} = V_{\mathrm{Mo}} = V_{\mathrm{Ho}} = 0$

$r = 5\%$ so $\beta \approx 0.377$

Recursive relation: Stage 1

$V_{L,1}$ $= \text{Max}\{\ [R_{LN} + \beta\,(p_{LLN}\,V_{L0} + p_{LMN}\,V_{M0} + p_{LHN}\,V_{H0})],$
$[R_{LC} + \beta\,(p_{LLC}\,V_{L0} + p_{LMC}\,V_{M0} + p_{LHC}\,V_{H0})]\}$

$= \text{Max}\ \{[0 + 0.377(.4\,(0) + .6\,(0) + .0\,(0))],$

$[0 + 0.377(.4\,(0) + .6\,(0) + .0\,(0))]\} = 0\ (\text{N})$

$V_{M,1}$ $= \text{Max}\{\ [R_{MN} + \beta\,(p_{MLN}\,V_{L0} + p_{MMN}\,V_{M0} + p_{MHN}\,V_{H0})],$
$[R_{MC} + \beta\,(p_{MLC}\,V_{L0} + p_{MMC}\,V_{M0} + p_{MHC}\,V_{H0})]\}$

$= \text{Max}\ \{[0 + 0.377(.0\,(0) + .3\,(0) + .7\,(0))],$

$[4472 + 0.377(.4\,(0) + .6\,(0) + .0\,(0))]\} = 4472\ (\text{C})$

Recursive relation: Stage 1 (cont)

$$V_{H,1} = \text{Max}\{ [R_{HN} + \beta\,(p_{HLN}\,V_{L0} + p_{HMN}\,V_{M0} + p_{HHN}\,V_{H0})],$$
$$[R_{HC} + \beta\,(p_{HLC}\,V_{L0} + p_{HMC}\,V_{M0} + p_{HHC}\,V_{H0})]\}$$

$$= \text{Max}\,\{[0 + 0.377(.05\,(0) + .05\,(0) + .9\,(0))],$$
$$[7254 + 0.377(.4\,(0) + .6\,(0) + 0\,(0))]\}$$

$$= 7254\ (C)$$

Decision: [No cut, cut, cut] = [N C C]

Recursive relation: Stage 2

$V_{L,2}$ $= \text{Max}\{\ [R_{LN} + \beta\ (p_{LLN}\ V_{L1} + p_{LMN}\ V_{M1} + p_{LHN}\ V_{H1})],$
$[R_{LC} + \beta\ (p_{LLC}\ V_{L1} + p_{LMC}\ V_{M1} + p_{LHC}\ V_{H1})]\}$

$= \text{Max}\ \{[0 + 0.377(.4\ (0) + .6\ (4472) + .0\ (7254))],$

$[0 + 0.377(.4\ (0) + .6\ (4472) + .0\ (7254))]\} = 1011\ (\text{NC})$

$V_{M,2}$ $= \text{Max}\{\ [R_{MN} + \beta\ (p_{MLN}\ V_{L1} + p_{MMN}\ V_{M1} + p_{MHN}\ V_{H1})],$
$[R_{MC} + \beta\ (p_{MLC}\ V_{L1} + p_{MMC}\ V_{M1} + p_{MHC}\ V_{H1})]\}$

$= \text{Max}\ \{[0 + 0.377(.0\ (0) + .3\ (4472) + .7\ (7254))],$

$[4472 + 0.377(.4\ (0) + .6\ (4472) + .0\ (7254))]\} = 5483\ (\text{C})$

Recursive relation: Stage 2 (cont)

$$V_{H,2} = \text{Max}\{ [R_{HN} + \beta\,(p_{HLN}\,V_{L1} + p_{HMN}\,V_{M1} + p_{HHN}\,V_{H1})],$$
$$[R_{HC} + \beta\,(p_{HLC}\,V_{L1} + p_{HMC}\,V_{M1} + p_{HHC}\,V_{H1})]\}$$

$$= \text{Max}\,\{[0 + 0.377(.05\,(0) + .05\,(4472) + .9\,(7254))],$$
$$[7254 + 0.377(.4\,(0) + .6\,(4472) + 0\,(7254))]\}$$

$$= 8265\ (C)$$

Decision: [N C C]

Long-run solution

- After 10 iterations, the algorithm converges on an equilibrium solution given below:

State i	Decision	Net present value ($/ha)	Long-run probability (π_i)
L	No cut	1,623	0.40
M	Cut	6,095	0.60
H	Cut	8,877	0.00

Calculation of long-run probabilities is illustrated below

How do we find the long-run probability vector and expected returns?

- Create a new transition matrix by taking, for each decision, the row out of the matrix associated with that decision.
- Example: Suppose the transition matrices are ‘reachable’ and the optimal policy is u_1 if in state 2 and u_2 when in state 1 or 3. Then:

$$P = \begin{bmatrix} p_{11}^2 & p_{12}^2 & p_{13}^2 \\ p_{21}^1 & p_{22}^1 & p_{23}^1 \\ p_{31}^2 & p_{32}^2 & p_{33}^2 \end{bmatrix} = \begin{bmatrix} p^{u_2}[1] \\ p^{u_1}[2] \\ p^{u_2}[3] \end{bmatrix}$$

Let $\pi = [\pi_1\ \pi_2\ \pi_3]$ be the probabilities of being in states 1, 2 and 3 in the long run. We can solve π as was done earlier, or by solving $\pi = \pi P^n$, which is the same as finding:

$$\pi = \lim_{n \to \infty} P^n = \begin{bmatrix} \pi \\ \pi \\ \pi \end{bmatrix} = \begin{bmatrix} [\pi_1 & \pi_2 & \pi_3] \\ [\pi_1 & \pi_2 & \pi_3] \\ [\pi_1 & \pi_2 & \pi_3] \end{bmatrix}$$

Note: Each row is the same

Problem: Since probabilities have the property that

$$0 \le \text{prob} \le 1,$$

as $n \to \infty$, P collapses to a null matrix. Another approach is needed

We can find π as follows:

Let

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad D = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Then $\Pi = \begin{bmatrix} \pi_1 & \pi_2 & \pi_3 \\ \pi_1 & \pi_2 & \pi_3 \\ \pi_1 & \pi_2 & \pi_3 \end{bmatrix} = \begin{bmatrix} \pi \\ \pi \\ \pi \end{bmatrix} = D\,(I + D - P)^{-1}$

and: Expected returns = ER = $[\pi_1\ \pi_2\ \pi_3] \times \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix}$

where R_i refers to the returns to state i under the optimal policy regime.

- In the previous cut/no harvest timber management example, the decision rule is **no cut** whenever in state L, and **cut** in states M and H
- Taking the ‘L row’ from the ‘no cut’ matrix and the ‘M’ and ‘H’ rows from the ‘cut’ matrix gives:

$$P = \begin{bmatrix} 0.4 & 0.6 & 0 \\ 0.4 & 0.6 & 0 \\ 0.4 & 0.6 & 0 \end{bmatrix}$$

$$I+D-P = \begin{matrix} 0.6 & -0.6 & 1.0 \\ -0.4 & 0.4 & 1.0 \\ -0.4 & -0.6 & 2.0 \end{matrix}$$

$$Inv(I+D-P) = \begin{matrix} 1.4 & 0.6 & -1.0 \\ 0.4 & 1.6 & -1.0 \\ 0.4 & 0.6 & 0 \end{matrix}$$

$$D*Inv(I+D-P) = \begin{matrix} 0.4 & 0.6 & 0 \\ 0.4 & 0.6 & 0 \\ 0.4 & 0.6 & 0 \end{matrix}$$

Summary

- From the SDP algorithm (previous table on slide 57), the long-run expected returns (ER) for each state:

  ER if initially in L: $1623/ha

  ER if initially in M: $6095/ha

  ER if initially in H: $8877/ha

- The long-run expected return is found by multiplying the π vector by the returns vector, which gives $4306.20/ha.
- Note that you never let trees grow to reach state H since they are harvested in state M.

3. Mathematical Programming Models: Validation and Calibration

- Norton & Hazell suggest tests for validating a sectoral model:
  - Capacity test for over-constrained models
  - Marginal cost test: Marginal costs of production plus implicit opportunity costs of fixed inputs must equal output price
  - Comparison of dual values with actual rental values (e.g. land)
  - Three additional validations look at input use, production levels and product prices.
- Problem: Must ensure that the number of binding constraints in the optimal solution are less than the number of non-zero activities (variables, controls) observed in the base solution.
- If there is enough data to specify a constraint set that reproduces the optimal, base-level solution, no additional calibration is required.

Calibration

- Two broad approaches to reducing specification error in optimization models
  - Demand Based: Employ a range of methods to add risk or endogenize prices, thus modifying the objective function, but substantial calibration issues usually remain.
  - Constrain activities by flexibility constraints, or step functions, over multiple activities. Contracts and/or quotas (e.g., that limit output from a power generating source), rotational requirements (in crop production), artificial but sensible constraints (e.g., no timber harvest in first 40 years after planting), other input limitations, etc can be used
- Nonlinear calibration is best, but risk alone provides insufficient calibration terms to calibrate a model

Why PMP?

- Consider land allocated to crops in a certain region. Our objective is to model the underlying decision process that leads to this allocation.
- Suppose we observe 8 different crops being grown.
- For each crop, we have information on yields per ha, crop prices, average costs of production, etc.
- If we impose all reasonable constraints, an LP will cause us to choose perhaps 2 or 3 crops, even though we observe eight; a nonlinear program might lead us to choose 3-4, but not all eight. What's going on?

Why PMP? (cont)

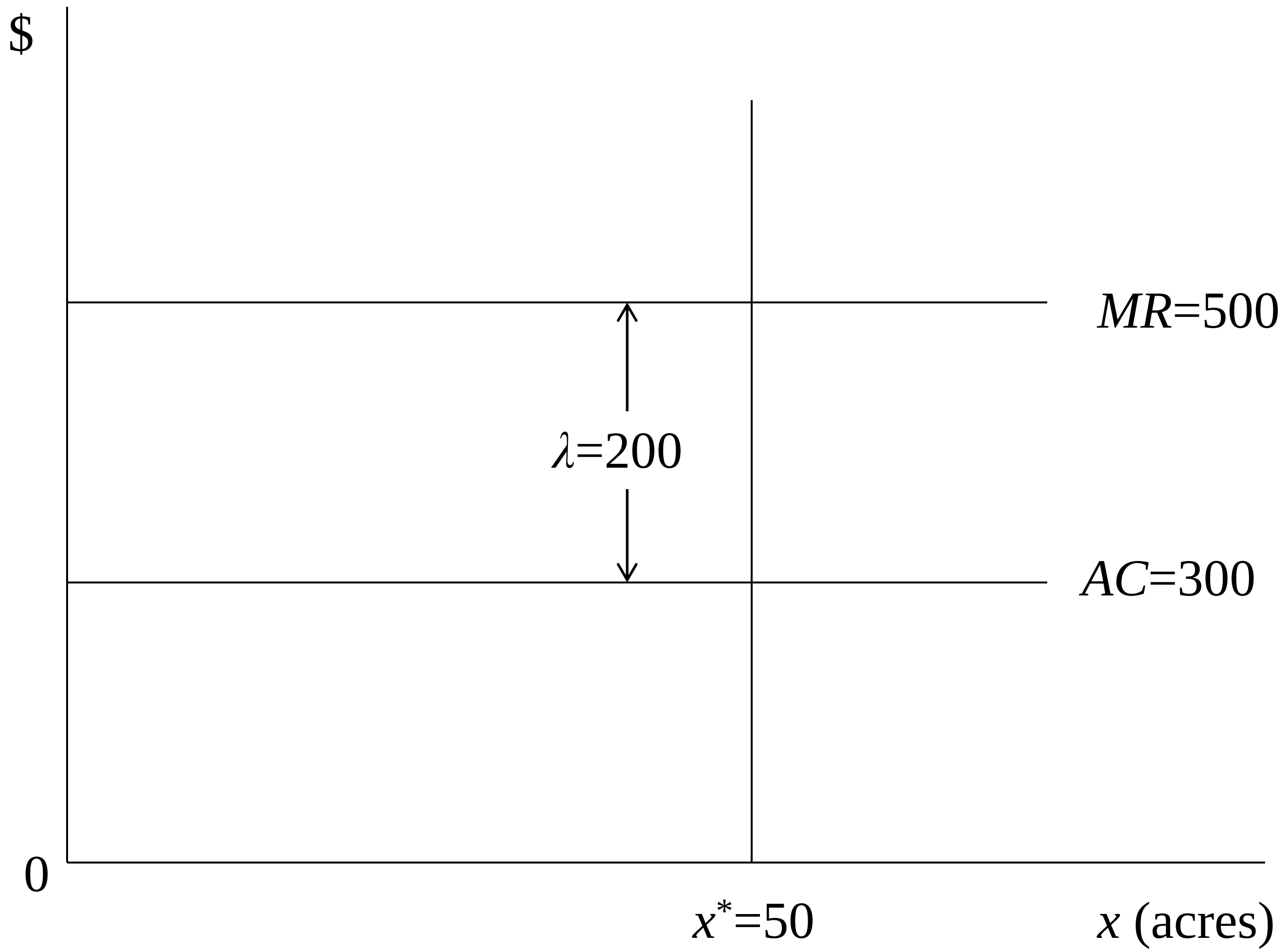
- To get the model to plant the requisite eight crops, it is necessary to add arbitrary (flexibility) constraints.
  - Calibration for the base year is possible, but there is no theoretical or other basis for the calibration, so one cannot be sure that scenario analysis is at all accurate.
  - Such calibration is arbitrary, not always useful as a policy guide, and not a preferred approach
- PMP gets around this problem by using a calibration that takes into account the reasons for planting multiple crops, namely, risk, interactions among crops (to reduce pests, say), unobserved costs, between year interactions (benefits or costs), etc.

Cost-based approach to PMP calibration

- **Key Observation**: every linear constraint in an optimization problem can just as well be represented by a nonlinear cost function with appropriately chosen coefficients.
- Illustrate using a single crop cost-based PMP calibration
  – A single linear crop production activity is measured by the acres *x* allocated to the crop. The yield is assumed constant. The data available to the modeler are:
    - Marginal revenue/acre is constant at $500/acre
    - Average Cost = $300/acre
    - Observed acres allocated to the crop = 50 acres (unconstrained optimum)

<u>Key point</u>: We have data on average not marginal cost!

PMP Calibration: Single-crop Example



PMP Calibration: Single-crop Example

A measure of the value of the residual cost needed to calibrate the crop acreage to 50 (by setting marginal revenues equal to marginal cost at that acreage) is obtained from:

$$\text{Maximize} \quad \pi(x) = 500\,x - 300\,x$$

$$\text{subject to} \quad x \leq 100$$

Solution is clearly x=100, but we observe x=50

The calibration proceeds in five steps:

- **Step I:**

Nonlinear calibration proposition: if we need calibration constraints, the cost and/or the production function is nonlinear. Define the total cost function to be quadratic in acres (x), as this is the simplest general form.

$$TC = \alpha\,x + \tfrac{1}{2}\,\gamma\,x^2$$

Other functional forms are possible: Cobb-Douglas, CES

- **Step II:**
  - Under unconstrained optimization crop acreage expands until the marginal cost equals marginal revenue.
  - Clearly, $MC = MR$ at $x = 50$, not $x = 100$ (we observe $x = 50$).
- **Step III:**
  - It follows that the value λ in the linear model is the difference at the constrained calibration value and is equal to $MR – AC$.
  - But (from Step II) $MR = MC \rightarrow \lambda = MC – AC$ (since $MR = MC$ at $x = 50$).
  - Assuming a quadratic total cost function TC, then:

$$MC = \alpha + \gamma x \qquad \text{and} \qquad AC = \alpha + \tfrac{1}{2} \gamma x$$

$$\rightarrow MC - AC = \alpha + \gamma x - (\alpha + \tfrac{1}{2} \gamma x)$$

$$\rightarrow \lambda = MC - AC = \tfrac{1}{2} \gamma x$$

and the cost slope coefficient is calculated as:

$$\gamma = 2\lambda/x^* = (2 \times 200)/50 = 8$$

- **Step IV:**
  - Now calculate the value of the cost function intercept α using the *AC* information (*AC*=300 in the basic data set):

    $300 = \alpha + (½ \times 8 \times 50) \rightarrow \alpha = 300 - 200 = 100$
- **Step V:**
  - Using the values for α and γ, the unconstrained quadratic cost problem is:

    $\text{Max } \pi(x) = 500x - \alpha\, x - ½\, \gamma\, x^2 = 500\, x - 100\, x - ½\, 8\, x^2$

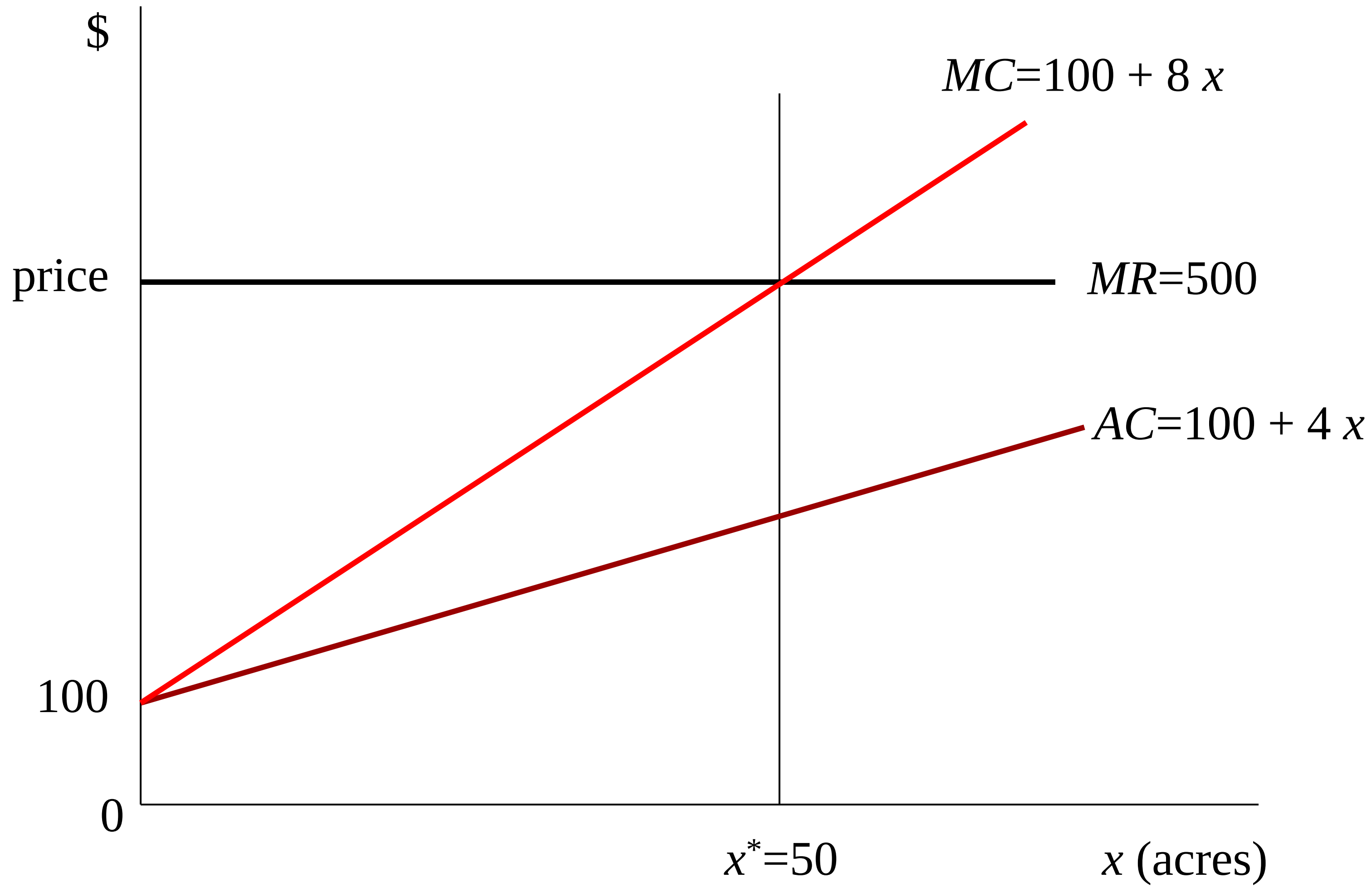
    Therefore, $\partial\pi(x)/\partial x = 500 - 100 - 8\, x$

  Setting $\partial\pi(x)/\partial x = 0 \rightarrow MR = MC$ and

  $8\, x = 400 \rightarrow x^* = 50$

We have thus calibrated the model.

PMP Calibrated Model



Notes

1. The unconstrained model now calibrates exactly in x and in π.
2. $MC = MR$ at $x = 50$.
3. $AC = 300$ at $x = 50$.
4. The cost function has been “tilted".
5. Two types of information are used: observed x^* and AC.
6. The observed x^* quantities need to be mapped into dual value space λ by the calibration constrained LP before it can be used.
7. The model now reflects the ‘preferences’, or the opportunity costs, of the decision maker (landowner or farmer).
8. The model is unconstrained by calibration constraints for policy/scenario analysis.

Three Stages to PMP Calibration

1. Constrained LP model is used to generate dual values for both the resource (λ_1) and calibration (λ_2) constraints. (Resource constraints are the usual technical constraints.)
2. The calibrating constraints dual values (λ_2) are used, along with the data-based average cost function, to derive unique calibrating cost function parameters (α_i and γ_i).
3. The cost parameters are used with the base-year data to specify the PMP model:

$$\text{Max } \Sigma_i \, [p_i \, y_i \, x_i - (\alpha_i + \tfrac{1}{2} \gamma_i \, x_i) \, x_i]$$

$$\text{subject to } A\,x \le b, \quad x \ge 0.$$

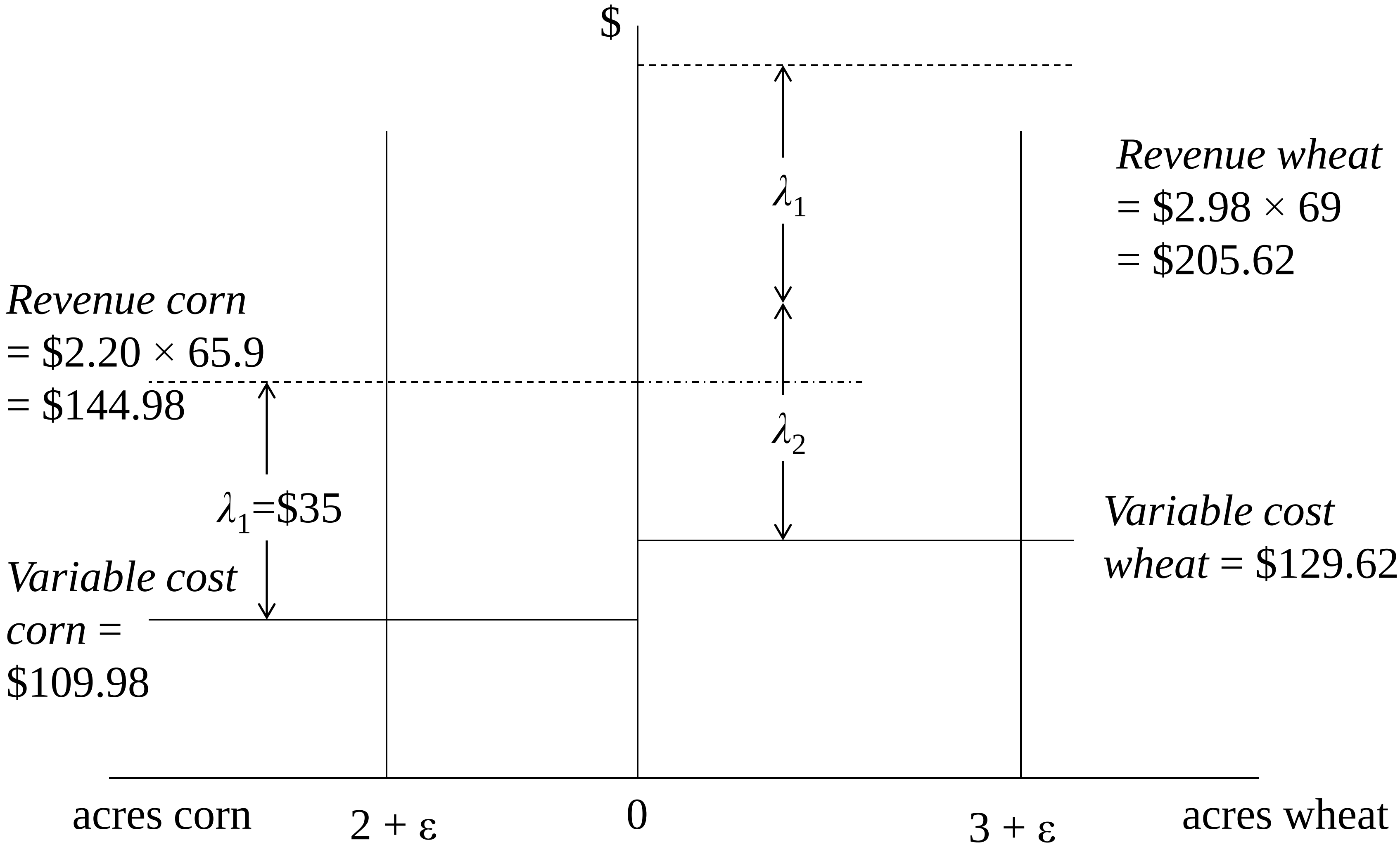
The resulting model calibrates exactly to the base solution and original constraint structure.

Two Crop Example

Item	WHEAT	CORN
Crop prices ($/bu)	$2.98	$2.20
Variable cost ($/acre)	$129.62	$109.98
Average yield (bu/acre)	69.0 bu	65.9 bu
Gross margin ($/acre)	$76.00	$35.00
Observed allocation (acres) (Total acres = 5)	3 ac	2 ac

Source: Howitt (2005, Chapter 5)

PMP Calibration: Two-crop Example



Mathematical Representation of Problem

Max $(\$2.98 \times 69 - \$129.62)x_W + (\$2.20 \times 65.9 - \$109.98)x_C$

s.t. (1) $x_W + x_C \leq 5$

(2) $x_W \leq 3.01$

(3) $x_C \leq 2.01$

$x_W, x_c \geq 0$

Recall the gross margins:

Wheat = \$76/ac

Corn = \$35/ac

Solving in Excel gives:

$x_W = 3.01$, $x_C = 1.99$; $\lambda_1 = 35$, $\lambda_2 = [41\ 0]$

NOTE: If you do not have the $\varepsilon = 0.01$ in constraints (2) and (3), then constraint (1) would be redundant!

- The dual value on land $\lambda_1 = 35$: dual values on the calibration constraint set $\lambda_2 = [41\ 0]$.
- Now solve the previous LP without the calibration constraints (2) and (3). Then $x_W=5$, $x_C=0$, the reduced gradient for corn is -41, and $\lambda=76$.
- Recall the total cost function for land:

$$\mathrm{TC} = (\alpha_k + \tfrac{1}{2}\,\gamma_k\, x_k)\, x_k,$$

where k refers to an activity related to the calibration constraints.

Then,

$$MC = \alpha_k + \gamma_k\, x_k \text{ and } AC = \alpha_k + \tfrac{1}{2}\,\gamma_k\, x_k$$

Now

$$MC_k - AC_k = f\,'(x^o{}_k) - f(x^o{}_k)/x^o{}_k = \lambda_{2k}$$

Then

$$(\alpha_k + \gamma_k\, x_k) - (\alpha_k + ½\, \gamma_k\, x_k) = \lambda_{2k}$$

Solving gives:

$$\gamma_k = 2\times\lambda_{2k}/x^o{}_k$$

Thus, for land in wheat

$$\gamma_W = (2 \times 41)/3 = 27.333$$

Recall: $AC_k = \alpha_k + ½\, \gamma_k\, x_k$

Then

$$\alpha_k = AC_k - ½\, \gamma_k\, x_k$$

Substituting the value of average variable cost per acre for wheat and the value of γ_W gives:

$$\alpha_W = 129.62 - (½ \times 27.333 \times 3) = 88.62$$

Using the cost function parameters, the primal PMP problem becomes:

$$\begin{aligned}
\text{Max}\quad & [(\$2.98\times69)x_W + (\$2.20\times65.9)x_C \\
& - (88.62 + ½\times27.333x_w)x_w - 109.98\, x_c] \\
\text{s.t.}\quad & x_W + x_C \le 5 \\
& x_W, x_c \ge 0
\end{aligned}$$

Calibrated Model Results

- Calibrated model results are provided on the next slide for the Excel solver. It leads to the choice of 3 acres in wheat and 2 acres in corn (as observed).
- We can also do some quick calculations:
  - $MC_{(w=3)} = \alpha_w + \gamma_w x_w = 88.62 + (27.333 \times 3) = 170.619$
  - $VMP_{(w=3)} = p_w y_w - MC = (2.98 \times 69) - 170.619 = 35.001$
  - $VMP_C = p_c y_c - MC = (2.20 \times 65.9) - 109.98 = 35$
  - The calibration is pretty well exact!!

NOTE: The diagram is provided after the Excel results.

D3 fx =117*B3+35*B4-13.6665*B3*B3

Wheat-Corn Land Allocation Calibrated using Positive Mathematical Programming

wheat	3.000036578	objective	298.0015		
corn	1.999963422	constraint	5	<=	5

Cell	Name	Original Value	Final Value
D1	objective	0	298.0015

Cell	Name	Original Value	Final Value
B1	wheat	0	3.000036578
B2	corn	0	1.999963422

Cell	Name	Cell Value	Formula	Status	Slack
D2	constraint	5	D2<=F2	Binding	0
B1	wheat	3.000036578	B1>=0	Not Binding	3.000037
B2	corn	1.999963422	B2>=0	Not Binding	1.999963

Adjustable Cells

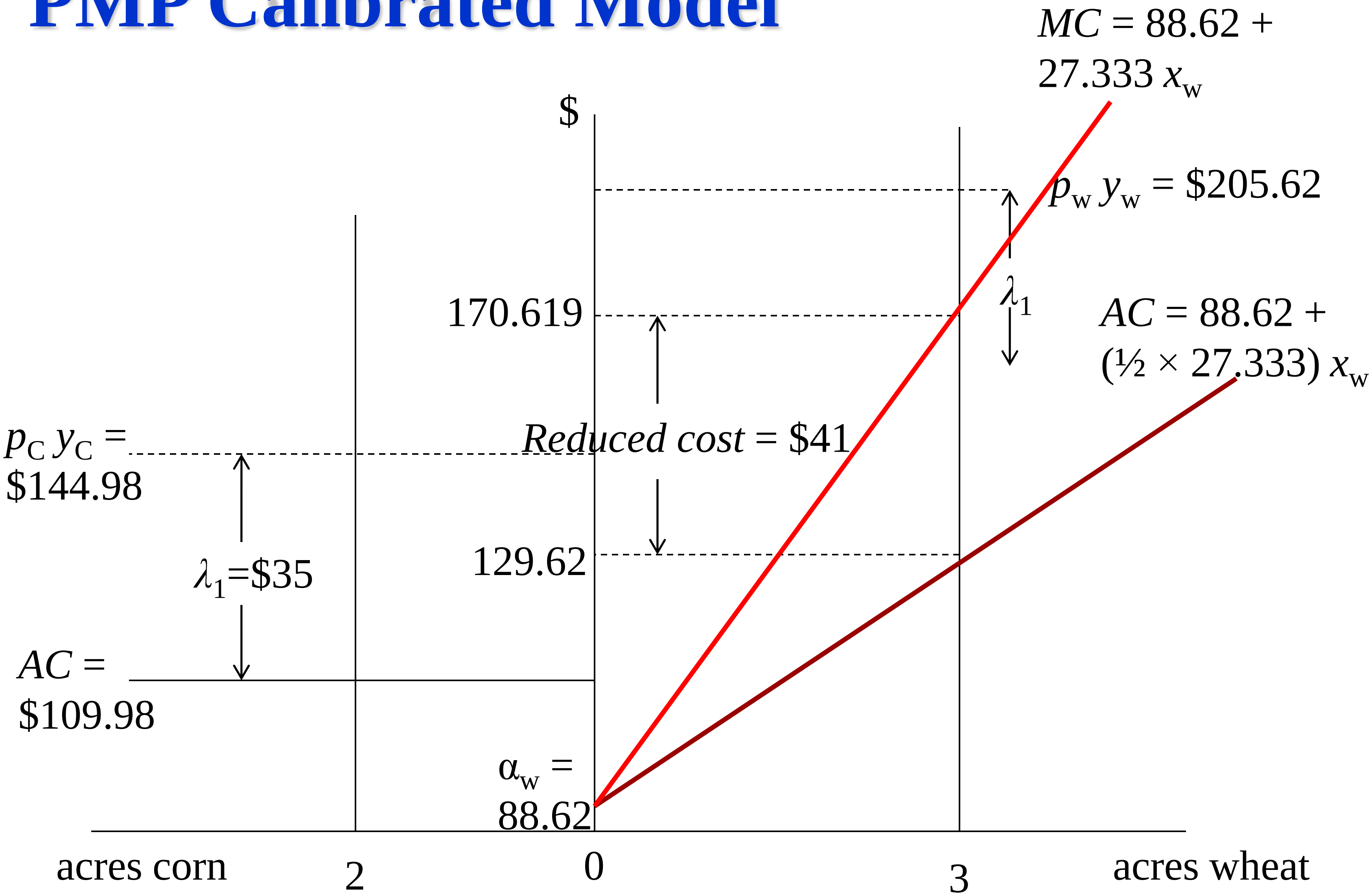
Cell	Name	Final Value	Reduced Gradient
B1	wheat	3.000036578	0
B2	corn	1.999963422	0

Constraints

Cell	Name	Final Value	Lagrange Multiplier
D2	constraint	5	34.99995804

Answer Report 1 / Sensitivity Report 1 / Limits Report 1 / Sheet1 / Sheet2 / Sheet3

PMP Calibrated Model



Notice the model is calibrated for one PMP activity and one LP activity, and the constraint on wheat still prevents an optimal

Broadening the Calibration

- Recall that, when we solved the wheat-corn problem with both resource and calibration constraints, we found: λ_1=35, λ_2=[41 0]; that is, $\lambda_{2C} = 0$
- The model was calibrated for one PMP activity, wheat, as it was more profitable than corn and the calibration constraint was binding.
- Additional information on the MC function is needed for crops with no binding calibration constraint – these are then marginal crops (x_m). When λ_2 is zero for these crops, you cannot distinguish between AC and MC (and need outside information)

Elasticity of supply

- Define elasticity of supply:

$$\eta_s = (\partial q/\partial p)\ (p/q)\text{, where } p \text{ is price}$$

- For our application:

$$\eta_s = (\partial x/\partial MC)\ (p/x^o)\text{, where } x^o \text{ is observed}$$

- Given that $\partial MC/\partial x = \gamma$, we find from calibration:

(1) $\gamma = p/(\eta_s \times x^o)$

(2) $\eta_s = p/\gamma\ x^o$

- Equation (1) can be used to determine γ if outside information (say from econometric studies) is available on η_s
- Equation (2) can be used to find η_s if enough information is available (namely on λ_2) to find the cost function and γ using the PMP method.
- Given γ and λ_2, we can find the intercept of MC as:

$$\alpha_k = AC_k + \lambda_{2k} - \gamma_k\ x^o{}_k$$

- Returning to the wheat-corn example, clearly we need outside information on corn to be able to find its MC
- Define an adjustment at x^o that is added to the LP average cost to obtain a nonlinear cost function:

  $$Adj = MC - AC = \tfrac{1}{2}\,\gamma\, x^o = p/2\eta_s$$

  The *Adj* value is the PMP term for the marginal activities, but, since *Adj* increases the marginal opportunity of the binding resources, it also changes all the non-marginal PMP values.

  $$\hat{\lambda}_{2i} = \lambda_{2i} + \mathrm{Adj}$$

How then do we proceed? We do not know beforehand which activities are in the x_m and x_k groups upon solving the stage I problem (with the resource and calibration constraints in place).

- Assume prior information: $\eta_{sc} = 2.25$
- Using the equation at the top of the previous slide for Adj_{m}, the adjustment term for corn is:

$$Adj_{\mathrm{m}} = \hat{\lambda}_{2C} = p/2\eta_{s} = (2.20 \times 65.9)/(2\times2.25) = 28.996$$

  Note that we use total revenue per acre as the price because the adjustment factor is per acre.
- It is also necessary to adjust the dual value of wheat:

$$\hat{\lambda}_{2W} = \lambda_{2W} + Adj = 41.0 + 28.996 = 69.996$$

- The new λ values are used for both wheat and corn to determine the nonlinear cost functions for the final PMP problem:

$$\text{Max} \quad (2.98\times 69)-[59.624+(0.5\times 46.664)x_W]x_W$$
$$+(2.20\times 65.9)-[80.984+(0.5\times 28.996)x_C]x_C$$

subject to:

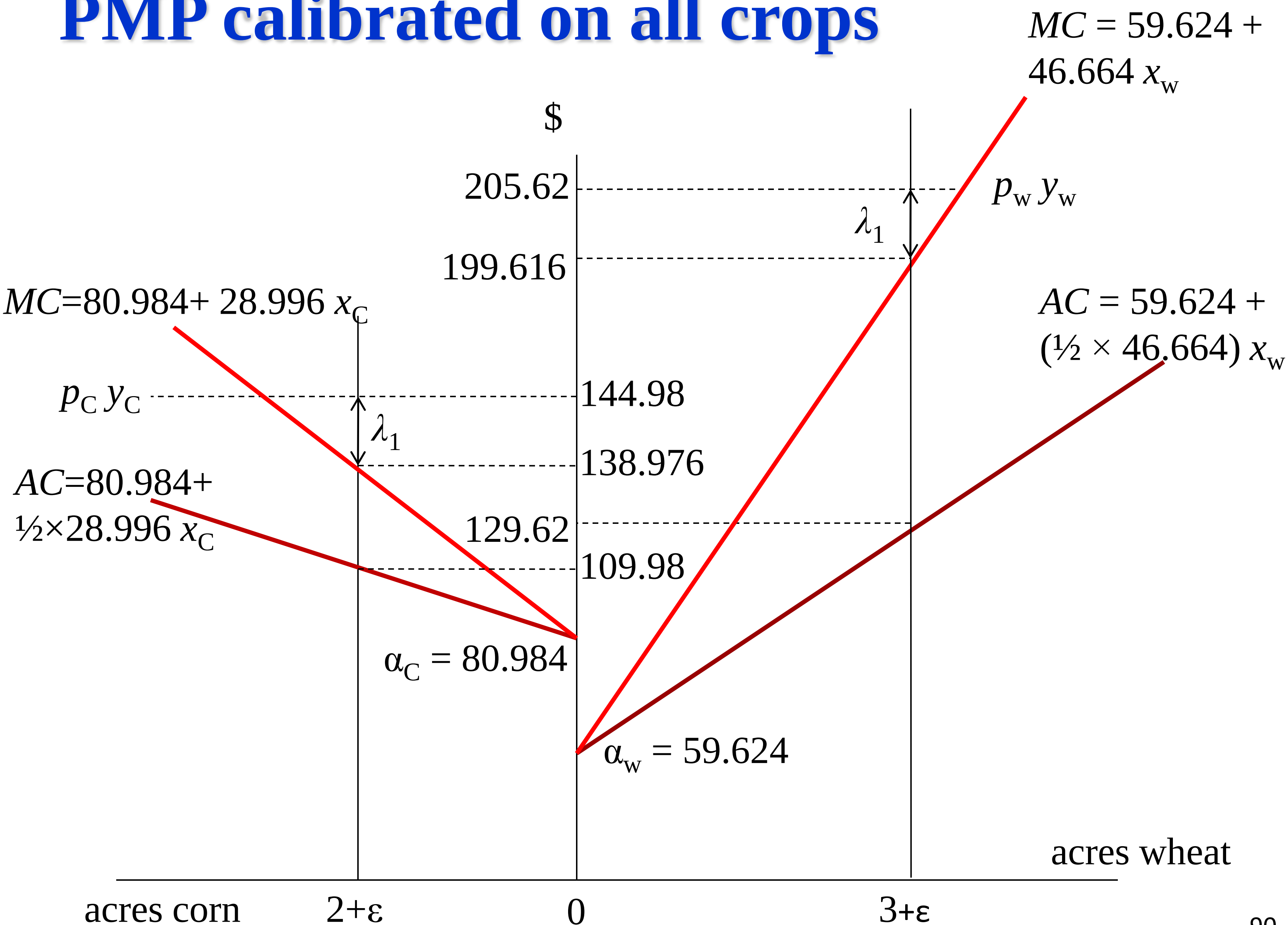
$$x_W + x_C \le 5$$
$$x_W, x_C \ge 0$$

The value of the marginal product of each crop is:

$$VMP_{W|x^{o}_{w=3}} = 2.98\times 69-(59.624+46.664\times 3) = 6.004$$
$$VMP_{C|x^{o}_{C=2}} = 2.20\times 65.9-(80.984+28.996\times 2) = 6.004$$

Note that the dual value of land = 6.004 (= 35 – 28.996)

PMP calibrated on all crops



Example

- Land use example of using PMP to calibrate land uses to existing ones is found here

APPENDIX
Some PMP Theory

- The general LP problem is:

$$\text{Max}_x \quad c'x$$
$$\text{subject to} \quad Ax \leq b,\ x \geq 0$$

- <u>Problem</u>: When we solve this problem, we find that x_m enter the final basis, while x_k do not (even though x_k are observed in practice). Hence, calibration is needed.
- Modify the general LP as follows:

$$\text{Max}_x \quad c'x$$
$$\text{subject to} \quad Ax \leq b,\ Ix \leq x^o + \varepsilon,\ \ x \geq 0$$

where x^o refers to observed activities and ε is added to the calibration constraints to prevent degeneracy. (I is identity matrix.)

The optimal basic solution to the above problem can be written as:

$$Max\ c'_m\ x_m + c'_k\ x_k$$

$s.t.$ $\hat{A}x = \hat{b}$, which is partitioned as

$$\begin{bmatrix} \mathrm{B} & N \\ 0 & I \end{bmatrix} \begin{bmatrix} x_m \\ x_k \end{bmatrix} = \begin{bmatrix} b \\ x_k^o + \varepsilon \end{bmatrix}$$

The optimal dual constraints for this problem are

$\hat{A}'\lambda = c$, which in partioned form is as follows:

$$\begin{bmatrix} \mathrm{B}' & 0 \\ N' & I \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} c_m \\ c_k \end{bmatrix}$$, which in turn can be solved as

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} \mathrm{B}'^{-1} & 0 \\ -N'\mathrm{B}'^{-1} & I \end{bmatrix} \begin{bmatrix} c_m \\ c_k \end{bmatrix}$$

The $k \times 1$ vector of dual values for the binding calibration constraints (λ_2) has the value

$$\lambda_2 = c_k - N' \, B'^{\,-1} \, c_m$$

The RHS of this equation is the difference between the gross margin of the calibrating activity c_k and the equivalent gross margin obtained from the less profitable marginal cropping activities c_m.

Explanation: The x_m activities are in the solution basis when there are no calibration constraints. Yet, the x_k activities are observed in practice, so they must be the more profitable activities – the x_m cropping activities must be less profitable.

This implies that λ_2 is the marginal opportunity cost of restricting the calibrated activities by the amount needed to bring the marginal x_m activities into the expanded basis. This cost of restricting the more profitable activities x_k in the basis is similar to the familiar reduced cost term.

Two things:

(1) When land is the numeraire (Leontief production needs a common unit of measurement), the corresponding coefficients in N and B are one.

(2) $\lambda_2 \geq 0$ as a marginal increase in the RHS upper bound on the more profitable activities (x_k) will increase the value of the objective function.

Dual values associated with the binding calibration constraints (λ_2) are independent of the resource and technology dual values (λ_2).

Upon calibration, an increasing nonlinear cost function, $f(x_k)$ is added to the objective function, so MC and AC of producing x_k differ. Net return to land from x_k now decreases as acreage increases, until an internal equilibrium is reached where they equal the opportunity cost of land set by the marginal crops x_m. This is the "equimarginal" principle of optimal input allocation.

If calibration constraints are removed and a nonlinear cost function for x_k is added, the mathematical program becomes:

$$\text{Max } c'_m x_m + c'_k x_k - f(x_k)$$

$$s.t. \quad \begin{bmatrix} B & N \end{bmatrix} \begin{bmatrix} x_m \\ x_k \end{bmatrix} = b$$

The reduced gradient for x_k is found by rewriting the constraints as

$$x_m = B^{-1}b - B^{-1}Nx_k$$

Substituting this back into the objective function defines the problem as an unconstrained function of x_k

$$\text{Max } c'_m (B^{-1}b - B^{-1}Nx_k) + c'_k x_k - f(x_k)$$

The unconstrained gradient is defined as the reduced gradient by taking derivatives and transposing:

$$c_k - N'B'^{-1}c_m - \nabla f'(x_k) = 0,$$

where $\nabla f'(x_k)$ is the gradient of $f(x_k)$ and a row vector. Finally,

$$c_k - N'B'^{-1}c_m = \nabla f'(x_k) = \lambda_2$$

The optimal solution to the calibrated problem on the previous slide responds to changes in the linear gross margin (c), the RHS values (b), and/or the constraint coefficients (B or N). Note that

$$\lambda_2 = (p_k y_k - AC_k) - (p_m y_m - AC_m),$$

or the difference in gross margins per unit of land for x_k and x_m.

Since the gross margin for x_m is equal to the opportunity cost of land, and since the land coefficients in N and B equal one, we have:

$$\lambda_2 = (p_k y_k - AC_k) - \lambda_1 \text{ or } \lambda_2 + \lambda_1 = (p_k y_k - AC_k)$$

At the optimal allocation of land, the equimarginal principle requires all crops to have marginal net return equal to the opportunity cost of land, so:

$$\lambda_1 = p_k y_k - MC_k = p_k y_k - AC_k - (MC_k - AC_k)$$

Substituting λ_1 into the previous equation yields:

$$\lambda_2 = MC_k - AC_k$$

Formal proofs are found in:

Howitt (AJAE, 1995; Howitt 2005, Chapter 5)