Methods and software tools for agro-ecological evaluation at farm level: the livestock component within the SIPEAA project

M. Speroni, A. Annunziata, E. Basile, G. Pirlo

Istituto Sperimentale per la Zootecnia, Cremona, Italy

Corresponding author: Marisanna Speroni. Istituto Sperimentale per la Zootecnia. Via Porcellasco 7, 26100 Cremona, Italy – Tel: +39 0372 43029 – Fax: +39 0372 435056 – Email: marisanna.speroni@isz.it

INTRODUCTION

Livestock farms need to adapt to social and economics changes. Often, farmers must take decision in order to: - remain competitive, meanwhile market and public aids change; - reduce impact on environment. Improvement of efficiency seems to be the only way to conciliate these two different needs. Livestock farm are very complex systems involving physical and biologic components. These components interact each other and with external environment so that management consequences are not easily predictable. Experiments give support to decisional processes with a specific forage system or under specific weather or economic conditions, but they take long time to be carried out. Simulation models can be very useful in order to accelerate the knowledge process and they can provide decision support tools in less time. Softwares exist that simulate production, storage and distributions of feedstuffs and manure, that predict milk yield and growing process. Some systems integrate more of these tools but generally they are not flexible or not well documented. Ministry of Agricultural and Forestry Policies (MIPAF) granted a multidisciplinary project having the aim of produce a system easy in handle and maintenance, expandable and transparent in contents: the SIPEAA (Software Tools for Eco-compatible Farm Planning) project (http://www.sipeaa.it, Donatelli et al., 2002). First, a SIPEAA Farm ontology (FO) (Mazzetto et al., 2004) was developed in order to reach an agreement about the relevant concepts of a farm firm. According to the FO, the whole behaviour of the farm is the result of the interactions among the decision taken at different levels (strategic, management and operational) in different times; farm is characterised by sets of production technologies (sub systems characterised by a specific main product, e.g. a crop, a type of livestock), resources (assets used and released by actions, e.g. land, building, plants), materials (assets consumed by actions e.g. milk, seeds) and actions. The SIPEAA software architecture includes databases, a Farm Configuration Builder (FCB), a Farm Simulator (FS) (Blocchetti et al., 2004), an Agricultural Production and
Externality Simulator (APES), a Farm Configuration Evaluator (FCE). From databases or from user input, the FCB produces farm configuration, i.e. each set of production technologies, resources, material and actions we want simulate or evaluate. The FS allows the biophysical simulation and handles events. The APES will link specific model components which actually perform the simulation of biophysical processes and will provide them common services (e.g. integrator, time handler, data store). Specific model components were developed for weather, machinery, soil, crops, pesticides, and livestock. FCE will perform an evaluation of farm performances based on a multicriteria approach. The linkage and flow of information for the overall SIPEAA system will be widely described in a final project document. The current paper reports some results about the model livestock component, simulating dairy herd in intensive free stall system (Dairy Herd) and its structure design.

MATERIALS AND METHODS – Guidelines for the SIPEEA components development are summarized in Donatelli et al. (2004). According to those basic concepts, we must separate parameters (quantities that will remain constant during simulation) from state variables (variables that describe the state of the system and which change during simulation). DairyHerd has been developed paying attention to: - encapsulate the handling of parameters since they do not represent attributes of the SIPEEA system; - use interfaces to access to state and rate variables via the system control; individual components do not communicate directly among them, in order to reduce dependency between components; - separate interfaces from implementation, to enhance reusability; - use the same meta-data structure that the other components will use.

RESULTS AND CONCLUSIONS – Dairy Herd Model component. The conceptual model of DairyHerd can be represented as a flow of energy and nutrients throughout an animal resource (dairy herd) changing in mass and producing milk. The dairy herd resource is described by 7 animal groups: calves from birth to 90 days, female calves under one year old, heifers over one year old, heifers over the last three months in pregnancy, primiparous lactating cows, pluriparous lactating cows, dry cows. Input parameters for all groups are DMI, crude protein content in diet (CPdiet), body weight (BW), crude protein in BW (CPbody) and P content in body weight of each group; lactating categories are described also using potential Milk Yield (MY), potential milk fat content, potential milk crude protein content (CPmilk), urea nitrogen in milk (MUN), and days in milk (DIM).

The feedstuff material is defined by the content in DM, CP, TDN, ENI, NDF, and P. The output material, milk and body mass, are defined by: milk sold; crude protein, fat and somatic cells contents in the sold milk; CP content in body weight gain (CP gain) of each group. Actions are: formulating diets, milking, registering calving births and deaths. DairyHerd contains two main sub-components that simulate: animal requirements (Requirements), nitrogen excretion (NitrogenExcretion) and some tools-subcomponent (auxiliary subcomponents) providing parameters for the main sub-components. Implementation of a sub-component simulating the formulation of diets (Feeding) is still in progress. The Requirements subcomponent predicts the average milk yield (predMY) and the average DMI (predDMI) for the rolling herd groups. It also estimates the average requirements in metabolisable protein (reqMP), NE\textsuperscript{L} (reqNEL) and P (reqP) for each group except that for calves from birth to 90 days. A classical hierarchy (from gross to digestible, to metabolisable, to net energy) and a classical partitioning of feed energy (into growth, maintenance and milk) were retained. Computations are based upon the empirical equations derived from NRC (2001). The NitrogenExcretion sub component allows the computation of the nitrogen excreted yearly as difference between nitrogen ingested yearly and nitrogen produced yearly as milk and body weight. Nitrogen ingested yearly is calculated for each group on the basis of DMI, CPdiet, and days of staying in the group (Days); for DMI and CPdiet of each category default values are provided, based on literature data; Days for each groups are constants; the total output for the herd is computed as sum on the basis of number of animal in each group. Nitrogen produced yearly in BW can be calculated as sum of CPbody and BW; limitedly to Friesian breeds, CPbody and BW default values for each group are provided.
Default data are based on literature and experimental data. Each group is used to compute the average output and total output for the herd as computed as sum on the basis of number of animal in each group. Nitrogen produced yearly in milk is computed from MY and CPmilk; MY can be directly input by user or it can be estimated using the milk sold value, that is an information usually available in all farm, adding the estimate colostrum yield and the estimate amount of milk discharged for sanitary reasons. NitrogenExcretion also allows the prediction of urine or fecal nitrogen production as functions of MUN or MUN, MY and CPmilk.

*Dairy Herd Component structure and decomposition.* Three different kind of object form the frame of each DairyCows sub-components: data, interface and simulation. Data objects are independent either from the component that uses them either from the SIPEAA system. The data object allows to document variables specifying attributes like max value, min value, default value, unit, description and type. Interfaces contains the terms of the contract that the component must comply in order to work in the system. They say how the component must be used, not how they work. The simulation objects perform specific calculations. The choices of separating data object from method components as well the use of interfaces have been crucial in order to achieve reusability of model components and sub-components and the system modularity. This strategy will allow increasing complexity of the DairyHerd adding sub-component and substituting the existing sub-component.

**ACKNOWLEDGMENTS** – Research under the auspices of MIPAF, Paper SIPEAA n° 29.

**REFERENCES** – Bellocchi, G., Donatelli, M., Fila, G., Bechini, L., Monti, C., 2004. Designing a farming system simulator. Proc. VIII ESA Congress, Copenhagen, 579-580. Donatelli, M., Acutis, M., Danuso, F., Mazzetto, F., Naselli, P., Nelson, R., Omicini, A., Speroni, M., Trevisan, M., Tugnoli, V., 2002. Integrated procedures for evaluating technical, environmental and economical aspects in farms: the SIPEAA project. Proc. VII ESA Congress, Cordoba, 271-272. Donatelli, M., Omicini, A., Fila, G., Monti, C., 2004. Targeting reusability and replaceability of simulation models for agricultural systems. Proc. VIII ESA Congress, Copenhagen, 237-238. Mazzetto, F., Sacco, P., Bonera, R., 2004. A farm ontology for farm modelling. Proc. VIII ESA Congress, Copenhagen, 249-250. National Research Council, 2001. Nutrient requirement of dairy cattle. 7th revised edition. National Academy Press, Washington, DC, USA.