AGRISPACE – a state-of-the-art recursive-dynamic regionalized Partial Equilibrium model for Norway presenting the full farm population

Wolfgang Britz[1] and Klaus Mittenzwei[2]

1 Institute for Food Resource Economics, University Bonn

2 Nibio, Oslo

Introduction

There is growing recognition that formal economic quantitative analysis can contribute to evidence-based policy making (Podhora et al. 2013). The EU, to give an example, now requires for any larger legislative project a formal impact assessment, and for policy arenas such as the Common Agricultural Policy, the application of economic modeling is explicitly recommended in the related guideline (EU 2009). Up-to-date, there was limited use of economic model results in agricultural policy design in Norway, which partly reflects the way policy instruments are decided about. But it might also be that available tools which could inform policy processes might have lacked features which render their use inviting, such as a state-of-the-art methodology, easy ways to explore results and salient outcomes matching the information needs of stakeholders. The AGRISPACE project financed by the Norwegian Research Council now offers the unique opportunity to build a new tool whose design reflects from the beginning desired properties to inform policy processes.

Desired properties

At least when existing policy instruments and budgetary spending is used as an indicator, the Norwegian society puts a high value on maintaining agricultural production and preventing high rates of structural change in the Norwegian farm population as well as to preserve a certain degree of self-sufficiency in food markets. Therefore, a tool to inform policies should model developments of agri-food demand, supply and trade and aspects of structural change in agriculture. As the agricultural sector in Norway in small in economic terms (about 1.5 % of GDP including considerable subsidies), a partial equilibrium model seems the appropriate tool.

Norwegian policy instruments related to the agri- food sector, both domestic ones and relating to border protection, aim to maintain agriculture even in regions where natural conditions are highly unfavorable. Accordingly, an appropriate tool must offer sufficient regional dis-aggregation to reflect differences in space. We draw here on the experiences with an existing tool, the agricultural sector model Jordmod (ref) which depicts Norway by 32 production regions. These regions reflect differences in subsidies to the farming sector, which turn relate to differences in agricultural production conditions such as climate or access to markets. As Norway is characterized by a low population density in larger parts of its area combined with large distances between regions, transport costs can be an important for input and output prices faced by farmers, and are partly subject to policy interventions. A spatial equilibrium approach seems therefore appropriate.

As there is an obvious interest in question of structural change in agriculture and its impact e.g. on productivity, but also land abandonment, the proposed tool models shall simulate on a yearly basis changes in the composition and size of the farm population. Relevant drivers at farm level such as profits are simulated for the full farm population based on results at the level of clusters of farms at regional level based on simple rules.

The market model

Basic structure

The market model is a Multi-Commodity model (MCMs) with behavioral equations to determine demand and explicit production functions for agricultural sectors, similar to Computable General Equilibrium (CGE) models. The use of MCMs is widespread for agricultural policy analysis (cf. Britz and Heckelei 2008); international organization such as the OECD and FAO maintain and apply such models (AGLINK-COSIMO, OECD 2007), but also research institutes respectively universities (e.g. FAPRI, Devadoss et. al. 1989; CAPRI, Britz and Witzke 2014; ESIM, Banse et al. 2005). Two strands of MCMs can be broadly distinguished: models where individual equations are econometrically estimated such that the equations describing specific markets differ structurally (e.g. in FAPRI and AGLINK-COSIMO), and template models where equations are structurally identically (e.g. in ESIM and CAPRI), and differences are expressed by parameters, an approach pioneered by Roningen et al. 1991. Our approach falls in the later class of templated models, as do most CGEs.

Based on the spatial equilibrium (SPE) approach, we consider commodities as homogenous, such that price differences in space depend on transport margins and policy instruments such as import tariffs, and not a spatially differentiated quality differences. It reflects spatial arbitrage, i.e. price differences between two regions cannot exceed the bi-lateral per unit transport and transaction costs. The use of the spatial equilibrium approach is relatively seldom in a Multi-Commodity setting; it is found in some single commodity models (e.g. Anania 2005, Nolte 2008) and in models which combine Mathematical Programming with price endogeniety (McCarl and Spreen 1980). One reason for the limited application of SPE in international trade analysis is the fact that calibration against observed trade remains challenging (e.g. Paris et al. 2011, Wieck et al. 2012), especially as the typically observed counter-trade in international data bases cannot be captured. However, in Agrispace, we use that approach only to describe price formation inside of Norway, where these problems are of limited relevance. The more widely used approach to depict bi-lateral trade flows in equilibrium models is based on the Armington assumption (1969), especially in CGEs, but also e.g. in the global market module of CAPRI. Assuming goods differentiated by regional origin inside Norway seems however not appropriate; it would also be unclear how to construct a set of interregional flows to calibrate an Armington based model against as observation on such interregional flows are not available. A net-trade approach, on the other hand, would face the challenge how to properly reflect the large transport costs between Norwegian regions.

Furthermore, we assume that markets are competitive, and that producers minimize costs and consumer utility, standard assumptions in CGE models. These assumptions are reflected in the choice of functional form and parameterization for the demand side: we use (semi-)flexible function forms and ensure global adherence to regularity conditions. That approach is still not widely applied in agricultural MCM analysis. Whereas non-templated MCM models basically cannot control for regularity, most other MCMs (e.g. ESIM) typically use double-log functions where regularity can only be imposed locally. To the knowledge of the authors, the models mentioned above do not control for curvature. For the supply side, we use FOC derived from CES production functions under constant-Returns-to-scale in combination with CET function to describe factor supply, the usual approach in CGEs.

A Generalized Leontief (GL) expenditure system drives final demand. The system has the advantage that curvature can be easily imposed globally, but then does not allow for Hicksian complementarity between products, a restriction of limited relevance in our setting. The choice of functional forms reflects long-standing experience with the CAPRI system (Britz and Witzke 2014). The GL system used is based on the following family of indirect utility functions:

$$U(p,y) = \frac{-G}{(y-F)}$$

depending on consumer prices *p* and income *y* where G and F are functions of degree zero in prices (Ryan and Wales 1998). Using Roy's identity, the following Marshallian demands are derived:

$$x_i = F_i + \frac{G_i}{G}\left(y - F\right)$$

F is simply chosen as the sum of commitments, i.e. additive term which describe consumption quantities independent of prices and income, while term G is based on the GL function form, where b is a matrix of parameter describing own and cross-price effects:

$$G = \sum_i \sum_j b_{ij} \sqrt{p_i \; p_j}$$

While using the spatial arbitrage condition in the model reflects the relation between price difference in space and trade, we refrain from using the Takayama-Judge approach which requires explicitly maximizing a social welfare function. In agricultural PEs, the SPE approach is typically combined with Mathematical Programming as proposed by McCarl and Spreen 1980, i.e. a Leontief presentation of agricultural supply. Such a set-up is applied e.g. by GLOBIOM (Havlik et al. 2011), but also the original JORDMOD model. Price endogeniety is typically implemented by integrating linear inverse demand functions which only reflect own-price effects in the objective function, which is not in line with a regular demand system. These functions are often piecewise linearized to yield a price endogenous LP, which allows for a very high number of endogenous variables.

Given the focus on Norway, the possible advantages for staying completely linear are limited, such that we rather maintain micro-economic stringency. Therefore, instead of using the direct optimization approach used in these LP based models mentioned above, we incorporate the first order conditions reflecting optimal behavior of the individual agents (producers, demanders, traders) as behavioral equations in our model, as typically done in Multi-Commodity and Computable General Equilibrium models.

Transport and tariff cost minimizing behavior, which is embedded in the spatial equilibrium approach, yields the Kuhn-Tucker-Karush based spatial arbitrage condition which requires capturing properly the complementarities between trade flows and inter-regional price differences. The latter necessitates the MCP (Mixed Complemenarity Problem) solution format.

Summarizing, the model has basically four types of (in)equalities: (1) FOC derived from cost minimizing under an explicit production function, (2) the demand equations which define demand quantities at the endogenous prices, (3) market clearing conditions which define regional prices, and (4) spatial arbitrage which simultaneously determines trade flows and differences in regional prices. As Norway is considered a small country, we treat international prices as fixed.

Supply side

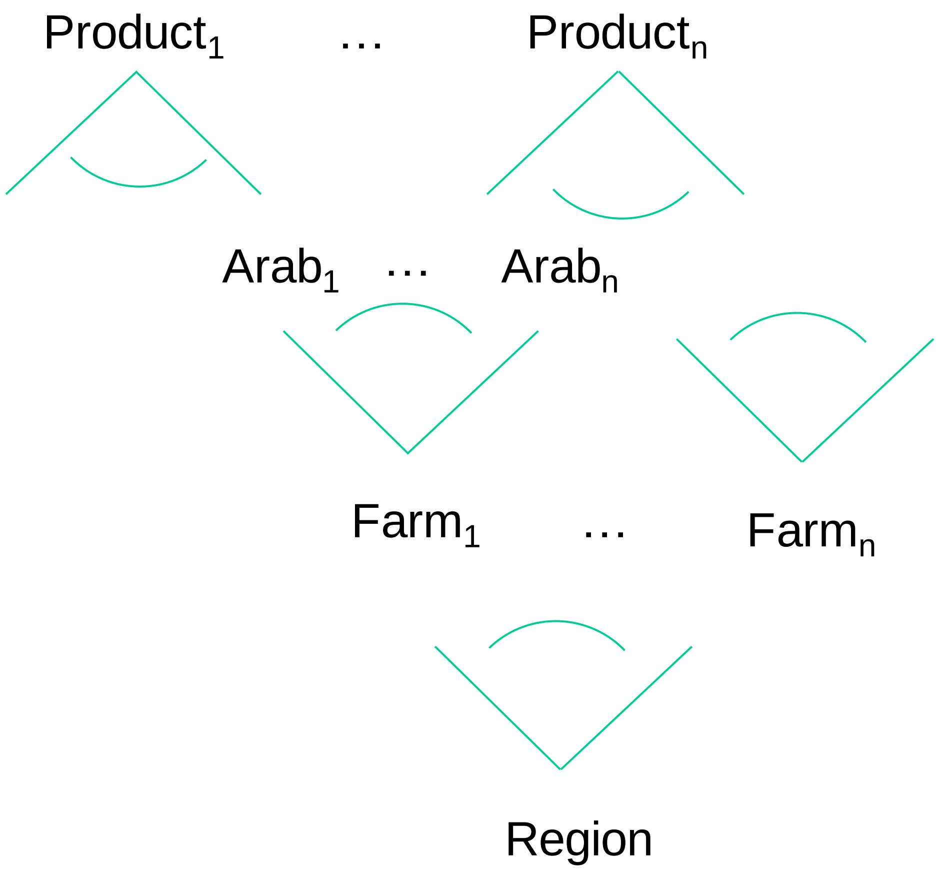
The supply side of the models breaks down each of the 32 regions in 4-16 clusters of farms which are modeled by aggregate production and factor supply functions. The clusters are derived by statistical analysis from the about 40.000 individual farms underlying the agricultural census. That implies that each cluster represents in average about 140 farms.

Production output for each cluster is determined by product specific CES function with constant returns to scale. Capital, land – differentiated by arable, permanent and grassland -, labor, general intermediate inputs, fodder and concentrates are the six input categories differentiated. The production activities compete at cluster farm level for the two quasi-fixed factors capital and labor and the three land categories. The competition for these factors is described by CET-functions, we hence assume that input qualities used in the different production branches differ such that factor returns need not be identical. Total supply of capital and labor to the cluster of farms is driven by a linear function depending on the relation on average returns on farm and exogenous off-farm prices which can hence be understood as a farm-household model with preferences for on/off farm employment and capital use.

The three land categories feature regional markets, where the total land stock reacts with land supply elasticities to changes average regional returns. The average regional returns represent a CET nest which distributes land to the different farm group (clusters) according to average returns to each land category in the cluster. That in turn implies that land markets are not perfect: returns between the aggregate cluster farms are not identical.

The model hence allows to simultaneously simulated allocation decisions at the level of farm groups as in a classical aggregate programming model, however user non-linear production functions, and aspects of structural change, depicted by simulating changes in labor and financial assets (=capital) on and off farm and competition in land markets between farm groups. At the same time it allows as a typical MCM to depict changes in production, demand, trade and price at national level, and related to the SPE approach, at regional level inside Norway.

The Graphic below depicts that process for the example of arable land. Different products compete at aggregate farm level, i.e. for each cluster of farms, for arable land. The relative returns per ha of each product determine the arable land shares at farm group level. The average returns per ha of arable land at farm group level drives the total arable land at farm group level, now depending on returns relative to other farm groups in the same region.



The structure of the model is now discussed in more detail based on the model's equation.

Factor supply and demand

The offer of labor and capital to the farming sector is based on linear equations:

```
*
* --- Behavioral equation for supply of primary factors
*
 e_primFacOffer(rn,clEmpty,primFacI) $ p_cnstPrimFac(rn,clEmpty,primFacI) ..

     v_inputUseCL(rn,clEmpty,primFacI)
           =E=  [ p_cnstPrimFac(rn,clEmpty,primFacI)
                    + p_slopePrimFac(rn,clEmpty,primFacI)
                      * v_inputPriceCL(rn,clEmpty,primFacI)/p_inputPriceOffFarm(rn,PrimFacI) ];
```

Where *v_inputUseCl* on the LHS is total factor use in agriculture, and the RHS determines the supply driven by a linear function. The price *v_inputPr*iceCL depicts marginal returns to factors in agriculture in that region.

The competition between different farm types for the supplied factors is based on the following two equations, where the first distributes the regional supply to the different farm types based on relative factor returns:

```
*
* --- CET aggregator from supply from region to clusters
*     (only if factor supply function is not cluster specific)
*
 e_CETPrimFacCL(rn,cl,primFacI) $ ((p_shareParCL(rn,cl,primFacI) $ rn_cl(rn,cl))
                                      and (not p_cnstPrimFac(rn,cl,primFacI)))  ..

      v_inputUseCL(rn,cl,primFacI) =E=

          v_inputUseCL(rn,"",primFacI) * p_shareParCL(rn,cl,primFacI)
             * [v_inputPriceCL(rn,cl,primFacI)/v_inputPriceCL(rn,"",primFacI)] ** p_CETElasCL(rn,primFacI);
```

And the second provides the dual price aggregator which defines average returns across farm types:

```
 e_inputPriceCL(rn,empty,primFacI) $ (sum(rn_cl(rn,cl), p_shareParCL(rn,cl,primFacI)) ) ..

      v_inputPriceCL(rn,empty,primFacI)
          =E= sum( rn_cl(rn,cl) $  p_shareParCL(rn,cl,primFacI),

                   p_shareParCL(rn,cl,primFacI)
                    * v_inputPriceCL(rn,cl,primFacI) ** (1+p_CETElasCL(rn,primFacI))

              )**(1/(1+p_CETElasCL(rn,primFacI)));
```

Inside each farm types *cl*, factors are distributed according to the factor returns (= *v_inputPriceCL*) of the different production activities *xxAgr*:

```
 e_CETPrimFacXX(rn,cl,primFacI,xxAgr) $ (p_shareParXX(rn,cl,primFacI,xxAgr) $ rn_cl(rn,cl)) ..

      v_inputUse(rn,cl,primFacI,xxAgr)
         =E= v_inputUseCL(rn,cl,primFacI) * p_shareParXX(rn,cl,primFacI,xxAgr)
               * [v_inputPrice(rn,cl,primFacI,xxAgr)/v_inputPriceCL(rn,cl,primFacI) ] ** p_CETElasXX(rn,primFacI);
..
```

The average return to factors in each cluster is again based on a dual price aggregator:

```
 e_inputPriceXX(rn,cl,primFacI) $ (sum(xxAgr, p_shareParXX(rn,cl,primFacI,xxAgr)) $ rn_cl(rn,cl)) ..

      v_inputPriceCL(rn,cl,primFacI)
          =E= sum(xxAgr $ p_shareParXX(rn,cl,primFacI,xxAgr)   ,

                  p_shareParXX(rn,cl,primFacI,xxAgr)
                   * v_inputPrice(rn,cl,primFacI,xxAgr) ** (1+p_CETElasXX(rn,primFacI))

              )**(1/(1+p_CETElasXX(rn,primFacI)));
```

The individual factor demands are driven by nested-CES structures:

```
* ---  Demand for inputs inside technology nests

e_xftNesteq(rn,cl,inputs,xxAgr) $ (sum(tNest_f_a(tNest,inputs,xxAgr),p_sigmaNest(rn,tNest,xxAgr))
                                    and v_costShare.l(rn,cl,inputs,xxAgr)) ..
*
   v_inputUse(rn,cl,inputs,xxAgr) =e= sum(tNest_f_a(tNest,inputs,xxAgr) $ p_sigmaNest(rn,tNest,xxAgr),
*
*    --- factor cost share times demand for technology nest
*
     v_costShare(rn,cl,inputs,xxAgr)*v_xtNest(rn,cl,tNest,xxAgr)
*
*   --- relative price impact (in case sigmaNest is not 0)
*
     * [ { (v_pTNest(rn,cl,tNest,xxAgr)/v_inputPrice(rn,cl,inputs,xxAgr))
                              **p_sigmaNest(rn,tNest,xxAgr)
         } $ (p_sigmaNest(rn,tNest,xxAgr) ne eps)
       +  1 $ ( p_sigmaNest(rn,tNest,xxAgr) eq eps) ]

*   --- technology shifter
                  * v_lambdaf(rn,cl,inputs,xxAgr)**(p_sigmaNest(rn,tNest,xxAgr)-1));
```

where the cost share parameter *v_costshare* are fixed during simulation. The prices for the nests are calculated as above from dual price aggregator function:

```
* --- Price for technology nests

e_ptNest(rn,cl,tNest,xxAgr) $ p_data(rn,cl,"Q",tNest,xxAgr,"%firstYear%") ..

   v_ptNest(rn,cl,tNest,xxAgr) =e=
*
*  --- aggregate over factors  assigned to technology nests
*
      [ sum(tNest_f_a(tNest,inputs,xxAgr) $ v_costShare.l(rn,cl,inputs,xxAgr),
             v_costShare(rn,cl,inputs,xxAgr)
                 * (v_inputPrice(rn,cl,inputs,xxAgr)/v_lambdaf(rn,cl,inputs,xxAgr))**(1-p_sigmaNest(rn,tNest,xxAgr)))
*
*  --- aggregate over sub technology nests assigned to that technology nests
*
      + sum(tNest_n_a(tNest,tNest1,xxAgr) $ v_costShare.l(rn,cl,tNest1,xxAgr),
             v_costShare(rn,cl,tNest1,xxAgr)
                 * v_ptNest(rn,cl,tNest1,xxAgr)**(1-p_sigmaNest(rn,tNest,xxAgr)))

      ]**(1/(1-p_sigmaNest(rn,tNest,xxAgr)))
  ;
*
```

The demand for sub-nests is defined as follows:

```
e_xtNest(rn,cl,tNest,xxAgr) $ v_costShare.l(rn,cl,tNest,xxAgr)  ..

    v_xtNest(rn,cl,tNest,xxAgr) =e=
*
*        --- nest is part of other nests
*
      + sum(tNest_n_a(tNest1,tNest,xxAgr), v_costShare(rn,cl,tNest,xxAgr) * v_xtNest(rn,cl,tNest1,xxAgr)
*
*   --- relative price impact (in case sigmadNest is not 0)
*
      * [ { (v_ptNest(rn,cl,tNest1,xxAgr)/v_ptNest(rn,cl,tNest,xxAgr))
                                  **p_sigmaNest(rn,tNest1,xxAgr)
          } $ (p_sigmaNest(rn,tNest1,xxAgr) ne eps)
      + 1 $ (p_sigmaNest(rn,tNest1,xxAgr) eq eps) ]);
```

The price for the top level nest defines the unit costs – under constant returns to scale, average are equal to marginal costs – which must match the expected producer incentive price:

```
e_unitCost(rn,cl,"top",xxAgr) $ p_data(rn,cl,"Q","top",xxAgr,"%firstYear%") ..

       v_priceProdE(rn,cl,xxAgr)     $ (not sameas(xxAgr,"fodd"))
     + v_inputPriceCL(rn,cl,"fodd") $ (    sameas(xxAgr,"fodd"))
        =E= v_ptNest(rn,cl,"top",xxAgr);
```

A special case provides fodder – as seen also above – which is assumed to be non-tradable:

```
e_foddPrice(rn,cl,"fodd",xxAgr) $ p_data(rn,cl,"Q","fodd",xxAgr,"%firstYear%") ..
    v_inputPrice(rn,cl,"fodd",xxAgr) =E= v_inputPriceCL(rn,cl,"fodd");


e_foddMrk(rn,cl,"fodd") $ p_data(rn,cl,"Q","top","fodd","%firstYear%") ..
     v_xtNest(rn,cl,"top","fodd") =E= sum(xxAgr $ p_data(rn,cl,"Q","fodd",xxAgr,"%firstYear%"),
                                                   v_inputUse(rn,cl,"fodd",xxAgr));
```

Dairy sub-system

The average prices for dairy outputs *v_dairyPrice* must exhaust dairy production costs:

```
e_dairyPrice(rn) $ p_shiftParDairy(rn)   ..

   v_dairyPrice(rn) =e=
*
*  --- aggregate over factors  assigned to technology nests
*
      sum(inputs $ p_costShareDairy(rn,inputs),
             p_costShareDairy(rn,inputs)
                 * [(  v_inputPriceDairy(rn,inputs) $ (not sameas(inputs,"milk"))
                     + v_priceMarket(rn,"milk") $ sameas(inputs,"milk"))
                         /v_lambdaD(rn,inputs)]**(1-p_sigmaDairy(rn)))
        **(1/(1-p_sigmaDairy(rn)))
  ;
```

While total dairy output is a CES aggregate of the inputs, under decreasing returns to scale

```
e_dairyOutput(rn) $ p_shiftParDairy(rn) ..

  v_dairyOutput(rn) =E= p_shiftParDairy(rn)
        * sum(inputs $ p_costShareDairy(rn,inputs),
                 p_costShareDairy(rn,inputs)*v_lambdad(rn,inputs)
                   *v_inputUseDairy(rn,inputs)**((p_sigmaDairy(rn)-1)/p_sigmaDairy(rn)))**(0.99*p_sigmaDairy(rn)/(p_sigmaDairy(rn)-1));
```

Input demand of the dairy is driven by relative prices and total dairy output:

```
*
  e_inputUseDairy(rn,inputs) $ p_costShareDairy(rn,inputs) ..

      v_inputUseDairy(rn,inputs)

       =E=
*
*    --- factor cost share times demand for technology nest
*
       p_costShareDairy(rn,inputs)*v_dairyOutput(rn)
*
*   --- relative price impact (in case sigmaNest is not 0)
*
          * (v_dairyPrice(rn)/(   v_inputPriceDairy(rn,inputs) $ (not sameas(inputs,"milk"))
                                + v_priceMarket(rn,"milk") $ sameas(inputs,"milk")))**p_sigmaDairy(rn)

*   --- technology shifter
                 * v_lambdad(rn,inputs)**(p_sigmaDairy(rn)-1);
```

The distribution of total output to individual dairy products follows a CET:

```
e_dairyQuant(rn,xxDairy) $ p_outputShareDairy(rn,xxDairy) ..

  v_dairyQuant(rn,xxDairy) =e=

     p_outputShareDairy(rn,xxDairy)*v_dairyOutput(rn)
            *    ( v_priceMarket(rn,xxDairy)/v_dairyPrice(rn)) ** p_CETElasDairy(rn);
```

Which consequently requires value exhaustion:

```
e_dairyPriceCET(rn) $ p_shiftParDairy(rn) ..

  v_dairyPrice(rn) =e= sum(xxDairy $ p_outputShareDairy(rn,xxDairy),
                        p_outputShareDairy(rn,xxDairy)
                         * v_priceMarket(rn,xxDairy) ** (1+p_CETElasDairy(rn))
                 )**(1/(1+p_CETElasDairy(rn)));
```

Feed industry

The feed industry mixes different concentrates (FCER – cereal rich, FPRO – protein rich, FMIL – based on dairy products, FOTH – based on other components) from individual ingredients *v_feedQuant* under a CES production technology:

```
e_feedQuant(rn,xxAgr) $ p_data(rn,"","","feedQuant",xxAgr,"%firstYear%") ..

    v_feedQuant(rn,xxAgr)
      =E= sum(feed_to_o(feed,xxAgr),
            v_feedQuant(rn,feed) * p_shareParFeed(rn,xxAgr)
              * (v_priceProd(rn,feed)/v_priceProd(rn,xxAgr)) ** p_subsElasFeed(rn,feed));
```

The average price of the concentrate is based on a dual price aggregator:

```
e_priceFeedDual(rn,feed) $ p_data(rn,"","","feedQuant",feed,"%firstYear%") ..

     v_priceProd(rn,feed)
      =E= sum(feed_to_o(feed,xxAgr),
            p_shareParFeed(rn,xxAgr) * v_priceProd(rn,xxAgr) ** (1-p_subsElasFeed(rn,feed))
         )**(1/(1-p_subsElasFeed(rn,feed)));
```

Which enters as the input price into the agricultural production system:

```
e_priceFeedXX(rn,cl,feedI,xxAgr) $ p_data(rn,cl,"Q",feedi,xxAgr,"%firstYear%") ..
      v_inputPrice(rn,cl,feedI,xxAgr) =E= v_priceProd(rn,feedI);
```

Price system

Price differences in space are driven by the spatial arbitrage decision, i.e. they cannot exceed per unit transport costs and tariffs:

```
*
   e_spatAbri(rm,rm1,xx) $ ( (v_tradeflows.range(rm,rm1,xx) ne 0) $ (not sameas(rm,rm1)) $ traded(xx)) ..
*
       v_priceMarket(rm,xx) + p_transCost(rm,rm1,xx)
                            + p_tariff(rm,rm1,xx)
                            + v_flexTariff(rm,rm1,xx) $  (rh(rm1) and sameas(rm,"row") and (p_guarPrice(xx) gt eps))
                            + v_trqRent(xx)           $  (rh(rm1) and sameas(rm,"row") and (p_trq(xx) gt eps))
            =G= v_priceMarket(rm1,xx);
```

Flexible tariffs ensure that market prices cannot fall below guaranteed prices, that mechanism is integrated with an appropriate pairing in the model definition (`e_guarPrice.v_flexTariff`)

```
  e_guarPrice("row",rh,xx) $ (p_guarPrice(xx) gt eps) ..

    v_priceMarket(rh,xx) =G= p_guarPrice(xx);
```

A similar mechanism defines rents under Tariff Rate Quotas (`e_trq.v_trqRent`):

```
    e_trq(xx) $ (p_trq(xx) gt eps) ..

      p_trq(xx) =G= sum(rh, v_tradeFlows("row",rh,xx)) ;
```

Where *rh* depicts Norwegian harbours used as import points from the world market.

The demand price v_priceDem differs from the market price by a fixed processing and marketing margins and potential consumer subsidy equivalents *p_cse*:

```
*
*  --- consumer prices equal to market prices plus margins
*
   e_priceDem(rn,xx) $ (v_priceDem.range(rn,xx) $ p_data(rn,"","","finDemQuant",xx,"%firstYear%")) ..

      v_priceDem(rn,xx)
         =E= v_priceMarket(rn,xx) + p_priceDemMarg(rn,xx) + p_cse(rn,xx);
```

Producer prices *v_priceProd* are equal to market prices:

```
 e_priceProd(rn,xx) $  (   v_priceProd.range(rn,xx)
                         $ p_data(rn,"","","priceProd",xx,"CUR")
                         $ (not (feed(xx) or sameas(xx,"fodd")))) ..
*
     v_priceProd(rn,xx) =E= v_priceMarket(rn,xx);
```

The expected producer incentive price considers potentially lagged prices, coupled payments and production quota rents:

```
  e_priceProdE(rn,cl,xx) $ (p_data(rn,cl,"","priceProdINC",xx,"cur") and (not sameas(xx,"fodd"))) ..

      v_priceProdE(rn,cl,xx) =E=
                   p_priceExpWeightCur(rn)   * v_priceProd(rn,xx)
             + (1-p_priceExpWeightCur(rn)) * p_priceProdPast(rn,xx)
             + p_couplPayment(rn,cl,xx)
             - v_quotRent(rn,cl,xx)       $ (p_prodQuot(rn,cl,xx) gt eps);
```

Market clearing

The aggregate agricultural supply quantities *v_prodQuant* are equal to the sum of the top level CES nest quantities for each farm type:

```
e_aggSup(rn,xxAgr)  $ (p_data(rn,"","","prodQuant",xxAgr,"%firstYear%") and (not sameas("fodd",xxAgr))) ..

    v_prodQuant(rn,xxAgr)
       =E= sum(rn_cl(rn,cl) $ p_data(rn,cl,"Q","top",xxAgr,"%firstYear%"), v_xtNest(rn,cl,"top",xxAgr));
```

Market clearing in each region needs to take bi-lateral trade flows into account:

```
 e_mrkBal(rm,xxFin) $ (  (v_priceMarket.range(rm,xxFin) or p_trim)
                        $ ( not (feed(xxFin) or sameas("fodd",xxFin) or sameas("feed",xxFin) ) ))  ..

*
*   --- Imports plus production
*
            sum( rm1, v_tradeFlows(rm1,rm,xxFin)) $ traded(xxFin)
          + v_prodQuant(rm,xxFin) $ p_data(rm,"","","prodQuant",xxFin,"%firstYear%")
          + v_dairyQuant(rm,xxFin)  $ (not sameas(xxFin,"milk"))

          =E=
*
*   --- final demand plus feed plus milk use by dairy plus export
*
           v_finDemQuant(rm,xxFin)
         + v_feedQuant(rm,xxFin)
         + sum(sameas(xxFin,"milk"), v_inputUseDairy(rm,"milk"))
         + sum(rm1, v_tradeFlows(rm,rm1,xxFin)) $ traded(xxFin);
```

Demand system

The details of the demand have been discussed above, here only the implementation in GAMS is shown:

```
*
* --- Definition of human consumption for the Generalised Leontief expenditure funtion
*
  e_finDemQuant(rn,xx) $ ( p_data(rn,"","","finDemQuant",xx,"cur")
                           $ (v_finDemQuant.range(rn,xx) or p_trim))  ..
*
    v_finDemQuant(rn,xx)/p_pop(rn) =E=
*
          v_GLDemGi(rn,xx)/v_GLDemG(rn) * ( p_inc(rn) - v_GLDemF(rn)) + pv_pdGL(rn,xx);
*
* --- Definition of F as sum of Di multiplied with prices for the Generalised Leontief expenditure function
*     PD: commitments / linear terns in the individual demand functions

  e_GLDemF(rn) $ sum(xx, p_data(rn,"","","finDemQuant",xx,"cur")) ..
*
     v_GLDemF(rn) =E= SUM(xx, v_priceDem(rn,xx) * pv_pdGL(rn,xx));
*
* --- Definition of function G for the Generalised Leontief expenditure funtion
*     (per capita)
*
  e_GLDemG(rn) $ sum(xx, p_data(rn,"","","finDemQuant",xx,"cur")) ..
*
     v_GLDemG(rn)

      =E= SUM( (xx,yy),
                   pv_bGL(rn,xx,yy)
                         * SQRT(v_priceDem(rn,xx)*v_priceDem(rn,yy)));
*
* --- Definition of first derivatives of G called Gi for the Generalised Leontief expenditure function
*
  e_GLDemGi(rn,xx) $ (p_data(rn,"","","finDemQuant",xx,"cur")
                           $ (v_finDemQuant.range(rn,xx) or p_trim))  ..

     v_GLDemGi(rn,xx)
               =E= SUM( yy,
                            pv_bGL(rn,xx,yy)
                                  * SQRT(v_priceDem(rn,yy)/v_priceDem(rn,xx)));
```

Baseline

The baseline of the model is technically a normal model run which uses exogenous shifters under no-change in policy. Specifically, we use population projections at regional level and an assumption of a 2% increase in real income. In order to reflect that fact that income elasticity for food expenditure at high GDP per capita level reflects rather quality than quantity demand, we update the consumer price margins dynamically based on the income effect. Specifically, we assume that 80% of the income elasticity goes into the margins, i.e. demand for processing, transport, marketing, services related to food consumption such as eating outside.

We assume that the increase in real per-capita income wage rates, currently we assume by 2.5% p.a., while capital returns are assumed to increase by 1.5%. These exogenous changes would without higher returns of labor and capital in the farming sector imply reduced labor and capital use. In agriculture, we assume an exogenous increase in the marginal productivity of capital by 1.5% and general intermediate inputs and for labor by 1%. Border prices for agricultural products are assumed to fall by 2% p.a., which also reflects the assumption that the NKR will appreciate against the EURO and dollar in the medium term.

Parameterization

Demand

The demand system is parameterized based on elasticities for broader food categories reported by the Economic Research Service of the US department for Agriculture. Following the approach applied in CAPRI (Britz and Witzke 2014), we use substitution elasticities inside the food groups to derive individual own and cross price elasticities. Based on a Bayesian approach, using a Highest Posterior Density estimator (HPD, Heckelei et al. 2005), we find a parameter set for the GL system which reflects the a-priori information on the demand system (see also Witzke and Britz 1996).

Supply

In order to parameterize the CES production function, we estimate the cost shares from the given farm data in a Bayesian framework based on assumed cost shares for each activity. Actual costs shares need to exhaust given revenues plus reported costs at farm level:

```
*
*  --- value of input use at farm level, from individual activities
*
    sum( (rn_cl(rnCur,clCur),xxAgr) $ ( (p_data(rnCur,clCur,"","prodQuant",xxAgr,"cur")
                                         or sameas(xxAgr,"fodd"))
                                                     $ p_costShare(xxAgr,inputs)),

         (     p_data(rnCur,clCur,"","prodQuant",xxAgr,"cur")
           + v_foddOutput(rnCur,clCur) $ sameas(xxAgr,"fodd"))

             * p_data(rnCur,clCur,"","priceProdInc",xxAgr,"cur")

                * v_costShares(rnCur,clCur,xxAgr,inputs))

        =E=  (    p_inputUseFarm(rnCur,clCur,inputs)
               +  v_foddOutput(rnCur,clCur) $ sameas(inputs,"fodd" ))
                   * v_inputPriceR(rnCur,clCur,inputs);
```

Where *p_data* is given (production quantities, input use and producer incentive prices) and aggregated from individual farms to clusters, and cost shares as well input prices are to be estimated. As specific is fodder output which is assumed as non-tradable, such that the production value is estimated endogenously:

```
e_rev(rnCur,clCur,xxAgr) $ (p_data(rnCur,clCur,"","prodQuant",xxAgr,"cur")
                               or sameas(xxAgr,"fodd")) ..

  v_rev(rnCur,clCur,xxAgr) =E=      (    p_data(rnCur,clCur,"","prodQuant",xxAgr,"cur")
                                       + v_foddOutput(rnCur,clCur) $ sameas(xxAgr,"fodd"))
                                         * p_data(rnCur,clCur,"","priceProdInc",xxAgr,"cur");
```

A special case is land use where observed acreages need to be maintained:

```
*
*  --- land use is fixed to observation
*
  e_landUse(rnCur,clCur,xxAgr,landInp) $ p_landUse(rnCur,clCur,xxAgr,landInp) ..

      p_landUse(rnCur,clCur,xxAgr,landInp) *  v_inputPriceR(rnCur,clCur,landInp)

        =E= (  p_data(rnCur,clCur,"","prodQuant",xxAgr,"cur")
             + v_foddOutput(rnCur,clCur) $ sameas(xxAgr,"fodd"))

             * p_data(rnCur,clCur,"","priceProdInc",xxAgr,"cur")

                 * v_costShares(rnCur,clCur,xxAgr,landInp)  $ p_costShare(xxAgr,landInp);
```

The supply behavior is driven by a larger extent by the assumed substitution elasticities:

```
p_sigmaNest(rn,tNest,xxAgr)   = 0.25;

p_sigmaNest(rn,"feed_fodd",xxAgr) = 0.75;
p_sigmaNest(rn,"feed",xxAgr)      = 2;
p_sigmaNest(rn,"top",anim)        = 0.1;
```

As seen below, the top nest combines an aggregate of labor and capital with intermediate inputs, land and all type of feed, with a low substitution elasticity of 0.1. The feed subnest combines non-tradable fodder and concentrate feed with a substitution elasticity of 0.75, while concentrates can be substituted against each other with an elasticity of 2.0. Capital and labor can be substituted with an elasticity of 0.25.

```
tNest_n_a("top","feed_fodd",rum)        = YES;
tNest_n_a("top","feed",non_rum)         = YES;
tNest_n_a("top","cap_lab",anim)          = YES;
tNest_f_a("top","inpe",anim)             = YES;

tNest_n_a("top","cap_lab",rest)          = YES;
tNest_n_a("top","land_I",rest)           = YES;

tNest_f_a("cap_lab","cap",xxAgr)         = YES;
tNest_f_a("cap_lab","lab",xxAgr)         = YES;

tNest_f_a("land_i",landinp,rest)         = YES;
tNest_f_a("land_i","inpe",rest)          = YES;

tNest_n_a("feed_fodd","feed",rum)       = YES;
tNest_f_a("feed_fodd","fodd",rum)       = YES;

tNest_f_a("feed",feedi,anim)             = YES;
tNest_f_a("feed",feedi,anim)             = YES;
```

Factor exchanges between different agricultural activities are set based on the following transformation elasticities:

```
*
* --- calibrate factor supply, supply in the farm
*     (INPE has a fixed exogenous price, gras is only used for fodder production)
*
 p_CETElasXX(rn,"cap")                     =  1.0;
 p_CETElasXX(rn,"lab")                     =  2.0;
 p_CETElasXX(rn,"arab")                    =  5.0;
 p_CETElasXX(rn,"perm")                    =  0.5;
 p_CETElasXX(rn,"gras")                    =  eps;
```

Whereas exchange of land types between farms is relative elastic:

```
*
* --- supply of land between farms
*
 p_CETElasCL(rn,primFacI)                  =  2.0;
```

The factor supply elasticities are all inelastic, and quite low for land assuming that land-use cover change is governed by strong institutions:

```
*
* --- land supply elasticities are relatively small
*
   p_elasPrimFacOffer(rn,"arab") = 0.10;
   p_elasPrimFacOffer(rn,"gras") = 0.20;
   p_elasPrimFacOffer(rn,"perm") = 0.05;
*
* --- higher elasticities for labor and capital
*
   p_elasPrimFacOffer(rn,"lab")    = 0.60;
   p_elasPrimFacOffer(rn,"cap")    = 0.75;
```

Quantifying premiums at single farm level and simulating structural changes

Quantifying premiums at single farm level

Based on a code implementation close to the law book, we determine the different payments schemes which are one major element of the Norwegian agricultural policy. A specific aspects of several schemes is degressivity, i.e. the larger the farm, the lower the per unit support. Let *paymrate* denote the per unit subsidy paid in a certain period %1, payment scheme *stepPay* and production activity *poact*, and a limit level *step*. A seen from the statement below, if the production level of a single farm *curFarm* in region *rn* exceeds the current step level, it receives the payment for these first units in full. It if exceed the step level, only the fraction above step level is accounted for. In order to understand the logic, imagine a scheme where the first 10 units receive 100 € each, and any unit 50 € each. A farm producing 11 units will hence receive 10 x 100 + (11-10) * 50 €.

```
 paymentPerType(rn,curFarm,stepPay,poact,%1)
   $ (p_result(rn,"",curFarm,"levl",poact,%1) $ rn_f(rn,curFarm) $ payComb(stepPay,poact,""))
      = sum(stepp(step) $ paymstep(%1,stepPay,poact,step),
              [
*
*               --- fully included if activity level of farms exceed the current step threshold
*
                  (paymstep(%1,stepPay,poact,step)      - paymstep(%1,stepPay,poact,step-1))
                $ ( paymstep(%1,stepPay,poact,step) le p_result(rn,"",curFarm,"levl",poact,%1))
*
*               --- difference to next lower step included if above the threshold
*
              +  (p_result(rn,"",curFarm,"levl",poact,%1)- paymstep(%1,stepPay,poact,step-1))
                $ ((paymstep(%1,stepPay,poact,step) gt p_result(rn,"",curFarm,"levl",poact,%1))
                $ (paymstep(%1,stepPay,poact,step-1) le p_result(rn,"",curFarm,"levl",poact,%1))) ]
                        * paymrate(%1,stepPay,poact,step,rn));
```

The program also reflects that the legislation additionally stipulates maximum amounts in certain scheme a single farm can receive, in which case the premiums under that scheme are proportionally cut to stay under the ceiling:

```
*
* --- proportionally cut all activity specific payments of a payment type if
*     maximal support per farm is exceeded
*
  paymentPerType(rn,curFarm,paymtype,poact,%1) $ ( (paymsum(rn,curFarm,paymtype,%1)
                                                        gt paymmax(%1,paymtype))
                                                    $ payComb(paymType,poact,"")
                                                    $ rn_f(rn,curFarm))
     = paymentPerType(rn,curFarm,paymtype,poact,%1)
       * paymmax(%1,paymtype) / paymsum(rn,curFarm,paymtype,%1);
```

Furthermore, the law might prescribe that the premiums are cut unconditional by a certain amount, again, the resulting reduction is proportionally distributed to the different activities.

```
  paymentPerType(rn,curFarm,paymtype,poact,%1)
   $ (    paymdeduct(%1,paymtype)
       $ paymsum(rn,curFarm,paymtype,%1)
       $ paymentPerType(rn,curFarm,paymtype,poact,%1) $ rn_f(rn,curFarm))
   =  Max(0,paymentPerType(rn,curFarm,paymtype,poact,%1)
        - paymdeduct(%1,paymtype) * paymentPerType(rn,curFarm,paymtype,poact,%1)
                                  / paymsum(rn,curFarm,paymtype,%1));
```

Some of the premiums are paid to animals such as dairy cows which produce multiple outputs. Therefore, in order to convert them into a per-unit subsidy, distribution factors are used:

```
p_data(rn,cl,paymtype,"couplPaym","milk",%1) $ rn_cl(rn,cl)
  = sum(rn_cl_f(rn,cl,curFarm) $ p_lsuHerd(curFarm,"sum",%1),   paymentPerType(rn,curFarm,paymtype,"cmlk",%1)
                                                         +          paymentPerType(rn,curFarm,paymtype,"gmlk",%1)
                                                         +   2/3*   paymentPerType(rn,curFarm,paymtype,"goat",%1)
                                                         +   4/5 *  paymentPerType(rn,curFarm,paymtype,"dcow",%1)
                                                         +          paymentPerType(rn,curFarm,paymtype,"gras",%1)
                                                                      * (4/5*p_lsuHerd(curFarm,"dcow",%1)+2/3*p_lsuHerd(curFarm,"goat",%1))/p_lsuHerd(curFarm,"sum",%1));
```

Estimating changes at single farm level

The model operates at the level of farm group, so-called clustered, which were defined based on statistical analysis. In order to re-calculate the premiums and estimate farm exits, we map the simulated changes in output quantities and land use in relative terms into the single farms. That is a rather simplistic approach often applied in study combined macro- and micro-simulation(cf. Bourguignon et al. 2008 for use in macro-micro simulations for general or Deppermann et al. 2016 for an application to single farms), which seems however defendable given the relatively high level of differentiation in our farm group approach.

The mapping changes the production quantities and other variables for a single farm *curFarm* in region *rn* based on change in simulated production quantities in the farm group cluster *cl* it belongs:

```
p_result(rn,"",curFarm,"prodQuant",xxAgr,years) $ (p_result(rn,"",curFarm,"prodQuant",xxAgr,years-1) $ rn_f(rn,curFarm))
   = p_result(rn,"",curFarm,"prodQuant",xxAgr,years-1)
         * sum(rn_cl_f(rn,cl,curFarm), p_result(rn,cl,"","prodQuant",xxAgr,years)
                                     /p_result(rn,cl,"","prodQuant",xxAgr,years-1));
```

Change in labor and capital costs reflect simulated changes in production output level and cost shares:

```
p_result(rn,"",curFarm,"levl","cap",years) $ (p_result(rn,"",curFarm,"levl","cap",years-1) $ rn_f(rn,curFarm))
  = p_result(rn,"",curFarm,"levl","cap",years-1)
         *  sum((rn_cl_f(rn,cl,curFarm),xxAgr) $ p_result(rn,"",curFarm,"prodQuant",xxAgr,years),
        p_result(rn,"",curFarm,"prodQuant",xxAgr,years)*p_result(rn,cl,"X",xxAgr,"cap",years))
         / sum((rn_cl_f(rn,cl,curFarm),xxAgr) $ p_result(rn,"",curFarm,"prodQuant",xxAgr,years-1),
             p_result(rn,"",curFarm,"prodQuant",xxAgr,years-1)*p_result(rn,cl,"X",xxAgr,"cap",years-1));
```

Relative to the originally estimated labor and capital costs of each single farm. From there, it is possible to estimate the farm profits, given that the detailed premiums are calculated for each farm as well:

```
p_result(rn,"",curFarm,"levl","profit",years)
    =  p_result(rn,"",curFarm,"levl","reve",years)
     + p_result(rn,"",curFarm,"levl","paym",years)
     - p_result(rn,"",curFarm,"levl","cost",years);
```

Structural change

Currently, a simple algorithm is in use which uses a profit cutoff-level which depends on the farm size in hectares. The sample comprises a larger number of relatively small farms with low profits, often involved in sheep farming and having horses. Applying profit cut-off based on a comparison with factor returns out-of-agriculture will not reflect that intrinsic motive might determine to a larger extent the decision to farm. That let us linearly increase the cut-off level up to a 20 ha beyond which characterizes more commercial farms in the Norwegian context. We hence determine that farm specific minimum profit level:

```
*
*  --- determine farm specific profit cut-off level, depends on siz ein ha
*
   p_result(rn,"",curFarm,"levl","ProfitLim",years) $ ((p_result(rn,"",curFarm,"levl","ha",years) > 20) $ rn_f(rn,curFarm))
      = %minProfit20%;

   p_result(rn,"",curFarm,"levl","ProfitLim",years) $ ((p_result(rn,"",curFarm,"levl","ha",years) < 20) $ rn_f(rn,curFarm))
      = %minProfit0% + p_result(rn,"",curFarm,"levl","ha",years)/20 * (%minProfit20%-(%minProfit0%));
```

Next, we allow for different share of capital costs being accounted for, assuming that the remaining difference between revenues plus premiums minus variable costs will remunerate labor and land, assumed to be owned by the farming family:

```
*
*  --- if returns to labor and land are below threshold, remove farm
*
   dropFarm(curFarm) $ (sum(rn_f(rn,curFarm), p_result(rn,"",curFarm,"levl","profit",years)
                                              -p_result(rn,"",curFarm,"levl","cap",years)*%capCostPercent%/100
                                              -p_result(rn,"",curFarm,"levl","ProfitLim",years)) lt 0) = yes;
```

Next, we need to map these changes back to the cluster level. In order to do so, we first calculated the change in the production of the cluster by subtracting from the current output level the output of the exiting farms, in relation to current cluster output:

```
option kill=p_change;
p_change(rnCur,clCur,xxAgr) $ p_result(rnCur,clCur,"","prodQuant",xxAGR,Years)
   = (p_result(rnCur,clCur,"","prodQuant",xxAGR,Years)
          - sum(rn_cl_f(rnCur,clCur,dropFarm), p_result(rnCur,"",dropFarm,"prodQuant",xxAgr,years)*0.001))
         / p_result(rnCur,clCur,"","prodQuant",xxAGR,Years);

display p_change;
p_change(rnCur,clcur,xxAgr) $ ((p_change(rnCur,clcur,xxAgr) le 1.E-10) $ p_change(rnCur,clcur,xxAgr)) = eps;
```

Given that the clusters operated under constant-return-to-scale, we can use that correction factor to correct input use:

```
*
*  --- correct outputs/input demands according to change
*
   v_xtNest.l(rnCur,clCur,tNest,xxAgr) $ (v_xtNest.l(rnCur,clCur,tNest,xxAgr) $ p_change(rnCur,clCur,xxAgr))
    = v_xtNest.l(rnCur,clCur,tNest,xxAgr) * p_change(rnCur,clCur,xxAgr) $ (p_change(rnCur,clCur,xxAgr) gt eps);

   v_inputUse.l(rnCur,clCur,inputs,xxAgr)  $ (v_inputUse.l(rnCur,clCur,inputs,xxAgr) $ p_change(rnCur,clCur,xxAgr))
    = v_inputUse.l(rnCur,clCur,inputs,xxAgr)  * p_change(rnCur,clCur,xxAgr) $ (p_change(rnCur,clCur,xxAgr) gt eps);

   p_result(rnCur,clCur,"","prodQuant",xxAGR,Years)  $ p_change(rnCur,clCur,xxAgr)
     = p_result(rnCur,clCur,"","prodQuant",xxAGR,Years) * p_change(rnCur,clCur,xxAgr)
          $ (p_change(rnCur,clCur,xxAgr) gt eps);
```

That also implies the calibrated costs shares are not affected. In order to get a feedback on the simulation behavior, we recalibrate the CET-factor supply nests such that the reduced input demand quantities would be demanded at the old input prices, such as:

```
*
*  --- re-calibrate share parameters
*
   p_shareParXX(rnCur,cl,primFacI,xxAgr) $ (v_inputUse.l(rnCur,cl,primFacI,xxAgr) and p_cetElasXX(rnCur,primFacI))
       = v_inputUse.l(rnCur,cl,primFacI,xxAgr)/v_inputUseCL.l(rnCur,cl,primFacI)
               * ( v_inputPrice.l(rnCur,cl,primFacI,xxAgr)/v_inputPriceCL.l(rnCur,cl,primFacI))**(-p_cetElasXX(rnCur,primFacI));
```

That implies that the feedback will happen via the factor supply functions which are unchanged. With reduced factor demand, factor prices such as land prices will drop, and the remaining farm cluster will face lower prices which will lead to both adjustments in factor use in the remaining farms and consequently to changes in outputs and output prices.

Technical implementation

Similar to most other tools in that field (e.g. Britz and Kallrath 2012), we use GAMS (General Algebraic Modelling System, Bischopp and Meeraus 10082) to encode the model and all necessary data and parameter transformation. Coding conventions, e.g. with regard to a modular structure and use of mnemonics, follow guidelines developed for CAPRI (Britz 2010). For the Bayesian based parameter calibrations, we use CONOPT (Drud 1994, version 2014), the market model is solved in PATH (Ferris and Munson 2000) as an MCP. A Graphical User Interface allows steering the model and results exploitation, is it implemented GGIG (GAMS Graphical Interface Generator, Britz 2014).

A exemplary application

In order to

References

Anania, G. (2006). The 2005 WTO arbitration and the new EU import regime for bananas: a cut too far? European Review of Agricultural Economics, 33(4), 449-484.

Banse, M., Grethe, H., & Nolte, S. (2005). European simulation model (ESIM) in GAMS: model documentation. Model documentation prepared for DG AGRI, European Commission, Göttingen and Berlin.

Bisschop, Johannes; Meeraus, Alexander. On the development of a general algebraic modeling system in a strategic planning environment. Springer Berlin Heidelberg, 1982.

Britz W. (2013). Using price sensitivity analysis with the supply models to parameterize the market model. Technical Paper, University Bonn, http://www.capri-model.org/dokuwiki/..%5Cdocs%5Cprice_sens_market_model.pdf

Britz, W. (2010), The red book on CAPRI GAMS coding. Deliverable: D7.1 of CAPRI-RD (Common Agricultural Policy Regionalised Impact – The Rural Development Dimension), University Bonn, http://www.ilr.uni-bonn.de/agpo/rsrch/capri-rd/docs/d7.1.pdf

Britz, W. (2014). A New Graphical User Interface Generator for Economic Models and its Comparison to Existing Approaches. German Journal of Agricultural Economics, accepted

Britz, W. and Witze, P. (2014) (edts). CAPRI model documentation, version 2014. University Bonn, http://www.capri-model.org/docs/capri_documentation.pdf

Britz, W., & Heckelei, T. (2008, January). Recent developments in EU policies–challenges for partial equilibrium models. In 107th EAAE Seminar Modeling of Agricultural and Rural Development Policies.

Britz, W., & Kallrath, J. (2012). Economic Simulation Models in Agricultural Economics: The Current and Possible Future Role of Algebraic Modeling Languages. In Algebraic Modeling Systems (pp. 199-212). Springer Berlin Heidelberg.

Devadoss S., Westhoff P.C., Helmar M.D., Grundmeier E., Skold K.D., Meyers W.H., Johnson R.S. (1989): The FAPRI Modeling System at CARD: A Documentation Summary, Iowa State University, Food and Agricultural Policy Research Institute Technical Report 13, December 1989

Diewert, W. E., & Wales, T. J. (1988). A normalized quadratic semiflexible functional form. Journal of Econometrics, 37(3), 327-342.

Drud, A. S. (1994). CONOPT—a large-scale GRG code. ORSA Journal on Computing, 6(2), 207-216.

EU 2009. Part III: Annexes to Impact Assessment Guidelines, http://ec.europa.eu/smart-regulation/impact/commission_guidelines/docs/ia_guidelines_annexes_en.pdf

Ferris, M. C., & Munson, T. S. (2000). Complementarity problems in GAMS and the PATH solver. Journal of Economic Dynamics and Control, 24(2), 165-188.

Havlik, P., Schneider, U. A., Schmid, E., Böttcher, H., Fritz, S., Skalský, R., ... & Obersteiner, M. (2011). Global land-use implications of first and second generation biofuel targets. Energy Policy, 39(10), 5690-5702.

Heckelei, T., Mittelhammer, R., & Britz, W. (2005, February). A Bayesian alternative to generalized cross entropy solutions for underdetermined models. In 89th EAAE Seminar, February (pp. 3-5).

McCarl, B. A., & Spreen, T. H. (1980). Price endogenous mathematical programming as a tool for sector analysis. American Journal of Agricultural Economics, 62(1), 87-102.

Nolte, S. (2008). The future of the world sugar market: a spatial price equilibrium analysis. In 107th European Association of Agricultural Economists (EAAE) Seminar: Modelling Agricultural and Rural Development Policies.

OECD, 2007. Documentation of the AGLINK-COSIMO Model. Organization for Economic Co-operation and Development: Paris

Paris, Q., Drogué, S., & Anania, G. (2011). Calibrating spatial models of trade. Economic Modelling, 28(6), 2509-2516.

Podhora, A., Helming, K., Adenäuer, L., Heckelei, T., Kautto, P., Reidsma, P., ... & Jansen, J. (2013). The policy-relevancy of impact assessment tools: Evaluating nine years of European research funding. Environmental Science & Policy, 31, 85-95.

Roningen V.O., Sullivan, J. and Dixit P. (1991): Documentation of the Static World Policy Simulation (SWOPSIM) Modeling Framework. Staff Report AGES 9151. Washington, DC: ERS, USDA.

Ryan, D. L., & Wales, T. J. (1999). Flexible and semiflexible consumer demands with quadratic Engel curves. Review of Economics and Statistics, 81(2), 277-287.

Wieck, C., Schlüter, S. W., & Britz, W. (2012). Assessment of the Impact of Avian Influenza–related Regulatory Policies on Poultry Meat Trade and Welfare. The World Economy, 35(8), 1037-1052.

Witzke, H. P., & Britz, W. (1998). A maximum entropy approach to the calibration of a highly differentiated demand system (pp. 98-06). Capri Working Paper 98-06, University of Bonn.

Armington, P. S. (1969). A Theory of Demand for Products Distinguished by Place of Production. Staff Papers-International Monetary Fund, 159-178

F. Bourguignon, M. Bussolo, L. Pereira da Silva (Eds.), The impact of macroeconomic policies on poverty and income distribution, The World Bank, Oxford University Press, New York, NY (2008)

Deppermann, Andre, Frank Offermann, and Harald Grethe. "Redistributive effects of CAP liberalisation: From the sectoral level to the single farm." Journal of Policy Modeling 38.1 (2016): 26-43.