# Price Endogenous Programming

## Finding an Equilibrium With Math Programming

Suppose we have the LP

Max PX

s.t. X - Y < 0

 Y < 100

 X , Y > 0

And now we have demand curve

So we sub in

Max (10 – 0.5\*X) X – 2Y

s.t. X - Y < 0

 Y < 100

 X , Y > 0

# Price Endogenous Programming

## Finding an Equilibrium With Math Programming

Now using Karush-Kuhn –Tucker

Max (10 – 0.5\*X) X – 2Y

s.t. X - Y < 0

 Y < 100

 X , Y > 0

Plus non neg and complementary slackness

Solution

X=8

Y=8

And P=6

So supply cost =2 and demand price = 6.

 Price Endogenous Programming

Finding an Equilibrium With Math Programming

So this does not set P=MC

This is the Wrong way to find an equilibrium as the demand price is 6 and the supply price 2 with the gap in between being monopoly rent

Solution

X=8

Y=8

But P=6



So marginal supply cost =2 and P=demand price = 6.

Monopoly

Price Endogenous Programming

Finding an Equilibrium With Math Programming

So does not set P=MC



But what if we solve

Max

s.t. X - Y < 0

 Y < 100

 X , Y > 0

Then it works out to x=y=16 and p=MC=2

# Price Endogenous Programming

## Finding an Equilibrium With Math Programming

Basics:

Let the inverse demand equation be



And the inverse supply equation



Equilibrium has price and quantity equated



Can be achieved by solving



Sector Modeling

 Finding an Equilibrium with Math Prog



Lagrangian





Simultaneous solution of the first two conditions yields



and the third equation requires that

Qd = Qs

So solving the optimization problem generates a solution which is an equilibrium

# Price Endogenous Programming

## Finding an Equilibrium With Math Programming

One should recognize possible peculiarities

Markets could clear at zero with supply price > demand price.

Thus, market price (P\*) > the demand price



and less than the supply price,



These should only be inequalities when quantity supplied or demanded equals zero.



Quantity supplied must be > the quantity demanded



but if quantity supplied > quantity demanded,

then P\* = 0



Finally, price and quantities > 0



These look like Kuhn-Tucker conditions

# Price Endogenous Programming

Market Peculiarities

Normal Market



Ps=Pd Qs=Qd

Air Space Shuttle or

 Or Bumper Crop Wheat to Canada

#

# Ps=Pd=0 Qs>Qd Ps>Pd Qs=Qd=0 Price Endogenous Programming

## Finding an Equilibrium With Math Programming

The Karush-Kuhn Tucker Problem:

KKT conditions are

Which if you replace with P and break complementary slackness into 2 parts are same as above

# Price Endogenous Programming

## Finding an Equilibrium With Math Programming

The general form maximizes the integral of the area underneath the demand curve minus the integral underneath the supply curve, subject to a supply‑demand balance.



## Price Endogenous Programming

# Finding an Equilibrium With Math Programming

Practical interpretation of the shadow price

Consider what happens if the Qd - Qs < 0 constraint is altered to Qd - Qs < 1.



# Price Endogenous Programming

## Finding an Equilibrium With Math Programming

Example

Suppose we have



Then the formulation is



|  |
| --- |
| Table 13.2. Solution to Simple Price Endogenous Model |
| Variables | Level | Reduced Cost | Equation | Slack | Shadow Price |
| Qd | 10 | 0 | Objective function | 0 | -1 |
| Qs | 10 | 0 | Commodity Balance | 0 | 3 |

# Price Endogenous Programming

## Takayama and Judge Spatial Equilibrium Model

Common programming model involves spatial equilibrium

An extension of the transportation problem

Assumes production and/or consumption occurs in spatially separated regions which each have supply and demand relations.

Suppose that in region i demand for good is given by:



Supply function for region i is



"Quasi-welfare function" for each region



Total welfare function



Demand Balance



Supply Balance



# Price Endogenous Programming

## Takayama and Judge Spatial Equilibrium Model

Resultant Problem:



Kuhn Tucker conditions related to the problem variables are



# Price Endogenous Programming

##  Spatial Equilibrium Model – Example

Example

 (US, Europe, Japan)

Supply:



Demand:



The formulation of this problem is

 -

 

# Price Endogenous Programming

##  Spatial Equilibrium Model – Example Solution

|  |
| --- |
| Table 13.3. Solution to Spatial Equilibrium Model |
| Objective function = 9193.6 |
| Variables | Value | Reduced Cost | Equation | Level | Shadow Price |
| Supply |  |  | Supply Balance |  |  |
| U.S. | 79.6 | 0 | U.S. | 0 | 104.6 |
| Europe | 68.6 | 0 | Europe | 0 | 103.6 |
| Demand |  |  | Demand Balance |  |  |
| U.S. | 45.4 | 0 | U.S. | 0 | 104.6 |
| Europe | 51.4 | 0 | Europe | 0 | 103.6 |
| Japan | 51.4 | 0 | Japan | 0 | 108.6 |
| Shipments |  |  |  |  |  |
| U.S. to U.S. | 45.4 | 0 |  |  |  |
| U.S. to Europe | 0 | -4 |  |  |  |
| U.S. to Japan | 34.2 | 0 |  |  |  |
| Europe to U.S. | 0 | -2 |  |  |  |
| Europe to Europe | 51.4 | 0 |  |  |  |
| Europe to Japan | 17.2 | 0 |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# ice Endogenous Programming

##  Spatial Equilibrium Model – Example Solution

Consider model solution

 a) without any trade,

 b) with the U.S. imposing a quota of 2 units

c) with the U.S. imposing a 1 unit export tax while Europe imposes a 1 unit export subsidy.

|  |
| --- |
| 0 |
|  | Undistorted | No Trade | Scenario Quota | Tax/Subsidy |
| Objective | 9193.6 | 7506.3 | 8761.6 | 9178.6 |
| U.S. Demand | 45.4 | 62.5 | 61.5 | 46.4 |
| U.S. Supply | 79.6 | 62.5 | 63.5 | 78.6 |
| U.S. Price | 104.6 | 87.5 | 88.5 | 103.6 |
| Europe Demand | 51.4 | 60 | 40.7 | 50.4 |
| Europe Supply | 68.6 | 60 | 79.3 | 69.6 |
| Europe Price | 103.6 | 95 | 114.3 | 104.6 |
| Japan Demand | 51.4 | 0 | 40.7 | 51.4 |
| Japan Price | 108.6 | 160 | 119.3 | 108.6 |

# Price Endogenous Programming

##  Multi-Market Case

Supply

domestic supply: 

 import supply: 

and the demand curves:

 bread demand: ,

 cereal demand: ,

 export demand: .

 A QP problem which depicts this problem





## Price Endogenous Programming

##  Multi-Market Case – Solution

|  |
| --- |
|  Table 13.5. Solution to the Wheat Multiple Market Example |
| X b | 255.44 |
| X c | 867.15 |
| Xe | 1608.72 |
| Q d | 413.04 |
| Qi | 1391.29 |
| Pdb | 0.648 |
| Pdc | 0.540 |
| Pde | 3.239 |
| Psd | 3.239 |
| Psi | 3.239 |
| Shadow Price | 3.239 |

## Price Endogenous Programming

## Implicit Supply ‑ Multiple Factors/Products



|  |  |  |
| --- | --- | --- |
| β | = | index of different types of firms |
| k | = | index of production processes |
| j | = | index of production factors |
| i | = | index of purchased factors |
| h | = | index of commodities produced |
| Pdh(Zh) | = | inverse demand fn for the hth commodity |
| Zh | = | quantity of commodity h that is consumed |
| Psi(Xi) | = | inverse supply curve for the ith purchased input |
| Xi | = | quantity of the ith factor supplied |
| Qk | = | level of production process k undertaken by firm  |
| Chk | = | yield of output h from production process k |
| bjk | = | quantity of the jth owned fixed factor used in prod. Qk |
| aik | = | amount of the ith purchased factor used in prod. Qk |
| Yjβ | = | endowment of the jth owned factor available to firm β |

## Price Endogenous Programming

## Implicit Supply ‑ Multiple Factors/Products

Example:

|  |
| --- |
| Product Sale |
| Commodity | Price | Quantity | Elasticity | Computed Intercept (a) | Computed Slope (b) |
| Functional Chairs | 82 | 20 | -0.5 | 247 | -8.2 |
| Functional Tables | 200 | 10 | -0.3 | 867 | -66.7 |
| Functional Sets | 600 | 30 | -0.2 | 3600 | -100 |
| Fancy Chairs | 105 | 5 | -0.6 | 280 | -35 |
| Fancy Tables | 300 | 10 | -1.2 | 550 | -25 |
| Fancy Sets | 1100 | 20 | -0.8 | 2475 | -68.8 |
|  |  |  |  |  |  |
| Labor Supply |
| Plant | Price | Quantity | Elasticity | Computed Intercept (a) | Computed Slope (b) |
| Plant1 | 20 | 175 | 1 | 0 | .114 |
| Plant2 | 20 | 125 | 1 | 0 | .160 |
| Plant3 | 20 | 210 | 1 | 0 | .095 |

|  |
| --- |
|  |
|  |  |  | PLANT 1  | PLANT 2 |
| p |  |  Sell Sets FC FY |  Make Table FC FY | Sell Table FC FY | Sell Chair FC FY | Labor | Transport Chair FC FY | Make Functional Chairs Norm MxSm MxLg | Make FancyChairs Norm MxSm MxLg | LaborSupply | Transport Table FC FY | Transport Chair FC FY | Make TableFC FY | Make Functional Chairs Norm MxSm MxLg | Make Fancy ChairsNorm MxSm MxLg | Labor Supply |
| Objective |  W W  |  | W W  | W W  | -Z | -5 -5  | -15 -16 -16 | -25 -25 -26 | -Z | -7 -7 | -7 -7 | -80 -100 | -15 -16 -16.5 | -25 -25.7 -26.6 | -Z | Min |
| PLANT1 | Table FC |  1 |  -1 |  1 |  |  | -1 |  |  |  | -1 |  |  |  |  |  | ≤ 0 |
| Inventory FY |  1  |  -1  |  1 |  |  |  -1 |  |  |  |  -1 |  |  |  |  |  | ≤ 0 |
| Chair FC | 4 |  |  | 1 |  |  |  |  |  |  | -1 |  |  |  |  | ≤ 0 |
| Inventory FY |  6  |   |  |  1 |  |  |  |  |  |  |  -1 |  |  |  |  | < 0 |
| Labor  |  | 3 | 5 |  |  | -1 |  |  |  |  |  |  |  |  |  |  | ≤ 0 |
| Top Capacity |  | 1 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  | ≤ 50 |
| P L ANT2 | Chair FC |  |  |  |   |  | 1 | -1 -1 -1 |  |  |  |  |  |  |  |  | ≤ 0 |
| Inventory FY |  |  |  |   |  |  1 |  | -1 -1 -1 | - |  |  |  |  |  |  | ≤ 0 |
| Small Lathe |  |  |  |  |  |  | 0.8 1.3 0.2 | 1.2 1.7 0.5 |  |  |  |  |  |  |  | ≤ 140 |
| Large Lathe |  |  |  |  |  |  | 0.5 0.2 1.3 | 0.7 0.3 1.5 |  |  |  |  |  |  |  | ≤ 90 |
| Chair Bottom Cap |  |  |  |  |  |  | 0.4 0.4 0.4 |  1 1 1  |  |  |  |  |  |  |  | ≤ 120 |
| Labor |  |  |  |  |  |  | 1 1.05 1.1 | 0.8 0.82 0.84 | -1 | - |  |  |  |  |  | ≤ 0 |
| PLANT3 | Table FC |  |  |  |  |  |  |  |  |  | 1 |  | -1 |  |  |  | ≤ 0 |
| Inventory FY |  |  |  |  |  |  |  |  |  |  1 |  |  -1 |  |  |  | ≤ 0 |
| Chair FC |  |  |  |  |  |  |  |  |  |  | 1 |  | -1 -1 -1 |  |  | ≤ 0 |
| Inventory FY |  |  |  |  |  |  |  |  |  |  |  1 |  |  | -1 -1 -1 |  | ≤ 0 |
| Small Lathe |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
|  | -1 -1 -1 |  |
|  |  | -1 -1 -1 |
|  | 0.8 1.3 0.2 | 1.2 1.7 0.5 |
|  | 0.5 0.2 1.3 | 0.7 0.3 1.5 |
|  | 0.4 0.4 0.4 | 1 1 1 |
| 3 5 | 1 1.05 1.1 | 0.80 0.82 0.84 |
| 1 1 |  |  |

 | 0.8 1.3 0.2 | 1.2 1.7 0.5 |  | ≤ 130 |
| Large Lathe |  |  |  |  |  |  |  |  |  |  |  |  | 0.5 0.2 1.3 | 0.7 0.3 1.5 |  | ≤ 100 |
| Chair Bottom Cap |  |  |  |  |  |  |  |  |  |  |  |  | 0.4 0.4 0.4 | 1 1 1 |  | ≤ 110 |
| Labor |  |  |  |  |  |  |  |  |  |  |  |  | 1 1.05 1.1 | 0.80 0.82 0.84 | -1 | ≤ 0 |
| Top Capacity |  |  |  |  |  |  |  |  |  |  |  | 3 5 |  |  |  | ≤ 40 |

## Price Endogenous Programming

# Implicit Supply ‑ Multiple Factors/Products

Example - Solutions

Table 13.7. Solution of the Implicit Supply Example

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Rows | Slack | Shadow p |  |  | Variable Names | Level | Red Cost |
| Objective | 95779.1 |  |  | PLANT1 | Sell FC Set | 30.9 | 0 |
| PLANT1 | Table FC | 0 | 165.1 |  | Sell FY Set | 23.0 | 0 |
| Table FY | 0 | 228.5 |  | Sell FC Tables | 10.5 | 0 |
| Chair FC | 0 | 85.6 |  | Sell FY Tables | 12.9 | 0 |
| Chair FY | 0 | 110.8 |  | Sell FC Chairs | 19.6 | 0 |
| Labor  | 0 | 21.7 |  | Sell FY Chairs | 4.8 | 0 |
| Top Capacity | 50 | 20.0 |  | Make Table FC | 30.1 | 0 |
| PLANT2 | Chair FC | 0 | 80.6 |  | Make Table FY | 20.0 | 0 |
| Inventory FY | 0 | 105.8 |  | Hire Labor | 189.9 | 0 |
| Small Lathe | 140 | 35.6 |  | PLANT2 | Transport FC Chair | 105.0 | 0 |
| Large Lathe | 90 | 28.0 |  | Transport FY Chair | 48.9 | 0 |
| Carver | 90.9 | 0 |  | Make Table FC | 0 | -69.3 |
| Labor | 0 | 23.1 |  | Make Table FY | 0 | -115.5 |
|  |  |  |  | Make FC Chair N | 105.0 | 0 |
|  |  |  |  |  S | 0 | -11.6 |
| PLANT3 | Table FC | 0 | 145.1 |  |  L | 0 | -4.8 |
| Inventory FY | 0 | 208.5 |  | Make FY Chair N | 44.9 | 0 |
| Chair FC | 0 | 78.6 |  |  S | 0 | -7.76 |
| Inventory FY | 0 | 103.8 |  |  L | 4.0 | 0 |
| Small Lathe | 130 | 35.1 |  | Hire Labor | 144.4 | 0 |
| Large Lathe | 100 | 27.6 |  | PLANT 3 | Transport FC Table | 11.4 | 0 |
| Carver | 109.2 | 0 |  | Transport FY Table | 15.9 | 0 |
| Labor | 0 | 21.7 |  | Transport FC Chair  | 38.2 | 0 |
| Top Capacity | 27.32 | 0 |  | Transport FY Chair | 93.9 | 0 |
|  |  |  |  |  | Make FC Table  | 11.4 | 0 |
|  |  |  |  |  | Make FY Table | 15.9 | 0 |
|  |  |  |  |  | Make FC Chair N | 38.2 | 0 |
|  |  |  |  |  |  S | 0 | -11.3 |
|  |  |  |  |  |  L | 0 | -4.7 |
|  |  |  |  |  | Make FY Chair N  | 5.0 | 0 |
|  |  |  |  |  |  S | 0 | -7.6 |
|  |  |  |  |  |  L | 19.0 | 0 |
|  |  |  |  |  | Hire Labor | 227.9 | 0 |

## Price Endogenous Programming

# Integrability

The system of product demand functions is

P = G - HZ

and the system of purchased input supply functions is

R = E + FX

The Jacobians of the demand and supply equations are H and F, respectively. Symmetry of H and F implies that cross price effects across all commodity pairs are equal; i.e.,

∂Pdr/∂Qdh = ∂Pdh/∂Qdr for all r ≠ h

∂Psr/∂Qsh = ∂Psh/∂Qsr for all r ≠ h

In the case of supply functions, classical production theory assumptions yield the symmetry conditions.

The Slutsky decomposition reveals that for the demand functions, the cross price derivatives consist of a symmetric substitution effect and an income effect. The integrability assumption requires the income effect to be identical across all pairs of commodities or to be zero.

## Price Endogenous Programming

# Imperfect Competition

Suppose one begins with a classic LP problem involving two goods and a single constraint; i.e.,

 Max P1X - P2Q

 s.t. X - Q ≤ 0

 X, Q ≥ 0

However, rather than P1 and P2 being fixed, suppose we assume that they are functionally dependent upon quantity as given by

 P1 = a - bX

 P2 = c + dQ

Now suppose one simply substitutes for P1 and P2 in the objective function. This yields the problem

 Max aX - bX12 - cQ - dQ2

 s.t. X - Q ≤ 0

 X, Q ≥ 0

Note the absence of the 1/2's in the objective function. If one applies Kuhn‑Tucker conditions to this problem, the conditions on the X variables, assuming they take on non‑zero levels, are

 a - 2bX1 - λ = 0

 -(c + 2dQ) + λ = 0

# Price Endogenous Programming

## Imperfect Competition

[I] Monopolist – Monopsonist

Max aX - bX2 - cQ - dQ2

 s.t. X - Q ≤ 0

 X, Q ≥ 0

[II] Monopolist – Supply Competitor

Max aX - bX2 - cQ - ½ dQ2

 s.t. X - Q ≤ 0

 X, Q ≥ 0

[III] Demand Competitor - Monopsonist

Max aX - ½ bX2 - cQ - dQ2

 s.t. X - Q ≤ 0

 X, Q ≥ 0

[IV] Demand Competitor – Supply Competitor

Max aX - ½ bX2 - cQ - ½ dQ2

 s.t. X - Q ≤ 0

 X, Q ≥ 0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | I | II | III | IV |
| X b | 127.718 | 142.335 | 226.067 | 255.46 |
| X c | 433.579 | 449.821 | 834.526 | 867.16 |
| X e | 804.357 | 1096.705 | 1021.340 | 1608.71 |
| Q d | 206.521 | 393.533 | 216.311 | 413.04 |
| Qi | 695.642 | 806.589 | 989.330 | 1391.29 |
| Pdb | 0.699 | 0.693 | 0.660 | 0.649 |
| Pdc | 0.669 | 0.665 | 0.550 | 0.540 |
| Pde | 3.3196 | 3.29033 | 3.2979 | 3.239 |
| Psd | 2.6196 | 3.18066 | 2.6489 | 3.239 |
| Psi | 3.6196 | 3.18066 | 3.1999 | 3.239 |
| Shadow Price | 3.2391 | 3.18066 | 3.2979 | 3.239 |

## Price Endogenous Programming

# Imperfect Competition



N firms in the market

½+1/(2N)

N- infinity 1/2+0

N=1

One more example

Suppose we have the joint products problem from chapter 5

Max

s.t.

Now suppose we think this is a typical farm of which there are N in the US

To convert to total input and output multiply by N

Now suppose we think this is a typical farm of which there are N in the US

Max

s.t.

But suppose firms are not a price taker

What do we do?

Now suppose we think this is a typical farm of which there are N in the US

Max

s.t.

Replace revenue and cost terms with integrals