Cournot model: formulation

- Objective function: profits

$$B_e = p \cdot q_e - C_e$$

- Inverse demand function

$$p = f\left(\sum q_e\right)$$

- Cournot conjecture: vertical supply

System of equations

$$\frac{\partial p}{\partial q_e} = \frac{\partial p}{\partial q}\frac{\partial (q_e + q_{e*})}{\partial q_e} = p' \leftrightarrow \frac{\partial q_{e*}}{\partial q_e} = 0$$

Own output decision will not have an effect on the decisions of the competitors

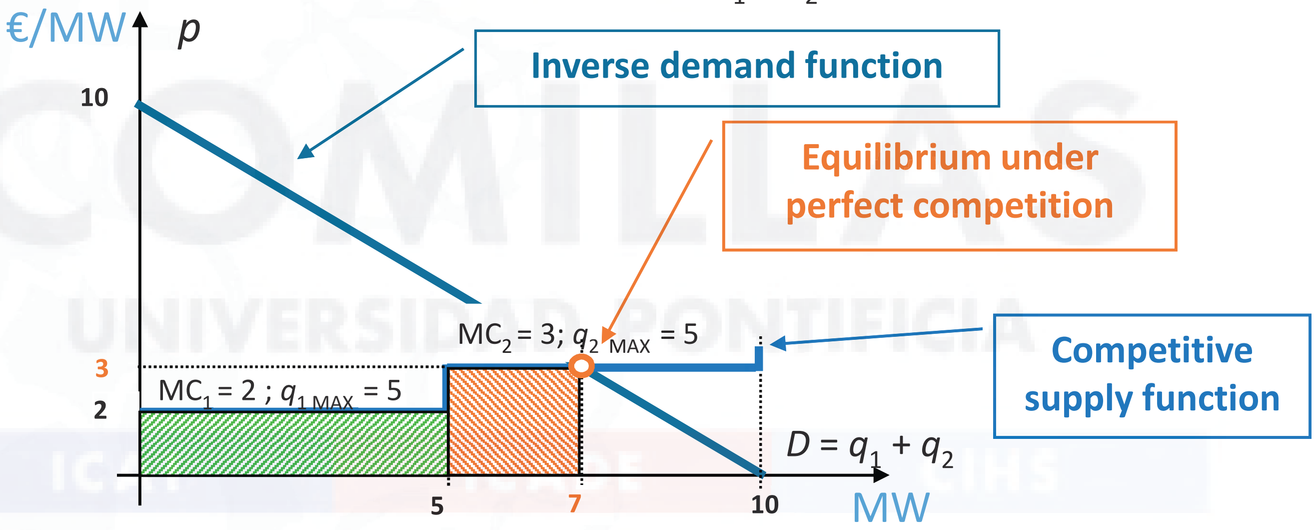
- Optimality conditions

$$\frac{\partial B_e}{\partial q_e} = 0 \rightarrow MR_e = p + q_e p' = MC_e(q_e) \rightarrow q_e = \frac{p - MC_e(q_e)}{-p'}$$

e	Company
$e*$	Other companies
B_e	Profits
p	Price
q_e	Output
C_e	Variable costs
MC_e	Marginal cost
MR_e	Marginal revenue
p-MC_e	**Price mark-up**

Cournot model: example (I)

- Perfect competition
  - Company 1: $MC_1 = 2$ €/MW $q_{1\ MAX} = 5$ MW
  - Company 2: $MC_2 = 3$ €/MW $q_{2\ MAX} = 5$ MW
  - Inverse demand function: $p = 10 - (q_1 + q_2)$



Cournot model: example (II)

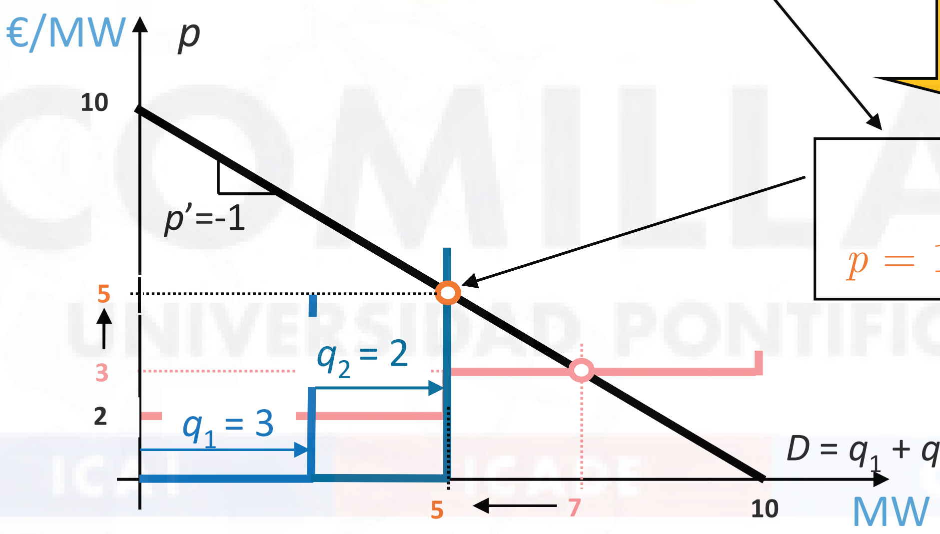
- Duopoly
  - Company 1: $MC_1 = 2$ €/MW
  - Company 2: $MC_2 = 3$ €/MW
  - IDF: $p = 10 - (q_1 + q_2)$; $p' = -1$

Cournot

$$\frac{\partial B_1}{\partial q_1} = 0 \rightarrow p + q_1 \cdot (-1) = 2$$
$$\frac{\partial B_2}{\partial q_2} = 0 \rightarrow p + q_2 \cdot (-1) = 3$$

Solving

$$q_1 = 3,\ q_2 = 2$$
$$p = 10 - (q_1 + q_2) = 5$$



Cournot Market Equilibrium

```
$Title Cournot Market Equilibrium Model

$OnText

Developed by

   Andrés Ramos
   Instituto de Investigación Tecnológica
   Escuela Técnica Superior de Ingeniería - ICAI
   UNIVERSIDAD PONTIFICIA COMILLAS
   Alberto Aguilera 23
   28015 Madrid, Spain
   Andres.Ramos@comillas.edu

   August 5, 2016

$OffText

sets
   p periods       / p1      /
   t thermal units / t1 * t2 /

parameters
   pMarginalCost(t) marginal costs                    [€ per MW] / t1 2, t2 3 /
   pMaxThrOutput(t) maximum output                          [MW] / t1 5, t2 5 /

   pPriceInter  (p) price intercept for each period [€ per MW  ] / p1 10      /
   pPriceSlope  (p) price     slope for each period [€ per MW^2] / p1 -1      /

positive variables
   vSMP       (  p) system marginal price [€ per MW]
   vThrOutput(t,p) thermal unit output          [MW]

binary   variables
   vCommitment(t,p) thermal unit commitment    [p.u.]
```

Cournot Market Equilibrium

```
equations
   eDemandFunction(  p) price as a function of the load [€ per MW]
   eDerivRevenue  (t,p) derivative of the net revenue   [€ per MW] ;

eDemandFunction(  p) .. vSMP(p) =g= pPriceInter(p)   + pPriceSlope(p)*sum[t, vThrOutput(t,p)] ;

eDerivRevenue  (t,p) .. pMarginalCost(t) =g= vSMP(p) + pPriceSlope(p)*       vThrOutput(t,p)  ;

model mCournot / eDemandFunction.vSMP eDerivRevenue.vThrOutput /

variable
   vTotalCost      total investment and operation cost

equations
   eTotalCost      total investment and operation cost [€]
   eLoadBalance(p) balance between generation and load [€ per MW] ;

eTotalCost      .. vTotalCost =e= - sum[(t,p), pMarginalCost(t)*vThrOutput(t,p)] ;

eLoadBalance(p) .. pMarginalCost('t2') =e= pPriceInter(p) + pPriceSlope(p)*sum[t, vThrOutput(t,p)] ;

model mPerfectCompetition / eTotalCost eLoadBalance /

variable
   vSocialWelfare  social welfare [€]

equations
   eSocialWelfare  social welfare [€] ;

eSocialWelfare .. vSocialWelfare =e= sum[p, (pPriceInter(p) + pPriceSlope(p)*sum[t, vThrOutput(t,p)]/2)*sum[t, vThrOutput(t,p)]]
                                                                                   - sum[(t,p), pMarginalCost(t)*vThrOutput(t,p)] ;

model mSocialWelfare / eSocialWelfare / ;

vThrOutput.up(t,p) = pMaxThrOutput(t) ;

* Cournot equilibrium model

solve mCournot using MCP ;

* perfect competition model (cost minimization)

solve mPerfectCompetition using MIP maximizing vTotalCost

* perfect competition model (social welfare maximization)

solve mSocialWelfare using QCP maximizing vSocialWelfare
```