EXTENSIONS AND REFINEMENTS OF THE DYNAMIC COW DIGESTION MODEL - MOLLY AND FURTHER DEVELOPMENT OF THE MODEL FOR EDUCATIONAL USE

Sven Smårs

Licentiate Thesis

Institutionen för lantbruksteknik

Swedish University of Agricultural Sciences
Department of Agricultural Engineering

Rapport 239
Report 239

Uppsala 1999
ISSN 00283-0086
ISRN SLU-LT-R-239-SE

ABSTRACT

This thesis is composed of two parts. Part I deals with the extension and refinement of the cow digestion model Molly and Part II describes how the extended model was transformed into a more educational form.

The purpose of **Part I** was to model the dynamics of the metabolic process from fodder to milk in the cow under Swedish conditions and under different management situations. After studying a number of models, it was decided to use the model "Molly" developed for U.S. conditions by Baldwin et al (1977; 1987a,b,c).

However, Swedish conditions differ from those in the U. S. in the sense that it is common practice in Sweden to schedule the feeding of forage and concentrate separately within a day. Separate parameters and facilities for scheduling the intake of four different feeds had to be included. A consequence of the more complex Swedish feeding conditions was that the model had to be extended to include four separate sets of physical and chemical feed descriptions instead of one. Therefore, parts of the rumen submodel were separated into four parallel submodels to allow for different degradation rates depending on source of feed.

The structure and a description of the model from an overview down to essential details are also treated in this part.

Finally, the differences in metabolism between the original U.S. Molly and the refined model under Swedish conditions were analysed in detail.

In **Part II**, the purpose was to transform the extended model into a form which could be used for teaching and training. The reason for that was that the original model, written in the simulation language ACSL, is far too complicated to be handled for educational use.

In operational terms the aims defined were: 1) The model must be hierarchically structured so that the whole process can be monitored and that the user can zoom into more detailed levels. 2) The model must be presented in a graphical form based on states and flows rather than in computer code. 3) Experimenting with the model by setting appropriate values to various parameters and initial conditions must be made in a user-friendly way without the need for entering programme code and recompilation. 4) The computer environment must support easy and flexible presentation of results in diagrams and figures. 5) The computer environment must have tools for handling pre-defined scenarios by using macro capabilities.

MATLAB/Simulink was chosen as the best platform for the Molly digestion model in cow, mainly because of its graphical tools and its capacity to handle hierarchical structures. This part of the thesis also shows how to use the educational model.

PREFACE

This work builds on the 1993 version of the cow digestion model Molly developed and published by Professor R. Lee Baldwin and colleagues during the years 1975-1995. The current work started with a co-operative project to model dairy feeding and production at cow and farm level initiated in 1991 by Åke Axenbom, from whom the author took over when he left the project in 1992. Participants in that co-operation were representatives of the Alfa-Laval company, the Department of Animal Nutrition and Management and the Department of Agricultural Engineering at the Swedish University of Agricultural Sciences.

Another project to adapt the Molly model for use in undergraduate education was started in 1993 as money was given for this purpose by the Swedish Council for Renewal of Undergraduate Education.

This thesis consists of two parts. Part I describes development of the model Molly and extension of the rumen submodel for the purpose of making it possible to use the model to predict effects of different Swedish common feeding routines. The model presented in Part I is the result of the co-operative project started in 1991. Part II is the result of the project to adapt the Molly model for use in undergraduate education. This part describes how the refined model was translated and visualised in another computer simulation language in order to improve the overview and transparency of the model for use in education.

Acknowledgement: I want to thank Alfa-Laval and the Swedish Council for Renewal of Undergraduate Education who founded this work. Thanks to Professor Thomas Nybrant for his help in keeping me on the track until this work was finally written. Thanks to Associate Professor Peter Udén for all the expert knowledge on the metabolism in the dairy cow with which he has assisted me through my work. Thanks also to Associate Professor Leif Gustafsson for encouraging and assisting me with technical advise on how to give this thesis its final format. I also want to thank my wife and family for their moral support during this work.

I

PART I - EXTENSIONS AND REFINEMENTS OF THE DYNAMIC COW DIGESTION MODEL - MOLLY

CONTENTS

1. INTRODUCTION

The way in which dairy cows are managed in Sweden has changed considerably during the past century. The industrialisation and urbanisation of the country has made people abandon the custom of keeping only one or a few cows and distributing the milk to local customers. Harvesting of feeds, milking and handling of manure have to a large extent been mechanised but the feeding of the cows has been mechanised only to a limited extent. Progress in breeding, together with improved fodder quality and better management, has led to an increased milk production per animal and also better feed utilisation. Understanding of the way the food is utilised in the animal has been useful in the improvement of dairy cow nutrition and management of feed crops. In order to more fully understand the complex system of metabolic pathways by which different nutrients are utilised, a systematic approach and the use of some kind of model is necessary.

Systems analysis is a general approach to analyse phenomena. It is obvious that anything under study can be regarded as a sub-system of a larger system. It is also obvious that a sub-system is a delimited part of a greater system and that it is delimited by a system boundary. The systems analyst should be aware of the boundaries of the system he choses to study and should try to include only the subsystems necessary to fulfil the objectives of the study. The objectives of the study are also crucial for the accuracy of data required, the necessary assumptions regarding the behaviour of the system, the choice of tools for analysis and time horizons of the study.

Different kinds of models have been used in order to understand different parts of reality ever since the time of the Ancient Greeks and probably even further back. Different models can be grouped in different model classes. According to Beever et al (1986), models can be classified as real, i.e. physical or abstract, i.e. mathematical. The models which will be dealt with here are mathematical. Beever et al (1986) classifies mathematical models as:

- dynamic or static
- deterministic or stochastic
- mechanistic or empirical

A *dynamic* model as opposed to a *static* model contains at least one relationship which can be described with a differential equation, usually with time as the independent variable. This means that the dynamic model contains a variable describing an cumulative amount of something that changes. The term *deterministic* as opposed to *stochastic* means that the relationships in the model are not dependent on randomness. The term *mechanistic* means that the relationships in the model are explained by natural laws, for example of a physical or chemical nature, whereas *empirical* means that they are based on empirical studies.

The choice of model class depends on several factors. Perhaps the most important of these is the objective of the study. Other factors are: availability of relevant data, skill and experience of the modeller.

Different types of models and systems have been used for feed evaluation and research on the subject. The current position is that virtually all systems used for practical feed evaluation and management in the world today are still based on static models. The tendency is, however, towards more and more sophisticated models and at least one of them, the AUSPIG decision support software, is built around an animal model which contains truly mechanistical and dynamic relations (Black et al, 1993). Another one, the Cornell Net Carbohydrate and Protein System, is a static model which to a large extent is mechanistic and has elements of flow kinetics in it (Russel et al, 1992).

The situation is different when it comes to research models. Here, models containing truly dynamic relations consisting of differential equations with time as the independent variable have been used for decades. These models often have qualities which may explain why they have not been adopted for practical use. Firstly, that they are usually not very user-friendly. Secondly, they may require data which is not available in the practical situation. A model may be good in predicting and explaining cause and effect within an objective of a study and still be inadequate in other aspects as a tool for management. However, qualities like these are inherited and have nothing to do with their being dynamic or mechanistic. Black et al (1993) give guidelines on how to make decision support software around mechanistic animal models which are likely to be adopted in practice. It is important to remember that while dynamic research models may not be adopted for practical use, they may still play an important role in increasing the knowledge and understanding of the metabolism in an animal.

Since larger farms in Sweden during recent years have started to adopt feed management systems in which total mixed rations are fed to groups of cows, it is desirable to be able to compare these systems with the traditional ones. In the traditional Swedish systems the cows are fed different types of concentrates separate from forage according to the estimated need of energy and protein for the individual cow. Concentrate is usually fed between 2-4 times and forage 1-2 times a day. The work presented here deals with simulation of effects of intakes of feeds from varying sources and times during the day.

2. OBJECTIVES

The objective of this study was to model the milk production of a cow. The model was to be used for studying the influence of feeding and milking management on the metabolism and milk production for cows under Swedish conditions and under different management situations. Therefore it had to be able to simulate the digestion of feeds, from varying sources with individual source-dependent degradation rates for appropriate nutrients, in situations where the cows were fed different concentrates at different discrete time periods over the day.

3. LITERATURE REVIEW

There have been several dynamic models of ruminant digestion during the three past decades.

Smith (1970) developed a biologically very detailed model of ruminant digestion. The model had eight or nine parts that represented different separate organs and an aggregate tissue with minor mechanisms inside. The model was written and run in a simulation language called Kinzym. The model was, however, over-parameterised i. e. an infinite number of parameter settings could produce the same responses (Baldwin, personal communication).

A new model of ruminant digestion for evaluation of factors affecting nutritive value was developed by Baldwin et al (1977). The objectives of this model were to model ruminant digestion for evaluation of the biochemical, microbial, physiological and chemical factors affecting feed utilisation. The model aimed to be quantitative and dynamic and was based on currently accepted and defensible concepts. The model also had to consist of theoretical rather than empirical equations. Another purpose was identification of aspects where current generally accepted concepts and/or data were inadequate. It also had to be possible to use the model to test hypotheses regarding factors affecting nutritive value. Fourteen interactive sub-units constituted the model, accommodating for the digestion of the twelve constituents: soluble carbohydrate, organic acids, starch, pectin, hemicellulose, cellulose, lipids, soluble protein, insoluble protein, non-protein nitrogen, lignin and ash.

The model was evaluated by studying the simulation results from a varied intake level and frequency of a reference lucerne chaff diet to a 40 kg sheep. The results indicated that the concepts of the model were appropriate for use in evaluation of biochemical, microbial, physiological, physiological, physical and chemical attributes of feeds to ruminant animals (Baldwin et al, 1977). It was also concluded that the factors in the model regulating the rate of passage out of the rumen and factors influencing microbial growth on low nitrogen diets were inadequate.

France et al (1982) constructed a rumen model consisting of nine state variables. These state variables were: Rumen metabolic volume, non-rumen degradable β-hexose, rumen degradable β-hexose, α-hexose, water soluble carbohydrates, non-protein nitrogen, rumen degradable protein, non-rumen degradable protein and micro organisms. The model inflow representing diet and saliva could be continuous or intermittent. Rumen outflow was compared with data on different diets fed to sheep. Reasonable agreement was obtained when the model was run in continuous mode. In intermittent mode, the discrepancies were considerable. Rumen pH was not explicitly modelled. This model was not suitable for use in this work because it was restricted to rumen functions and the ability to model degradation rates for at least cellulose as a function of rumen pH was a necessity with respect to our objective to simulate feeding of different concentrates at different discrete time periods over the day.

Baldwin and colleagues have published three articles on models related to the metabolism of a lactating cow. The first of these, Baldwin et al (1987a), describes a model of the animal elements of a cow. The model was structured and parameterised from data obtained from tissue level experiments conducted in vitro. This model consisted of ten state variables describing amino acids, acetic acids, fatty acids, glucose, lactose, protein in body, protein in viscera, protein in milk, triacyl glycerides in milk and triacyl glycerides in body. The first four of the above states represented metabolic pools and the remainder production pools (i. e. storage in body and milk). The model had an additional eleven intermediate "pools of zero size" (i. e. pools approximated to have input rates equal to output rates since their time constants are close to zero with respect to the resolution of time in the model) describing ATP, butyric acid, carbon dioxide, lactic acid in fat, lactic acid in lean body, lactic acid in viscera, oxygen, propionic acid, triose phosphate in fat and triose phosphate in viscera. Inputs to the model were the six nutrients: amino acids, acetic acids, butyric acid, fatty acids, glucose and propionic acid. In order to simulate continuous feeding, the input fluxes were held constant during each run. The model was evaluated by sensitivity and general behavioural analyses and by a simulated energy balance study. The results were presented and it was concluded that "concepts and data from tissue level experiments can be used to structure and parameterise whole animal models since the quantitative and dynamic behaviour of such a model is acceptable." These analyses also indicated that models of this type can be used "to evaluate factors which influence patterns of nutrient utilisation." The results from the energy balance study showed that simulated net efficiency of milk production agreed well with the net efficiency used in practice as obtained from empirical experiments. It was further concluded that the model could be helpful in the design and selection of experiments related to metabolism and energy balance. It was also concluded that this model could serve as a part in a larger model simulating the entire process of a lactating cow throughout the lactation cycle.

In the second article, Baldwin et al (1987b) describe a model of digestion within the rumen depending on the pattern of nutrient entry. The model was first built for use with a continuous inflow, and later adjusted for use with a pulsed entry of a total mixed ration. The model consisted of essentially twelve state variables describing physical, biological and chemical breakdown and dynamics of intermediate products of feed. The equations used for the kinetics of the model were mainly of either Michaelis-Menten or mass action type. The model was written in the computer simulation language ACSL and a fourth-order Runge-Kutta integration procedure was used for the numerical simulation. It was concluded from sensitivity and behavioural analyses that the stability and sensitivity of the model parameterisation were generally satisfactory both in continuous and discontinuous mode with a few exceptions. It was established that the description and parameterisation of aspects such as particle size in relation to availability, the particle outflow rate constants and affinity and rate constants for amino acid degradation needed improvement. The model behaviour in comparison with experimental data was considered satisfactory on forage-based and medium concentrate-containing diets but for diets containing more than 90% cereal the model predictions, especially for volatile fatty acids, were less satisfactory.

The third article, by Baldwin et al (1987c), described how the two models of digestion and metabolism presented above were merged together to serve as parts in one whole animal model by using the output flows in the form of absorbed nutrients in the digestion model as input flows to the model of metabolism. The model was further completed with a model of the udder metabolism adapted from Neal and Thornley (1983). A minor change was made to the model of metabolism,namely that constants regulating protein synthesis in lean body and viscera were replaced by equations describing protein synthesis as a function of the pools of protein in lean body and viscera. Quantitative and dynamic behaviour of mechanisms influencing partitions of nutrients within a day and energy balance over a full lactation were investigated in a number of simulations. The behaviour of the model was found to be realistic except for high concentrate diets.

Dijkstra (1993) published two rumen models focusing on mathematical representation of microbial metabolism and the degradation of neutral detergent fibre NDF by rumen protozoa. The two models consisted of 17 and 19 state variables, respectively. The second model was a further development of the first with the main objective of a more detailed modelling of rumen protozoa. The models meant improved mathematical representation of several microbial aspects. Protozoal substrate preference for nutrients such as starch and sugars compared with fibre and of insoluble compared with soluble protein was included in the model as was recycling of microbial matter represented by protozoal death and lysis related to nutrient availability. Chemical composition of rumen microbes (i. e. the carbohydrate content of the microbes) was allowed to wary with rumen availability of nitrogen and energy. Microbial activity and absorption of volatile fatty acids was allowed to change depending on rumen pH. Model predictions were compared to experimental observations. Predicted values showed reasonable agreement with observed values for fibre degradation and protozoal biomass. However, the predicted protozoal turnover time was less accurate. The models had the restriction that rumen pH was not predicted by the models and had to be supplied by the user. This limitation and the fact that the models were restricted to rumen functions made the models inappropriate for use as a starting point for the present work.

4. MODEL DESCRIPTION

At an early stage of the project it was decided to base our model mainly on the model by R. L. Baldwin (Baldwin et al, 1987c) which at that time was named Daisy, and in a later version Molly, by its creator. Since the main structure of the model for this project was taken from Baldwin's Molly, we decided to keep that name.

However, the Molly model described U. S. conditions when referring to feed composition and scheduling of feeding. We decided to replace these parts with corresponding Swedish conditions.

A consequence of the Swedish feeding routines was that different kinds of feeds had to be separately scheduled and that the rumen model had to be refined. These refinements are described in paragraphs 4.2 and 4.3.1.

Molly consists mainly of five entities: feed description, rumen digestion model, body tissue model, intestinal model and udder model. In addition to these parts there are algorithms implementing eating and milking patterns, depending on various feeding strategies and milking routines. An overview of the model is shown in Figure 1.

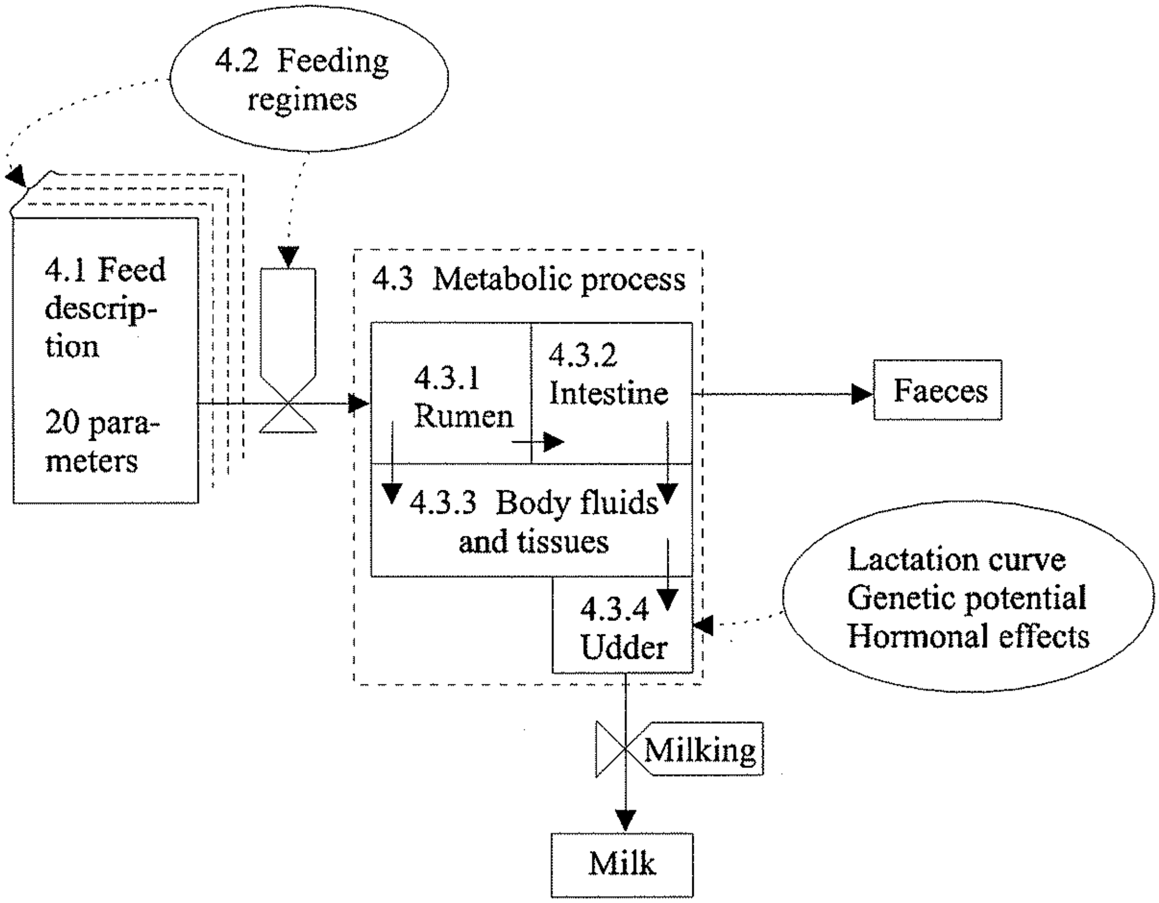

Figure 1. Overview of the model. Figure numbers refer to chapter and paragraphs.

4.1 Feed description

The feed is described by 20 feed attributes, of which two are physical parameters describing starch solubility and particle size. Starch solubility is the proportion of starch suspended in water which goes through a glass filter. The particle size factor is the proportion smaller than approximately 2 mm which can pass from the rumen. The remaining 18 feed attributes are variables assigned numbers describing the dry matter fractional content of different feed constituents, i. e. nutrients and ash. The feed constituents are: soluble carbohydrates, organic acids, pectin, lactate, lipids (vegetable), other fat, starch, hemicellulose, cellulose, soluble protein, insoluble protein, non-protein nitrogen, lignin, soluble ash, insoluble ash, acetate, butyrate and urea.

Since these last 18 parameters completely and exclusively cover the dry matter content of the feed and their numbers express the fraction of respective constituent in the interval zero to one, the sum of them should always be one. Mineral elements, vitamins, toxic and infectious agents are not included in the model.

4.2 Feeding regimes

The original cow model could only handle feeds in the form of complete mixed rations. Since such a feeding system is not very common in Sweden, it was decided to change the model so that it could handle feeding of forage and concentrate at different times during the day to adapt to Swedish practices.

It is not uncommon that farmers in Sweden handle two separate concentrates along with one or two types of forages. This makes it necessary to specify at least four types of feed separately. This was implemented so that an additional three sets of the feed specification parameters were added to the model.

The next problem was to administer the inflow of the four sets of feed mixtures. It was solved by creating a set of square pulses for each feed mixture, where content of each pulse represented one meal. The height of each pulse represented the rate of consumption of each particular meal, the duration of each pulse was equivalent to consumption time so the area of each pulse represented the amount of feed in each meal. By using these pulses it was possible to get a reasonable approximation of the consumption patterns of stall-tied cows fed concentrate and forage several times a day.

The administration of feed intake was implemented so that the model user can specify the amounts and consumption times for four meals per mixture of feed and day.

4.3 Metabolic model

The metabolic process in the cow model describes the degradation of the feed into nutrients and faeces occurring in the rumen and the intestine, the uptake of nutrients through the blood and body fluids and transformation into milk and body reserves such as viscera, meat and fat.

4.3.1 Rumen model

The largest part of the cow model is the rumen. The number of equations in the rumen outnumbers by far all other entities. An overview of the rumen model is shown in Figure 2. In this presentation the rumen model is divided into physical, microbial, chemical and water dynamic relations to be presented in separate paragraphs. Finally our refinement of the rumen to treat up to four different feeds simultaneously is treated.

Physical properties

When the feed enters the rumen it consists of a soluble fraction and of large and small particles. The nutrients from the large particles become available when the nutrients are hydrolysed as the large particles are broken down to small particles. In real life this process is mainly done by rumination and to a minor extent by fermentation. Small particles affected by the size reduction contain holocellulose (cellulose + hemicellulose), lignin, insoluble ash and insoluble protein.

In the model the large particle fraction of the feed, specified by the particle size factor, enters a storage pool which it leaves as it is broken down at a rate LpSp (large particle to small particle) :

$$LpSp = KLpSp*Lp*Rum$$

where

KLpSp = a constant (def value 4.5 d^{-1}),
Lp = large particle pool and
Rum = factor defining time spent ruminating.

The default value for rumination (Rum) is set to 0.33. Optionally it can be set to a sinusoidal equation by Murphy et al (1982).

Microbial properties

Microbial growth in the model is driven by ATP from fermentation and also affected by the availability of ammonia and presence of added fat. The function for microbial growth (MiG) is:

$$MiG = ATPG*YATP*FGAM*FGFA$$

where:

ATPG = ATP available to rumen microbes for growth (moles d^{-1}),
YATP = microbial growth yield coefficient (kg $mole^{-1}$) and
FGAM = a variable moderating factorial growth depending on ammonia concentration. Microbial growth increases with the availability of ammonia.

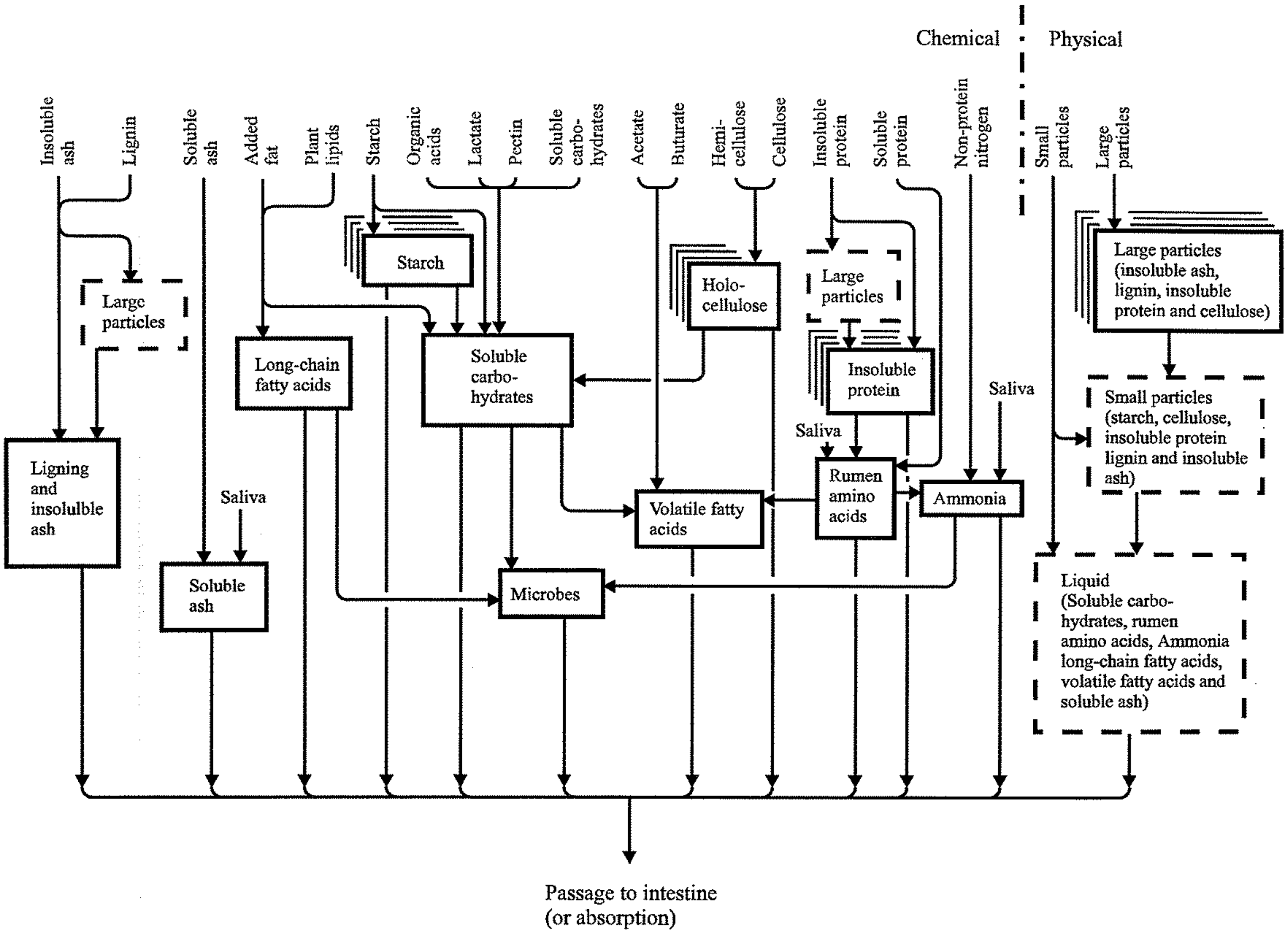

Figure 2. Diagramatic representation of the rumen model. Boxes enclosed by solid lines represent state variables. Boxes enclosed by dashed lines represent derived elements. Arrows identify transactions. Model extensions of the rumen are marked with four layers of boxes on top of each other. Sketched after Baldwin et al (1987 b).

FGFA is a variable moderating factorial growth due to ratio lipids from plants/other fat in feed. Hence growth decreases with added (animal) fats. Microbes are either in the liquid phase or attached to particles. Only substrates in contact with the liquid phase on the surface of the small particles are fermented.

There are two specific subgroups of microbes attached to small particles in the model, microbes associated with starch and holocellulose. The model of microbes is very intricate. Microbes are assumed to attach to small particles at a rate proportional to the concentration of microbes and the number of small particles. Some microbes are assumed to leave the rumen through passage. Others are assumed to get released from small particles due to hydrolysis. Some of the released microbes are assumed to reattach to other small particles of starch and holocellulose. The growth of microbes associated with starch and holocellulose depends on the ATP generated in fermentation of the respective carbohydrate. The equations for hydrolysis are of Michaelis Menten type and highly constrained so that implicit maximal rates of hydrolysis are specified.

Chemical properties

The following description of the chemical aspects of the model is divided into the sections *Indigestible fraction, Lipids, Carbohydrates, Protein and ammonia and Volatile fatty acids and lactate.*

Indigestible fraction

Starting at the left side of Figure 2, lignin and insoluble ash are combined to one small particle pool "Ot". Lignin and ash are distributed in both large and small particle fractions. The small particle fraction of lignin and insoluble ash enters directly into the Ot pool from the feed. The large particles are first ruminated into small particles. Lignin and insoluble ash leave the rumen through passage. The default value of passage rate constant is 0.33 d^{-1}.

There is soluble ash both in feed and in saliva. Saliva is assumed to contain 0.0085 kg ash l^{-1}. Saliva flow may be regulated to account for different secretion during eating, ruminating and resting. The default setting is such that the cow ruminates, eats and rests continuously with a salivation from rumination set at a value 0.333 of full rumination and salivation from eating at 2.2 times dry matter intake per day. Soluble ash is assumed to leave the rumen through passage of water to the lower tract and by absorption through the rumen wall. Default water passage rate constant is 3.5 d^{-1}. Absorption of soluble ash is assumed to be proportional to its rumen concentration (kg l^{-1}) and default absorption rate constant is set at 58.0 l d^{-1}.

Lipids

The pool of long chain fatty acids (Figure 2) is modelled to have two sources, plant lipids and other fat. The lipids in the feed are modelled to break down into fatty acids and the time for breaking down is not explicitly modelled. Plant lipids are assumed to have 1.8 fatty acids/triglyceride and other fats 3.0 acids/triglyceride. The unit for this pool is moles and the default mole weights are: for planι lipids 0.619 g $mole^{-1}$ and for other fat 0.885 g $mole^{-1}$. Long chain fatty acids are assumed to leave the rumen fatty acid pool through water passage and through uptake by microbes as they grow. Long chain fatty acid uptake is proportional to microbial growth by a factor 0.23 moles kg^{-1} microbial growth.

Carbohydrates
Starch is divided into soluble and insoluble starch. The soluble starch goes directly to the soluble carbohydrate pool. The rumen starch pool is called "Ha" and it has insoluble starch in the feed as the only source. Some starch leaves the pool through small particle passage and some starch is converted to soluble carbohydrates by microbes associated with Ha particles.

The other sources of soluble carbohydrates, except from the already mentioned soluble starch, are organic acids, lactate, pectin, soluble carbohydrates in feed, soluble carbohydrates from the breakdown of holocellulose and glycerol from feed lipids and fat. Most of the soluble carbohydrates are fermented to volatile fatty acids, but a small proportion is consumed by the microbes as storage carbohydrates and a proportion leaves the rumen through passage.

The holocellulose pool consists of the two subpools hemicellulose and cellulose. The reason for sometimes dividing the holocellulose pool into cellulose and hemicellulose is that their fermentation products differ and so can their rates of hydrolysis. The only source for the hemicellulose and cellulose pools is hemicellulose and cellulose in feed. Hemicellulose and cellulose inside large particles is not available for microbes and hydrolysis in the model and only the small particle fraction is modelled to enter these pools directly with the feed. The large particle fraction is modelled to enter the pools at the rate large particles are broken down into small particles. Some of the hemicellulose and cellulose is hydrolysed to soluble carbohydrates. The rate of hydrolysis is modelled to be influenced by rumen pH and dietary fat. The rate of hydrolysis of cellulose (SpCeCs) is calculated by the expression:

$$SpCeCs = KCeCs*(1-(fdfat/fdLi*KfatHb))*Ce*cMiHb \qquad (*)$$

where the degradation rate parameter KCeCs is assigned the default value 7.0 if the silage is mainly grass, 6.0 if it is mainly legumes and 9.0 if it is mainly corn silage. Fdfat is added fat, fdLi is plant lipids in feed and KfatHb is a variable with default value of 0.03. Ce is the current size of the cellulose pool and cMiHb is the concentration of microbes associated with holocellulose.

If rumen pH is lower than 6.2 the value of KCeCs is recalculated by the expression:

$$KCeCs = KCeCs-(KCeCs*1.875*(6.2-RPH)) \qquad (**)$$

where KCeCs to the right hand side is the old value of KCeCs and RPH is current value of rumen pH. The equation for hydrolysis of hemicellulose is calculated similarly. Note that the default values for KCeCs and the corresponding value for hemicellulose can be altered between runs.

Protein and ammonia
Protein is already in the description of the feed divided into soluble and insoluble protein. The soluble protein is modelled to immediately degrade to rumen amino acids. The inflow to the insoluble protein pool is modelled similarly as holocellulose with small and large particle protein. The equation for breakdown of insoluble protein to rumen amino acids and passage is exactly the same as for holocellulose. The only difference is that the parameter value of KPiAa, default value 7.0, which corresponds to KCeCs in equation (**), is not adjusted for different forage types or rumen pH. Another source of rumen amino acids is saliva. The default value of concentration of soluble protein in saliva is considered to be constant at 0.2%. Some of the rumen amino acids leave the rumen through passage, some are formed to volatile fatty acids and ammonia and some are used by microbes for growth. The fermentation rate of amino acids to volatile fatty acids (RAaFv) is:

RAaFv = VRAaFv*WaMi/(1.0+KRAaFv/cRAa)

where:

VRAaFv = proportional constant for amino acid and peptide fermentation (kg microbes)$^{-1}$,
WaMi = mass of microbes in rumen fluid (kg),
KRAaFv = affinity constant of microbes for amino acid fermentation and
cRAa = concentration of amino acids in rumen liquid (moles l^{-1}).

Volatile fatty acids and lactate

There are three different volatile fatty acids (VFA) explicitly modelled in the rumen model, namely acetate, propionate and butyrate. Other sources of acetate, apart from rumen amino acids, are soluble carbohydrates through fermentation, acetate in feed and acetate formed from lactate. For example the rate of acetate formed from soluble carbohydrates (CsAc) is:

CsAc = CsFv*CsFvAC

where:

CsFv is the rate of soluble sugar conversion to VFA in the rumen (moles d^{-1}) formulated as a function of the amount of microbes and concentration of soluble carbohydrates.
CsFvAC is moles of acetate formed per mole soluble carbohydrate fermented - a function of diet composition and relative rates of polysaccharide hydrolysis (moles mole^{-1}).

The formation of propionate, lactate and butyrate is modelled similarly except that there is no propionate in feed and no butyrate is formed from lactate.

The sources of rumen ammonia are rumen amino acids, saliva and transfer of plasma urea to rumen and hydrolysis to ammonia. The formation of lactate from amino acids is modelled to be proportional to the formation of VFAs from amino acids. The formation of ammonia from saliva is proportional to the concentration of plasma urea and secretion of saliva.

Water dynamics

Rumen liquid volume (Rlv) can be calculated either by using Rlv = Rdm /0.11-Rdm (default), where Rdm is rumen dry matter, or by using water dynamics and osmolarity from Dobson et al (1970). The second alternative is not generally applicable and should not be used for continuous feeding or unusual diets.

Refinements of the rumen model

In paragraph 4.2 it was described why the model was changed to contain four sets of feed description parameters instead of one. In order to make a more source-dependent behaviour concerning physical and chemical degradation of the four specified feeds possible, the large particle pool and each small particle pool affected by the large particle pool was divided into four subpools. The starch, cellulose and insoluble protein pools were divided in this way. The cellulose pool i.e. holocellulose (Hb) pool also consists of the two subpools cellulose (Ce) and hemicellulose (Hc). Hence

Hb = Ce+Hc.

The concentration of microbes associated with holocellulose (cMiHb) is defined as

cMiHb = HbMi/Hb.

Hence equation (*) in paragraph 4.3.1 is equivalent to

SpCeCs = KCeCs*(1-(fdfat/fdLi*KfatHb))*Ce/(Hc+Ce)*HbMi

The equation for the rate of hydrolysis of holocellulose (SpHbCs), which is identical to the equation (*) except that Ce is replaced by Hb, thus becomes:

SpHbCs = KHbCs*(1-(fdfat/fdLi*KfatHb))*HbMi

If one disregards effects of dietary fat and pH in the rumen, the rate of hydrolysis of the nutrients in all small particle pools by the microbes could be written in the basic form:

R = K * M

where:

R = rate of microbial hydrolysis,
K = degradation rate constant, and
M = respective subpool of microbes associated with starch, holocellulose and small particle microbes for insoluble protein.

In order not to change the behaviour of the sub-compartmentalized model in situations that could be modelled with the original model, each subflow Ri was modelled the same fractional way as was done for the breakdown of cellulose and hemicellulose as subpools to holocellulose in the equations above. This was written in the generalised form as:

Ri = K_i * M* P_i/P

where:

K_i = rate constant of subpool i,
Ri = rate of hydrolysis done by microbes in subpool i,
M = as above,
P_i = size of subpool i for respective nutrient in rumen and
P = ΣP_i = total size of pool for respective nutrient in rumen.

In situations where only one mixture of feed is used, P and R will obtain the same values as in the original model. In situations where the model user wants to handle different ingredients separately, this gives the possibility to specify different degradation rate constants concerning both physical and chemical degradation for these nutrients depending on sources of feed.

4.3.2 Intestine model

The intestine model consists of a number of equations computing the rate of moles of nutrients formed in the intestine. An example is the formation of glucose due to digestion of starch (LGHaGl):

LGHaGl = HaP*LGDCHa/MWSt

In these equations the variable on the left hand side is the rate of nutrient formation, in this case moles d^{-1}. HaP is passage of starch from the rumen and LGDCHa is the lower gut (intestine) digestion coefficient (default value 0.7). Division with MWSt converts kilograms of starch to moles of glucose with the factor of 0.162 (kg $mole^{-1}$).

Other sources of glucose computed in a similar way are glucose formed from intestinal digestion of microbes and moles of glucose equivalent available per day from microbial lipid. A glucose source that needs no computing is soluble carbohydrate passage from rumen. Since the absorbability of glucose is 100%, the absorption rate of glucose is simply computed as the sum of glucose formed from different sources. Other absorbed nutrients are computed in similar ways depending on digestibility and absorbability. The other absorbed nutrients computed in the intestine model are: amino acids, acetate, propionate, butyrate, ammonia, fatty acids and lactate. The nutrients absorbed in the intestine are transported via the blood to the udder and body models.

4.3.3 Body fluids and tissue

In this paragraph we treat the body fluids and the body model consisting of the parts adipose tissue, viscera and empty body weight.

Body fluids

Distribution volumes are used for all body compartments including blood. Here, there are four state variables: glucose, amino acids, acetate and fatty acids.

Entries to glucose are from propionate, lactate, glycerol, amino acids and direct absorption of glucose. Outputs are conversion of glucose to lactose in udder, glucose oxidation via pentose cycle for nicotine adenine diphosphate (NADPH2) in adipose tissue and udder, glucose conversion to triose-phosphate in adipose tissue and udder, glucose conversion to lactate in body and glucose oxidation to CO_2 and H_2O.

Inputs to the pool of plasma amino acids are amino acids absorbed from the whole intestine and amino acids from degradation of protein in body and viscera.

Outputs are amino acid incorporation into body and viscera protein, milk protein, gluconeogenesis from amino acids in viscera, salivary protein entry to the rumen amino acid pool and amino acid use for pregnancy.

Inputs to plasma acetate are acetate absorbed from the whole intestine and formation of acetate from amino acid degraded in the liver. Outputs are acetate oxidation to CO_2 and acetate used for fat synthesis in adipose tissue and for milk fat synthesis.

Inputs to the pool of plasma lipids are absorption of fatty acids in intestine and fatty acid formation from body fat hydrolysis. Outputs are lipid oxidation, uptake of plasma lipids and esterification in adipose and blood lipid incorporation into milk fat.

Body tissue

The body model consists of the four state variables empty body weight, adipose tissue, viscera and body protein.

The body model has the following structure. Body weight (BW) is:

$$BW = EBW+RUMVOL$$

where:

RUMVOL = the weight of rumen including dry matter (see Water dynamics in paragraph 4.3.1) and

EBW = empty body weight.

Empty body weight is defined as the sum of weights of fat, viscera, protein in body and weight of components of body other than protein.

Adipose tissue

Weight of fat is modelled as the sum of weights of adipose cytosol and triglycerides in body. Adipose cytosol is a constant calculated to 3% of body weight at the start of simulation and the triglyceride storage in body is dynamic. The triglyceride storage or pool is modelled to have two inflows; one is the rate of body fat synthesis from acetate, the other is the rate of lipid uptake and esterification. The only outflow from the pool is the rate of fat hydrolysis. These flows are modelled very intricately. For example, the equation for the rate of body fat synthesis from acetate is:

$$\mathrm{TsFaF1} = \frac{0.333 * \mathrm{VTsFaF} * (\mathrm{EBW} * * 0.75) * \mathrm{CHOR1} * \mathrm{T3}}{1.0 + (\mathrm{cFa} / \mathrm{K1TsFa}) * * \mathrm{EXP10} + (\mathrm{KTsFaF} / \mathrm{cTs}) * * \mathrm{THETA1}}$$

where:

VTsFaF = proportional constant for lipolysis in adipose 5.334 moles d^{-1} default value 0.1,

CHOR1 = concentration of catabolic hormone one (arbitrary units),

T3 = thyroid hormone expressed as multiple of basal/normal (arbitrary units),

cFA = concentration of plasma lipids (includes non-esterified fatty acids (NEFA) plus triglycerides),

K1TsFa = inhibition constant (default value 5.0E-4 moles l^{-1}),

KTsFaF = affinity constant for body fat lipolysis (0.2 moles l^{-1}),

cTs = concentration of triacylglycerol in adipose tissue (moles l^{-1}) and

THETA1 = steepness exponent for triacylglycerol effects on lipolysis in adipose tissue (5.0).

As can be seen from the above equation, the rate of body fat synthesis from acetate is mainly controlled by the concentrations of catabolic hormone, lipids, triacylglycerol in adipose tissue and glucose. Similarly the rate of fat synthesis from acetate is a function of concentration of acetate and glucose. The rate of fat hydrolysis is a function of injected insulin, lipids and glucose.

Viscera

The model of viscera consists of visceral protein and other weight viscera. Other weight viscera is a constant 4.4% of initial body weight. Visceral protein synthesis is modelled as a function of visceral DNA, plasma amino acids and glucose concentration. Visceral DNA is modelled as a state variable which is dependent on several parameters that can be altered between runs, but on no other variables. Rate of viscera protein degradation is simply modelled to be proportional to the visceral protein pool.

Body protein
The structure and equations of the body protein model are exactly the same as visceral protein in the model of viscera. The only differences are parameter values.

4.3.4 Udder

The rate of secretion of milk in the udder in this model is directly proportional to the conversion of glucose in blood into milk lactose. The reason for this is that the proportion of lactose in milk in reality is found to be fairly constant, controlling osmolarity which has to be in balance with the osmolarity in blood as lactose is the overall dominating factor. The glucose incorporation into lactose is assumed to depend on the presence of various udder enzymes, effect of retained milk and the proportion of glucose and amino acids in blood. The equation for the rate of transformation of glucose to lactose (GlLmV) is:

GlLmV = VGlLmV*Uenz*KMinh/(1.0+KGlLmV/cGl+KAaLmV/cAa) (***)

where:

VGlLmV = proportional constant for glucose conversion to lactose/unit Uenz (default value 0.0039 moles d^{-1}),
Uenz = udder enzymes (arbitrary units) and
KMinh = factor defining inhibition of milk synthesis by retained milk in the udder.
KGlLmV = affinity constant for glucose conversion to lactose in udder (default value 0.003 moles l^{-1}),
cGl = concentration of glucose in blood (moles l^{-1}),
KAaLmV = affinity constant for amino acids for alphalactalbumin synthesis (default value 0.002 moles l^{-1}) and
cAa = concentration of amino acids in blood (moles l^{-1}).

Retained milk effects

There are two effects of retained milk in the model, one long-term effect of average milk in udder on the degradation of udder enzymes (Uenz), and a short term effect on milk synthesis (KMinh) defined by:

KMinh = (Mlkmax-Umilk)/(Mlkmax-Umilk+KmlkI)

where:

Mlkmax = maximum mammary capacity for milk (kg)
Umilk = milk in udder (kg)
KmlkI = factor used to scale KMinh (kg)

The formation of the lactation curve is done by the variable Uenz. The equations for udder enzymes are adopted from the lactation model of Neal and Thornley (1983). However in that model, the effect of the change in the number of udder enzymes was interpreted as an effect of a change in the number of "differentiated secretory cells". The amount of udder enzymes forms a pool where the synthesis of enzymes (Usyn) occur according to the equation

Usyn = VUsyn*Ucells*Lhor*BST/(KUsyn+Lhor*BST)

where:

VUusyn = enzyme synthetic capacity per cell (1.0 moles d^{-1}) and
Ucells = udder cells (1,000).
Lhor = lactation hormones which enhance synthesis of udder enzymes,
BST = bovine somatotrophin (hormone) and
KUsyn = Michaelis Menten type constant defining effect of LHOR on rate of udder enzyme synthesis (0.2 kg).

The lactation hormone (Lhor) is a non-specific hormone where 1.0 kg is assumed to be present at the beginning of lactation. Its decline rate (DLhor) is:

DLhor = -KLhor*Lhor

where:

KLhor = a degradation constant for lactation hormone (default value 0.0102 d^{-1}).
Lhor = a non-specific lactation hormone.

The udder enzymes degrade at the rate (Udeg) given as:

$$\text{Udeg} = \frac{\text{Uenz}*(\text{KUdeg}+\text{KUdegM}*(\text{UMave}/\text{KMdeg})**\text{THETA5}}{(1.0+\text{UMave}/\text{KMdeg})**\text{THETA5}}$$

where:

KUdeg = rate constant for degradation of udder enzymes which is not dependent on retained milk (0.10 d^{-1}),
KUdegM = rate constant for degradation of udder enzymes which is dependent on retained milk (d^{-1}),
Umave = milk in udder - sort of rolling average (kg),
KMdeg = rate constant to decrease udder metabolic capacity when average retained milk is high (d^{-1}) and
THETA5 = steepness exponent for effect of retained milk on rate of udder enzyme degradation (10.0).

Secretion of milk protein and fat

The equation for synthesis of milk protein is similar to the equation for production of lactose. The structural difference is that when synthesis of lactose depends of the concentration of amino acids and glucose, the dependency of glucose concentration is excluded in the synthesis of protein equation. Compare the equation:

AaPmV = VAaPmV*Uenz*KMinh/(1.0+KAaPmV/cAa)

where:

AaPmV = rate of amino acid incorporation into milk protein (moles d^{-1}),
VAaPmV = proportional constant for milk protein synthesis per unit Uenz and
KAaPmV = affinity constant for amino acids for milk protein synthesis (0.0021 moles l^{-1})

with equation (***) above.

Milk fat is synthesised from blood acetate and lipids. The synthesis of milk fat from acetate (AcTmV) occurs according to the equation:

AcTmV = VAcTmV*Uenz*KMinh*INS**P1/(1.0+KAcTmV/cAc+ K1Actm/cGl)

where:

VAcTmV = proportional constant for milk fat synthesis from acetate/unit of Uenz (0.00649 moles d^{-1}),

INS = injected insulin as effector of glucose uptake (experimental option),
P1 = exponent or steepness parameter for insulin responses (2.0) and
KAcTmV = affinity constant for acetate for milk fat synthesis (0.0018 moles l^{-1}).
K1Actm = affinity constant for for milk fat synthesis from acetate depending on concentration of glucose in blood (0.001 moles l^{-1})

Synthesis of milk fat from blood lipids (FaTmV) is modelled with the similar equation:

$$\mathrm{FaTmV} = \frac{\mathrm{VFaTmV * Uenz * KMinh * INS ** P1}}{1.0 + \mathrm{KFaTmV / cFa} + \mathrm{K1FaTm / cGl}}$$

where:

FaTmV = rate of blood lipid (FA) incorporation into milk fat (moles d^{-1}),
VFaTmV = proportional constant for blood lipid uptake for milk fat synthesis,
KFaTmV = affinity constant for blood lipid to milk fat (5.0E-4 moles l^{-1}) and
cFa = concentration of blood lipid (moles l^{-1}).
K1FaTm = affinity constant for for blood lipid to milk fat depending on concentration of glucose in blood (0.0015 moles l^{-1})

5. DEBUGGING AND TESTING OF EXTENSIONS AND REFINEMENTS

To test if the extended model (Molly II) would perform as the original Baldwin model (Molly I) for the same types of simulations, a test was performed. The test was done by setting up a simulation of feeding a mixture of hay, silage, a common concentrate and grain twice a day. In the feed description to Molly I the feed parameters describing the content of starch, cellulose and insoluble protein constituted a weighed average of the four feeds whereas in Molly II they where parameterised separately for each feed. After debugging Molly II the simulation results from the two versions of the model converged. However since the initialization of the five sub-compartmentalized pools was not done in exactly the same way in both models, the results did not converge completely before the model had stabilised after some days. Two simulations were done to compare Molly II results to those of Molly I.

6. RESULTS OF THE REFINEMENTS

The next step after comparing the Molly II results to those of Molly I was to test a simulation for which Molly II was designed, namely a situation where a cow was fed different types of feeds, multiple times a day. The simulated scenario was that a cow was fed concentrate four times and forage two times daily.

Simulation A was an approximate simulation of the scenario which could almost be done by use of Molly I. The intake pattern of the scenario was used, but the four feeds were fed as a completely mixed ration and the degradation rate constants for the separate feeds were weighted together into one set of parameters. However, the testing of Molly II (see Chapter 5) showed that the results were the same when using the same degradation constants in Molly I as in Molly II. Instead of rebuilding Molly I to handle this feed intake pattern, Molly II was therefore used with the single set of parameters and degradation rate constants in all four sets of feed descriptions and model extensions.

Simulation B was the scenario with Molly II and separate feed description and intake parameters for hay, silage, concentrate and grain.

Amounts and times for simulations A and B are shown in Table 1.

Table 1. Amounts and times for feeds in simulations A and B.

Simulation	Feed	Time of day (hours and minutes)	Duration time (hours)	Amounts (kg dry matter)
A	Mixture	4:00, 4:30, 9:00, 14:00,14:30 and 19:00	0.5, 4.0, 0.5, 0.5, 4.0 and 0.5	4.75, 4.25, 4.75, 4.75, 4.25 and 4.75
B	Oats/Barley	4:00, 9:00, 14:00 and 19:00	0.5	1.75
	Concentrate	4:00, 9:00, 14:00 and 19:00	0.5	3.0
	Hay	4:30 and 14:30	4.0	0.5
	Silage	4:30 and 14:30	4.0	3.75

The four feed descriptions and degradation rate constants are presented in Tables 2 and 3.

Table 2. The physical and chemical feed description parameters used in simulations A and B.

	Simulation A	Simulation B			
	Mixture	Oats and barley	Concentrate	Grass silage	Grass hay
Feed composition	(%)	(%)	(%)	(%)	(%)
Starch solubility	36.506	40.0	24.6	0.0	0.0
Particle size factor	38.848	50.0	50.0	20.0	20.0
Soluble carbohydrates	8.71	1.55	16.89	2.0	11.0
Starch	15.7	47.55	8.141	0.0	0.0
Hemicellulose	15.969	18.0	9.866	22.5	26.0
Cellulose	14.221	9.35	10.406	22.5	32.0
Organic acids	0.455	0.0	0.0	1.0	5.0
Lactate	3.273	0.0	0.0	12.0	0.0
Acetate	0.545	0.0	0.0	2.0	0.0
Butyrate	0.0	0.0	0.0	0.0	0.0
Pectin	3.323	0.0	6.99	1.0	0.0
Lignin	2.96	3.5	2.367	3.0	6.0
Soluble protein	2.049	4.3	1.542	0.9	1.0
Insoluble protein	15.065	8.6	23.863	8.1	7.0
Non Protein Nitrogen	4.79	0.2	4.985	9.0	3.0
Lipids (vegetable)	4.398	3.5	5.453	4.0	1.0
Other fat	1.091	0.0	2.5	0.0	0.0
Soluble ash	5.111	2.8	4.037	9.0	5.0
Insoluble ash	2.385	0.65	2.962	3.0	3.0

Table 3. Degradation rate constants used in simulations A and B.

Simulation	Feed	Hydrolysis of: Cellulose (d^{-1})	Hemi-cellulose (d^{-1})	Starch (d^{-1})	Insoluble protein (d^{-1})
A	Mixture	5.685	5.298	12	14.559
B	Oats and barley	3.2	3.2	12	17.8
	Concentrate	8	8	12	8.9
	Grass silage	4.8	4.8	-	35.6
	Grass hay	6.4	6.4	-	35.6

This presentation focuses on the results in the five sub-compartmentalized state variables starch, holocellulose, hemicellulose, cellulose and insoluble protein. The results for a few other key parameters in the model are also presented and discussed. Results are from day ten with time (T) in days. The blue curves represent simulation A and the red curves simulation B.

6.1 Effects on particle size distribution

Figure 3 shows the result of using different particle size factors for the different feeds in simulation B compared with simulation A where a weighed average particle size factor was used. There was a slight difference in the inflow of large particles.

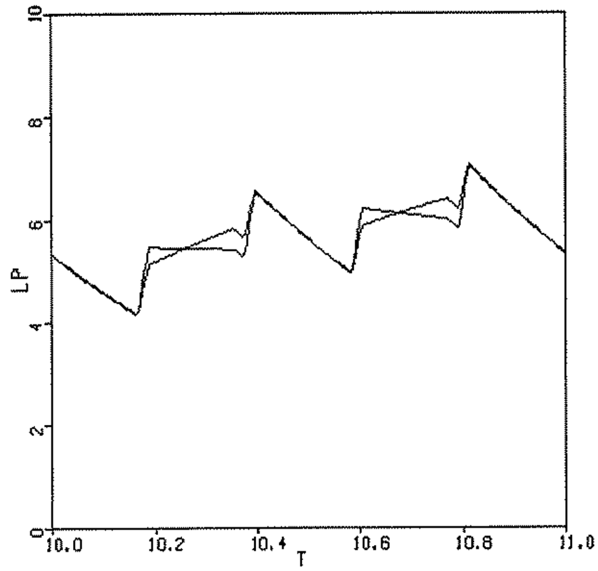

Figure 3. Large particle pool size (LP in kg). In simulation B a larger proportion of the large particles enters the rumen with the forage than in A where an average particle size factor was used for all feeds. The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

6.2 Dynamics of starch in the rumen model

The comparison of the dynamics of starch between the two simulations was more complicated than comparing the particle size since the behaviour of starch involves dynamics of microbes. The inflow to the starch pool (Ha) in simulation B was dependent on the content of starch in each of the four feeds: silage, hay, concentrate and grain (specified in the four parameters FdSt1, FdSt2, FdSt3 and FdSt4). Different solubilities for the starch in each ingredient could also be specified but in this example the starch solubility was set to the same value for all starch sources. The patterns of starch inflow to the rumen for simulations A and B are shown in Figure 4. Since the forage does not contain any starch, the pattern becomes quite different between the two simulations.

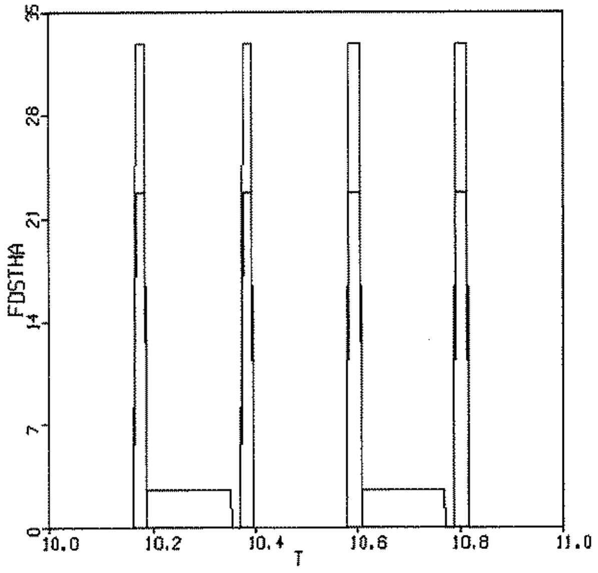

Figure 4. Inflow of starch from feed to the rumen (FDSTHA in kg d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

Starch escaping fermentation in the rumen was simply proportional to the size of the starch pool times the passage rate constant for small particles (KSPP).

The hydrolysis of starch to glucose (SPHACS), the first step in the digestive process, was assumed to be done by microbes associated with starch and is shown in Figure 5.

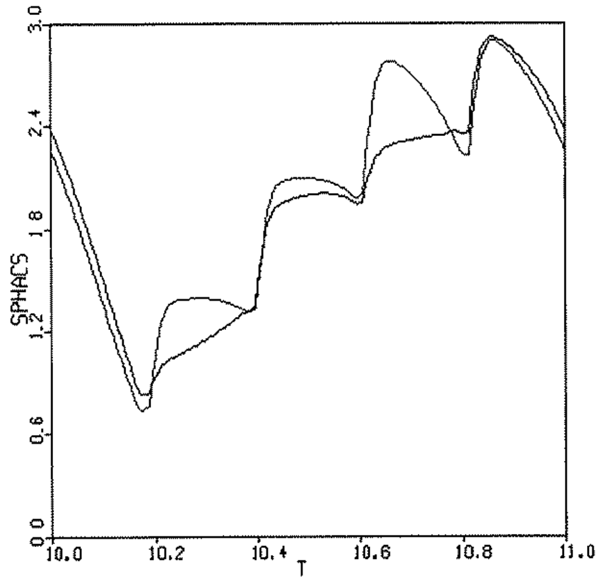

Figure 5. Hydrolysis of starch to glucose (SPHACS in kg d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

Note the similar pattern of the number of microbes associated to starch (HAMI) in Figure 6 as can be expected by the definition of the starch hydrolysis since the hydrolysis is proportional to the number of microbes. The difference in the pattern of microbial growth and hydrolysis between the two simulations was caused by the difference in inflow of insoluble starch.

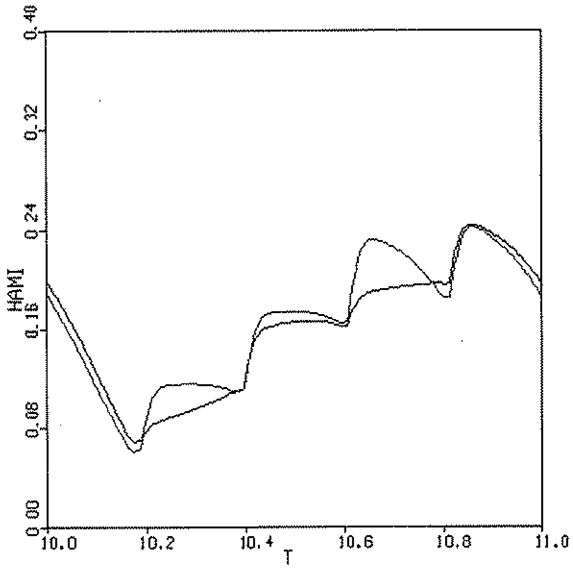

Figure 6. Microbes associated with starch (HAMI in kg). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

The behaviour of non-soluble starch (HA) is shown in Figure 7. Note the small dips on the curve for simulation A caused by the small break in feed intake starting at the same time as the forage meal ends in simulation B.

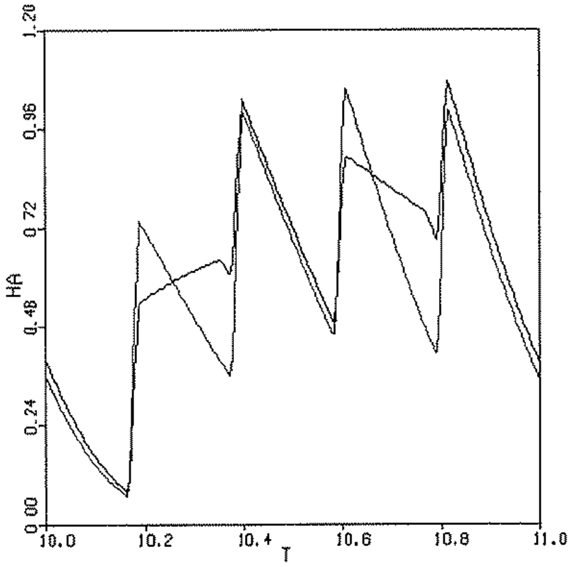

Figure 7. Insoluble starch in rumen (Ha in kg). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

6.3 Dynamics of cellulose

The comparison of cellulose dynamics between the two runs was even more complicated than for starch. There are three reasons for this: Firstly, cellulose exists in two pools - large and small particles. Some of the cellulose goes to the large particle pool before it enters the cellulose pool, depending on the different particle size factors for the ingredients. Secondly, different degradation rate constants were used for the forage and the concentrate, affecting the fermentation. Thirdly, the microbial fermentation of cellulose was affected by rumen pH.

Figure 8 shows the entry to the hemicellulose pool directly from feed and Figure 9 from the large particle pool.

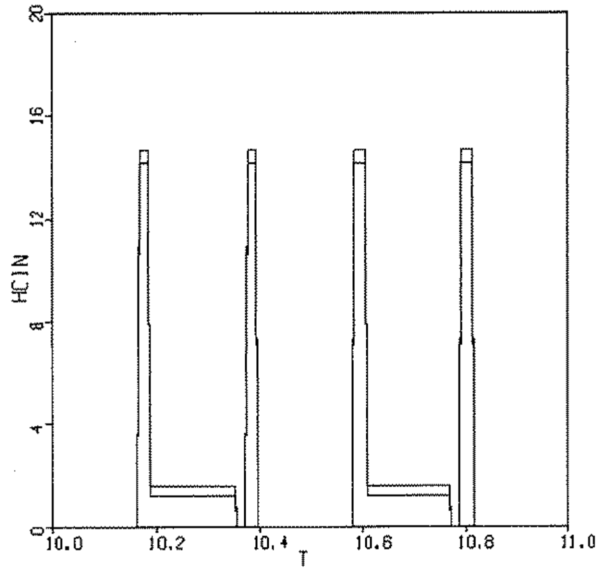

Figure 8. Entry of hemicellulose to the rumen with small particles in feed (HCIN in kg d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

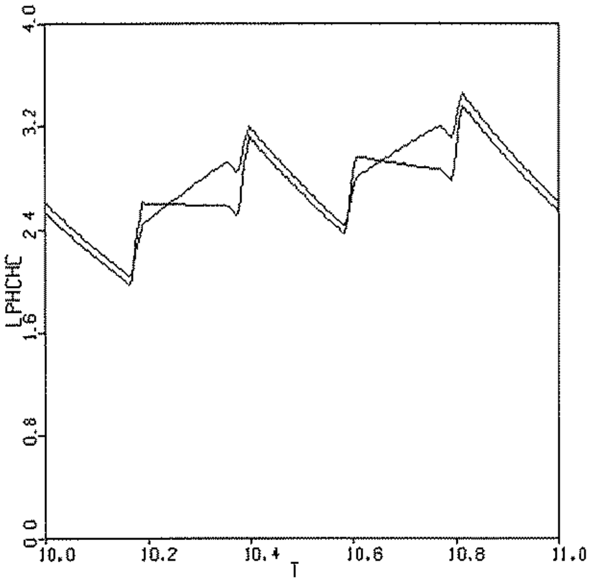

Figure 9. Rate of formation of small particle hemicellulose in the rumen from breakdown of large particles (LPHCHC in kg d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

As can be seen in Figure 9 the inflow from the large particle pool was slightly lower in simulation A. This was because the particle size factor in simulation A was weighted together from the content of all the chemicals constituting the large particles so that the average total large particle pool size was the same in the two simulations. Naturally this did not have the effect that the contents of individual nutrients were the same in the large particle pool.

Figure 10 shows the microbial breakdown of the hemicellulose.

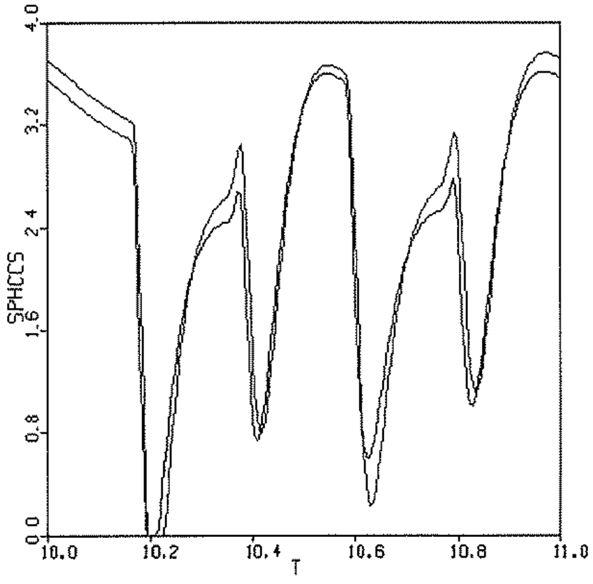

Figure 10. Rate of hydrolysis of small particle hemicellulose (SPHCCS in kg d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were

used. The blue curve shows the simulation A where these facilities were not used. Note the influence from rumen pH in equation (**).

Figure 11 shows the size of the hemicellulose pool.

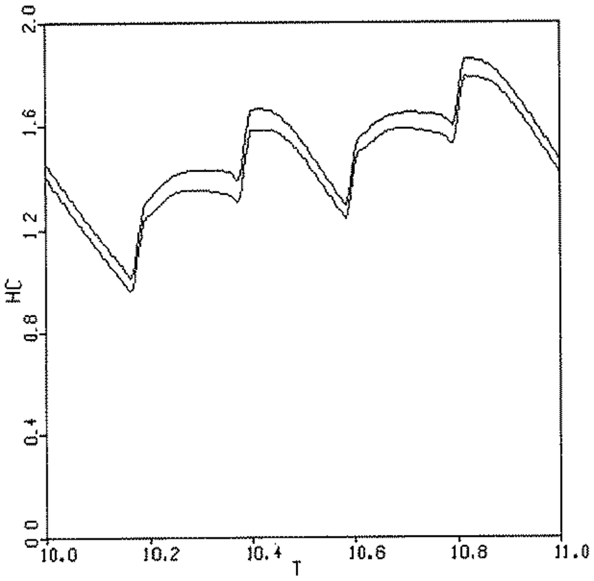

Figure 11. Hemicellulose in rumen (HC in kg). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

The outflow through passage was proportional to the size of the hemicellulose pool regulated by the passage rate constant 1.33 turnovers per day. The dynamics of holocellulose were in principle the same as those of hemicellulose.

6.4 Dynamics of insoluble protein and amino acids

As can be seen in Figure 12 most of the insoluble protein which entered the insoluble protein pool directly from feed came from the concentrate.

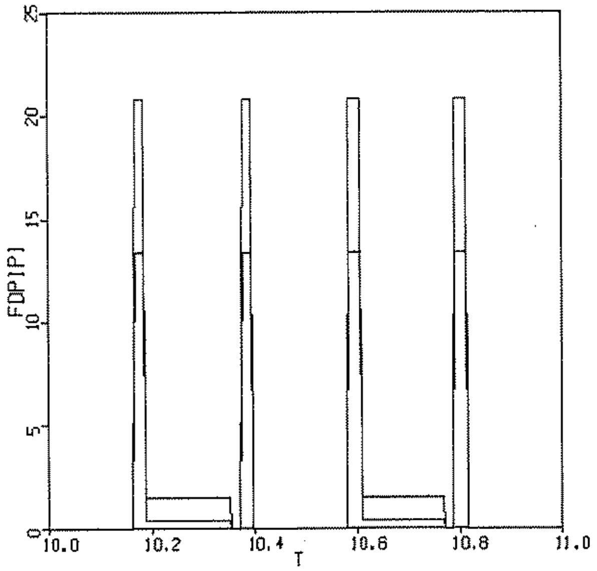

Figure 12. Entry of insoluble protein to the rumen with small particles in feed (FDPIPI in kg d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

Using only one average set of parameters for description of feed and particle size in simulation A resulted in overestimation of the part of insoluble protein going through the large particle pool (see Figure 13).

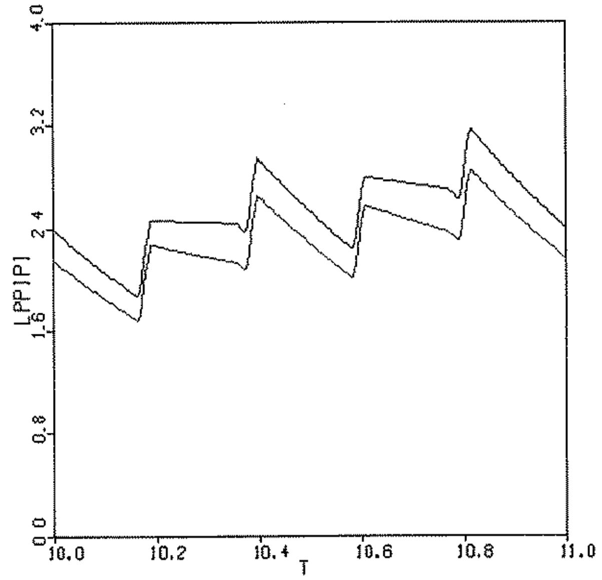

Figure 13. Rate of formation of small particle insoluble protein in the rumen from breakdown of large particles (LPPIPI in kg d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

The rate of microbial breakdown of insoluble protein to amino acids can be studied in Figure 14.

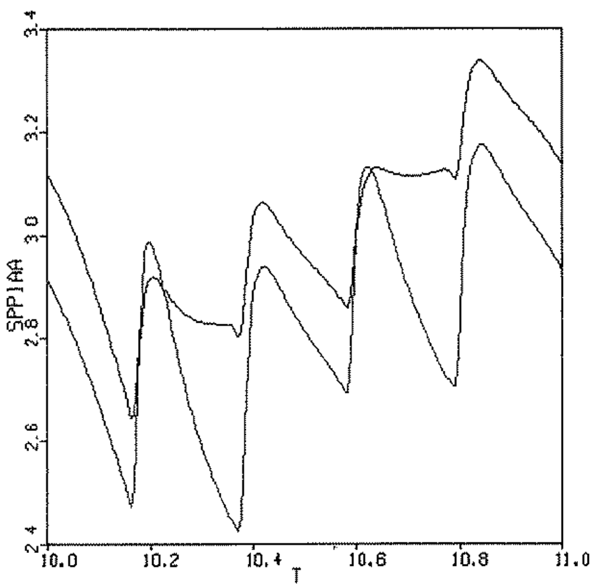

Figure 14. Rate of hydrolysis of insoluble protein in small particles (SPPIAA in kg d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

The rate was slightly lower for simulation B, mostly depending on the lower degradation rate for the insoluble protein in the concentrate. The lower microbial breakdown of insoluble protein gives a lower level of rumen amino acids and thereby a lower passage rate of rumen amino acids into the intestine, see Figure 15.

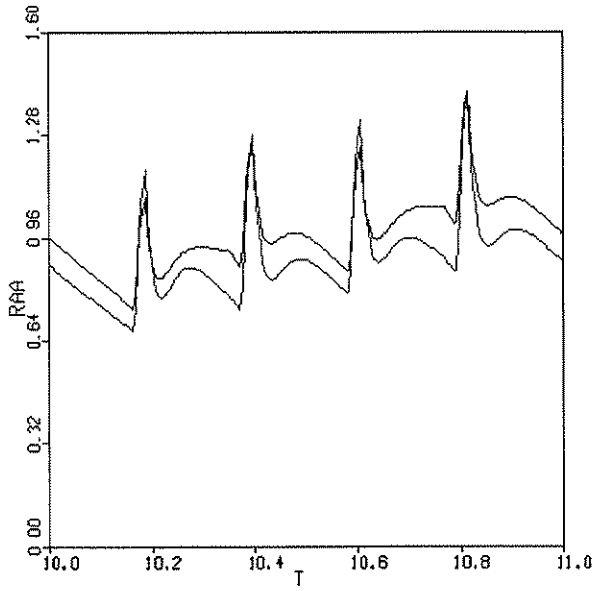

Figure 15. Rumen amino acids (RAA in moles). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

The level of amino acids in blood (AA) in simulation B in Figure 16 was, however, higher than in simulation A depending on the higher rate of absorption of amino acids in the intestine (ABSAA) shown in Figure 17.

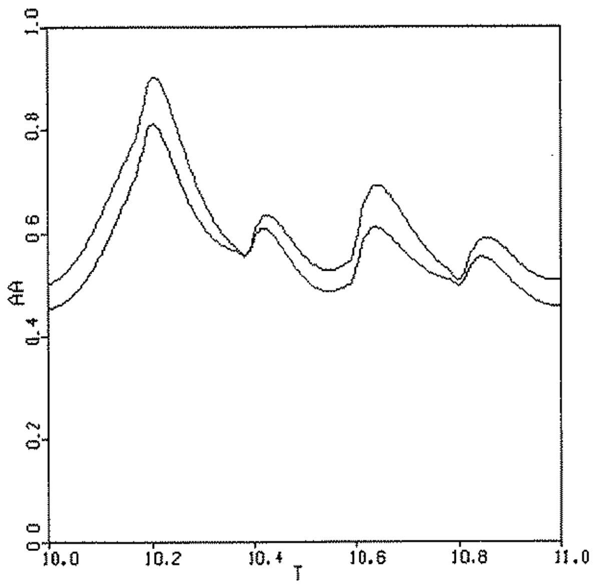

Figure 16. Plasma Amino Acids (AA in moles). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

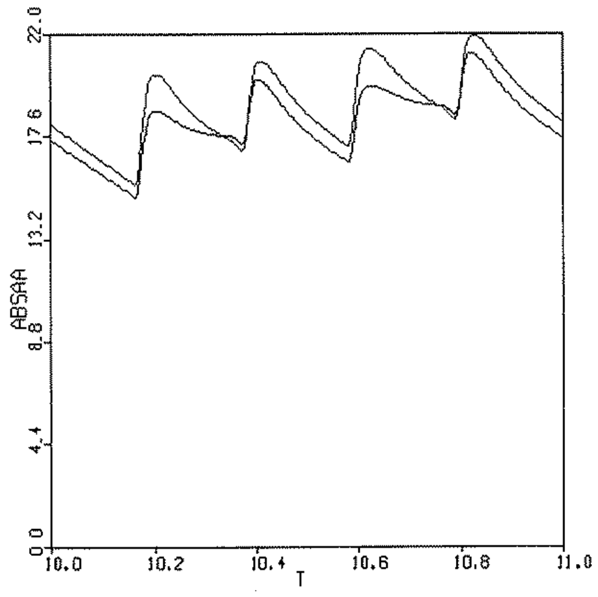

Figure 17. Absorbtion of amino acids in the intestine (ABSAA in moles d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

6.5 Dynamics of soluble carbohydrates, volatile fatty acids and rumen pH

The intake of soluble carbohydrates enters the rumen soluble carbohydrate pool instantly as it comes in as the sugar equivalents shown in Figure 18.

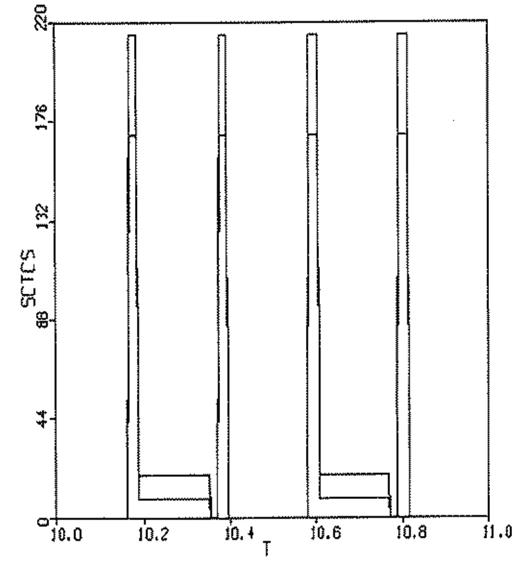

Figure 18. Total rate of entry of sugar equivalent in feed to the rumen soluble carbohydrate pool (SCTCS in moles d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

The difference in the intake pattern of soluble carbohydrates between the two runs was here explained by the simple fact that the concentrate contained more of the sources of soluble carbohydrates than forage. The difference in the intake pattern between simulations A and B was also the reason why the dynamics of the total pool of soluble carbohydrates in the rumen differed between the two simulations at the time of intake of forage (see Figure 19).

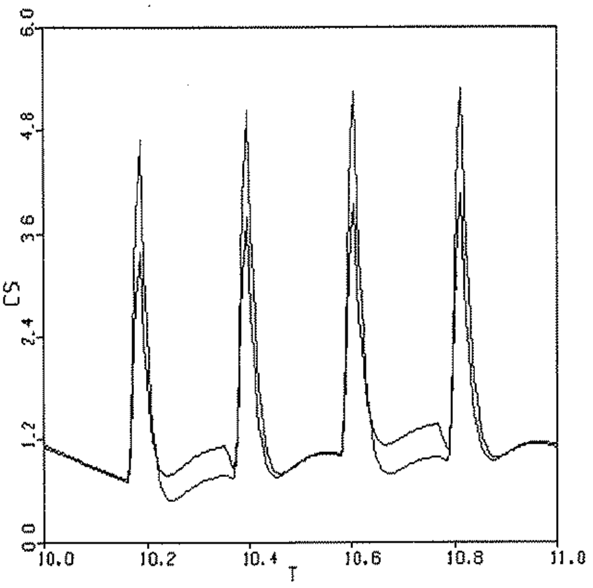

Figure 19. Soluble carbohydrate pool in rumen (CS in moles). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

The conversion rate of soluble carbohydrates to volatile fatty acids (CSFV) is shown in Figure 20.

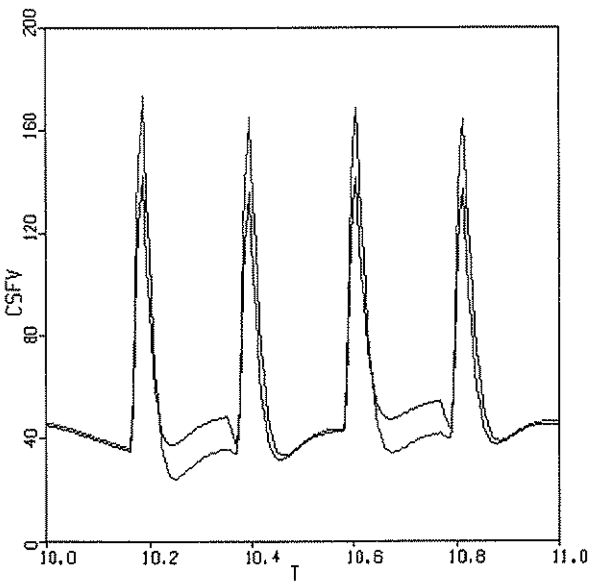

Figure 20. Rate of soluble carbohydrate conversion to volatile fatty acids in rumen (CSFV in moles d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

Figure 21 gives the predicted amounts of the pool of the volatile fatty acid propionate. The predicted rates for acetate and butyrate have similar patterns but are higher for acetate and lower for butyrate.

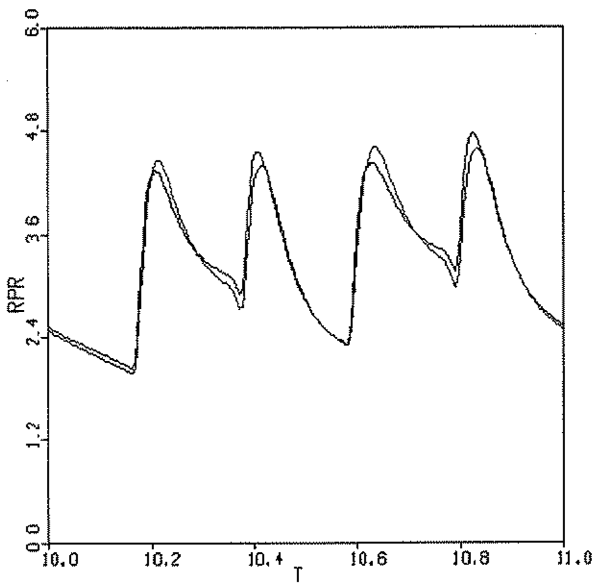

Figure 21. Rumen propionate (RPR in moles). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

The level of lactate was too high (Peter Udén, personal communication). The resulting fluctuation of rumen pH is shown in Figure 22.

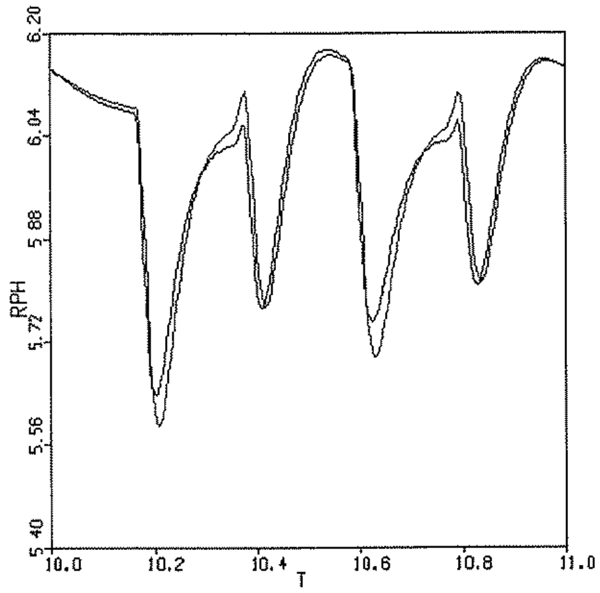

Figure 22. Rumen pH (RPH). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

As could be expected from the inflow of soluble carbohydrates, the rumen pH fluctuated slightly more in simulation B.

6.6 Dynamics of long chain fatty acids, glucose and secretion of milk

There was a small difference between the two simulations in the long chain fatty acids in the rumen depending on the differences in the pattern of inflow of vegetable lipids and animal fat and a slight difference in the microbial growth pattern.

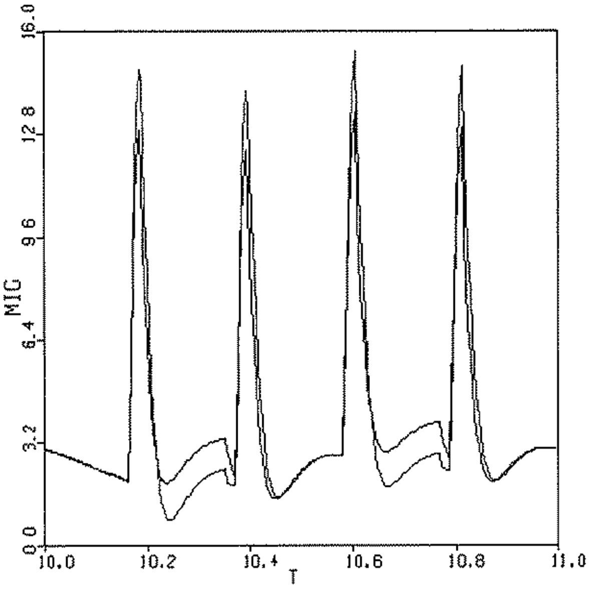

Figure 23. Rate of microbial growth (MIG in kg d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

Since the sources of the absorption of fatty acids in the intestine were the outflows from the pool of long chain fatty acids in the rumen, the pattern for the absorption rate of fatty acids in the intestine into blood was almost proportional to the pattern for the rumen long chain fatty acids. Absorption of fatty acids in the intestine and other sources and sinks for fatty acids in blood add up to a very small differences between the two simulations in the resulting pattern of fatty acids in blood.

Figures 24 and 25 show two of the five entries to the pool of glucose in blood. These figures also show that the uptake of glucose in the intestine is small compared to other sources. The rate of glucose formation from lactate was unrealistically high (Peter Udén, personal communication).

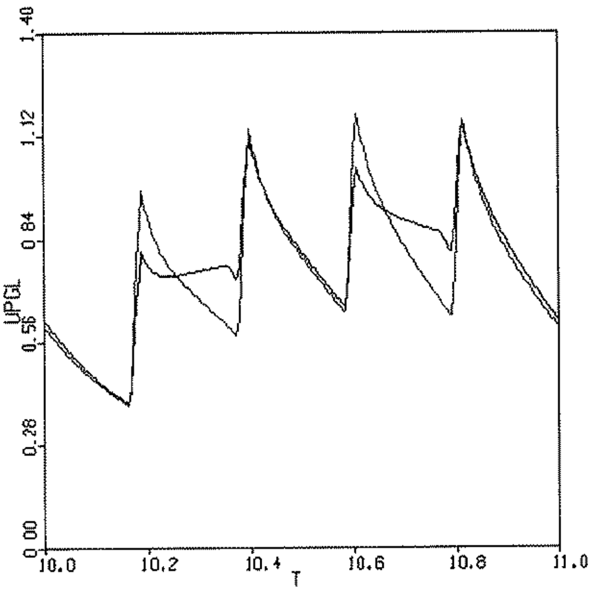

Figure 24. Uptake of glucose to the blood from the intestine. (UPGL in moles d^{-1}) The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

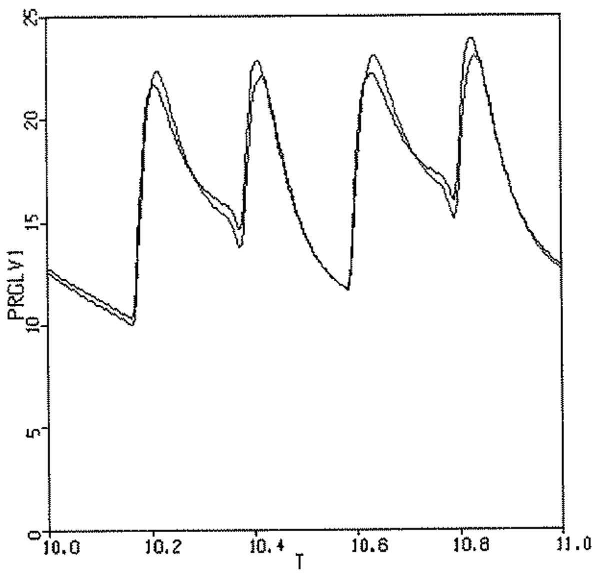

Figure 25. Rate of glucose formation from propionate (PRGLV1 in moles d^{-1}) The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

The net result of the inflows and outflows to the pool of glucose in the blood is shown in Figure 26. The difference between the two simulations of the glucose pool is small compared to the unrealistically large daily fluctuations. However, the differences between the two simulations in the concentration of plasma glucose and amino acids are the cause of the difference in the predicted daily secretion of milk shown in Figure 27, since all other parameters controlling the secretion rate were the same.

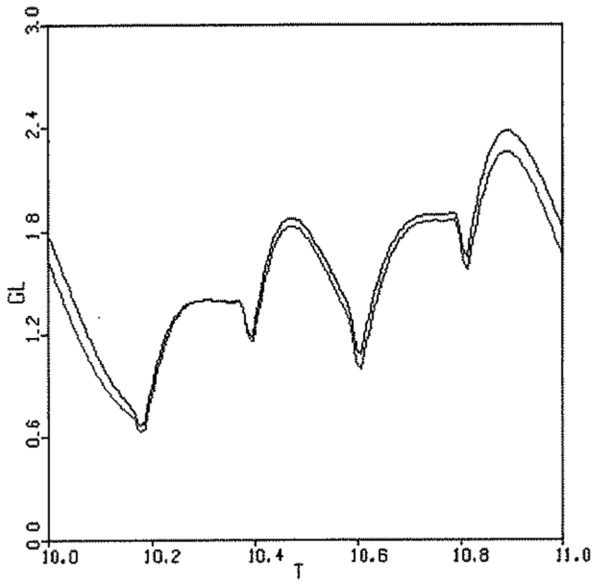

Figure 26. Plasma glucose (GL in moles). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

The rates of milk secretion in the two simulations are shown in Figure 27.

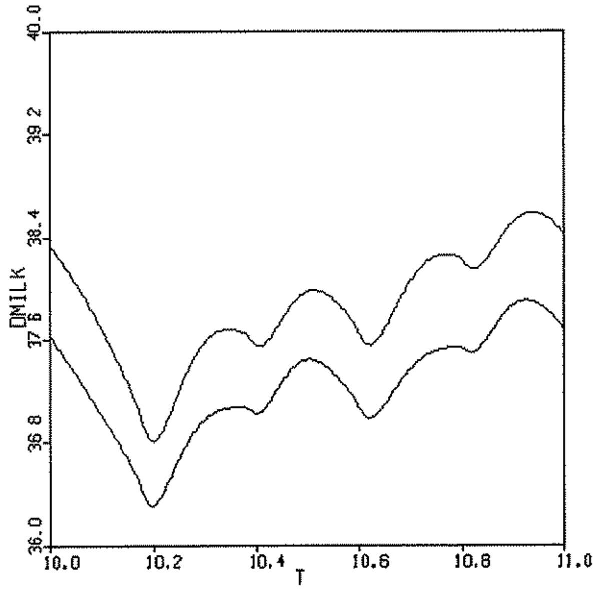

Figure 27. Rate of milk secretion (DMILK in kg d^{-1}). The red curve shows the more precise simulation B where the facilities of the refined Molly II were used. The blue curve shows the simulation A where these facilities were not used.

7. DISCUSSION

The purpose of this project was to model the dynamics of the metabolic process from fodder to milk in the dairy cow under Swedish conditions and under different management situations.

A literature review showed that a number of relevant studies have been carried out abroad. After studying several of those models it was decided to use the model "Molly" developed by Baldwin et al from 1975 through to 1987.

The chosen model (Molly) of the metabolism in the dairy cow was probably the best available and most tested dynamic model at the time this project started. However, Swedish conditions differ from those in the U. S. in the sense that it is common practice in Sweden to schedule the feeding of forage and concentrate separately within a day. Separate parameters and facilities for scheduling the intake of four different feeds had to be included.

A consequence of the more complex Swedish feeding conditions was that the model had to be extended to include four separate sets of physical and chemical feed descriptions instead of one. Therefore, parts of the rumen submodel were separated into four parallel submodels to facilitate different degradation rates depending on source of feed. An alternative to our approach could have been to sum up the individual constituents of the four feeds as an input to the rumen, rather than dividing parts of the rumen into four parallel compartments. This would, however, have eliminated the possibility of having separate degradation rates depending on source of feed. In the case where degradation rates are not assumed to be dependent on source of feed, this should have given the same result as in the chosen approach.

To make the modified model useful, we have devoted much effort to structuring and describing the whole model from an overview down to essential details in this thesis.

The original model by Baldwin et al (1977; 1987a,b,c) is regarded as well tested and reasonably validated. However, the credibility of the results varies for different parameters. In a few cases such as the level of rumen lactate and the daily fluctuations of glucose in the blood, the values seem too high.

After our extensions and refinements, the new parts of the model and their consequences had to be tested. First detailed debugging and testing was performed. Then the results of the modified model were compared to those of the old one under identical feeding conditions. This showed that the modifications of the model did not change its behaviour when simulating the same feeding of complete mixed rations of feed.

Finally, the differences in metabolism between the original U.S. Molly and the refined Molly II model under Swedish conditions were analysed in detail. The modifications resulted in distinct changes in various metabolic quantities over time, but the average levels of these quantities were less affected.

When considering the way the model is constructed, the results seems to be reasonable with some exceptions. Validating experiments comparing model quantities to corresponding quantities in the real cow have not been carried out. Therefore it is too early to say for which purpose division into sub-compartments in the model is necessary to produce better predictions of details like effects of rumen pH and animal performance on a whole animal level. Such experiments and subsequent refining of the model may be an area for future research and refinements.

8. REFERENCES

Baldwin R. L., Koong L. J. and Ulyatt M. J. 1977. A dynamic model of ruminant digestion for evaluation of factors affecting nutritive value. Agricultural Systems 2, pp 255-288.

Baldwin R. L., France J. and Gill M. 1987a. Metabolism of the lactating cow. I. Animal elements of a mechanistic model. Journal of Dairy Research 54, pp 77-105.

Baldwin R. L., Thornley J. H. M. and Beever D. E. 1987b. Metabolism of the lactating cow. II. Digestive elements of a mechanistic model. Journal of Dairy Research 54, pp 107-131.

Baldwin R. L., France J., Beever D. E., Gill M. and Thornley J. H. M.1987c. Metabolism of the lactating cow. III. Properties of mechanistic models suitable for evaluation of energetic relationships and factors involved in the partition of nutrients. Journal of Dairy Research 54, pp 133-145.

Beever D. E., France, J. and Theodorou M. K. 1986. Modelling of rumen function.) In: A. Neiman Sorensen (ed.) Agriculture New Developments and Future Perspectives in Research on Rumen Function, pp 109-140. Brussels: EEC.

Black J. L., Davies G. T. and Fleming J. F. 1993. Role of computer simulation in the application of knowledge to animal industries. Australian Journal of Agricultural Research 44, pp 541-555.

Dijkstra J. 1993. Mathematical Modelling and Integration of Rumen Fermentation Processes. Ph. D. Thesis, Department of Animal Nutrition, Wageningen Agricultural University, Haagsteeg 4,6708 PM Wageningen, The Netherlands.

Dobson A., Sellers A. F. and Shaw G. T. 1970. Absorption of water from isolated ventral sac of rumen of the cow. Journal of Applied Physiology 28, pp 100-104.

France J., Thornley J. H. M. and Beever D. E. 1982. A mathematical model of the rumen. Journal of Agricultural Science (Cambridge) 99, pp 343-353.

Murphy M. R., Baldwin R. L. and Koong L. J. 1983. A quantitative analysis of rumination patterns. Journal of Animal Science 56, pp 1236-1240.

Neal H. D. and Thornley J. H. M. 1983. The lactation curve in cattle: a mathematical model of the mammary gland. Journal of Agricultural Science 101, pp 389-400.

Russel J. B., O'Connor D. G., Van Soest P. J. and Sniffen C. J. 1992. A Net Carbohydrate and Protein System for Evaluating Cattle Diets: I. Ruminal Fermentation. Journal of Animal Science 70, pp 3551-3561.

Smith N. E. 1970. Quantitative simulation analyses of ruminant matabolic functions: basal; lactation; milk fat depression. Ph. D. Dissertation, University of California, Davis, CA.

9. APPENDIX TABLE OF ABREVIATIONS

AaPmV	Rate of amino acid incorporation into milk protein (moles d^{-1})
AcTmV	Milk fat synthesised from blood acetate and lipids
ATPG	ATP available to rumen microbes for growth (moles d^{-1})
BST	Bovine Somatotrophin (hormone)
BW	Body weight
cAa	Concentration of amino acids in blood (moles L^{-1})
Ce	Cellulose
cFA	Concentration of plasma lipids (includes NEFA plus triglycerides)
cFa	Concentration of blood lipid (moles L^{-1})
cGl	Concentration of glucose in blood (moles L^{-1})
CHOR1	Concentration of Catabolic Hormone One (arbitrary units)
cMiHb	The concentration of microbes associated with holocellulose
cRAa	Concentration of amino acids in rumen liquid (moles L^{-1})
CsAc	Rate of acetate formed from soluble carbohydrates
CSFV	Rate of soluble sugar conversion to VFA in rumen (moles d^{-1})
CSFVAC	Moles of acetate formed per mole soluble carbohydrate fermented
cTs	Concentration of triacylglycerol in adipose tissue (moles L^{-1})
DLhor	Rate at which the lactation hormones Lhor

	declines
DMILK	Rate of milk secretion (kg d^{-1})
EBW	Empty body weight
FaTmV	Rate of blood lipid (FA) incorporation into milk fat (moles d^{-1})
Fdfat	Added fat
fdLi	Plant lipids in feed
FDPIPI	Entry of insoluble protein into the rumen with small particles in feed (kg d^{-1})
FdSt1, FdSt2, FdSt3 and FdSt4	Parameters describing the fractional content of starch included in the four sets of feed specification parameters (dimension less unit in the interval 0 - 1).
FDSTHA	Rate of feed starch entry to small particle pool (kg d^{-1})
FGAM	Variable moderating factorial growth depending on ammonia concentration
FGFA	Variable moderating factorial growth due to ratio lipids from plants/other fat in feed
GlLmV	Rate of transformation of glucose to lactose in udder
HAMI	Microbes associated with starch
Hb	Holocellulose
HbMi	Microbes associated with holocellulose
Hc	Hemi-cellulose
HCIN	Enty of hemicellulose into the rumen with small particles in feed (kg d^{-1})
INS	Injected insulin as effector of glucose uptake (experimental option)
K	Rate constant
K1Actm	Affinity constant for for milk fat synthesis from acetate depending on concentration of

glucose in blood (0.001 moles l^{-1})

K1FaTm Affinity constant for for blood lipid to milk fat dependingon concentration of glucose in blood (0.0015 moles l^{-1})

K1TsFa Inhibition constant (default value 5.0E-4 moles L^{-1})

KAaLmV Affinity constant for amino acids for alphalactalbumin synthesis (default value 0.002 moles L^{-1})

KAaPmV Affinity constant for amino acids for milk protein synthesis (0.0021 moles L^{-1})

KAcTmV Affinity constant for acetate for milk fat synthesis (0.0018 moles L^{-1})

KCeCs Variable controlling rate of hydrolysis of cellulose default value 7.0 if the silage is mainly grass, 6.0 if it is mainly legumes and 9.0 if it is mainly corn silage

KfatHb Variable moderating effect of added fat, default value 0.03

KFaTmV Affinity constant for blood lipid to milk fat (5.0E-4 moles L^{-1})

KGlLmV Affinity constant for glucose conversion to lactose in udder (default value 0.003 moles L^{-1})

KLhor Degradation constant for lactation hormone

KLpSp Constant (default value 4.5 d^{-1})

Kmdeg Rate constant to decrease udder metabolic capacity when average retained milk is high

Kminh Factor defining inhibition of milk synthesis by retained milk in the udder

KMinh Factor defining inhibition of milk synthesis by milk

KmlkI Factor used to scale KMINH (kg)

KPiAa Variable controling rate of breakdown of

	insoluble protein to rumen amino acids
KRAaFv	Affinity constant of microbes for amino acid fermentation
KTsFaF	Affinity constant for body fat lipolysis (0.2 moles L^{-1})
Kudeg	Rate constant for degradation of udder enzymes which is not dependent on retained milk
KUdegM	Rate constant for degradation of udder enzymes which is dependent on retained milk
Kusyn	Michaelis Menten type constant defining effect of LHOR on rate of udder enzyme synthesis
LGDCHa	Lower gut (intestine) digestion coefficient for starch
LGHaGl	Rate of glucose formation due to digestion of starch in lower gut (intestine) (moles d^{-1})
Lhor	Lactation hormones which enhance synthesis of udder enzymes
Lp	Large particle pool
LPHCHC	Rate of formation of small particle hemicellulose in the rumen from breakdown of large particles (kg d^{-1})
LPPIPI	Rate of formation of small particle insoluble protein in the rumen from breakdown of large particles (kg d^{-1})
LpSp	Rate of large particle breakdown by rumination
M	Respective sub pool of microbes associated with starch, holocellulose and small particle microbes for insoluble protein
MiG	Microbial growth
Mlkmax	Maximum mammary capacity for milk (kg)
MWSt	Molecule weight for starch

NADPH2	Nicotine adenine diphosphate
P and ΣP_i	Total size of pool for respective nutrient in rumen
P1	Exponent or steepness parameter for insulin responses (2.0)
P_i	Size of sub pool i for respective nutrient in rumen
R	Rate of microbial hydrolysis
RAaFv	The fermentation rate of amino acids to volatile fatty acids
Rdm	Rumen dry matter
R_i	Rate of hydrolysis done by microbes in sub pool i
Rlv	Rumen liquid volume
RPH	Rumen pH
Rum	Factor defining time spent ruminating
RUMVOL	Volume of rumen
SpCeCs	The rate of hydrolysis of cellulose
SPHACS	Rate of hydrolysis of small particle starch (kg d^{-1})
SpHbCs	The rate of hydrolysis of holo-cellulose
SPHCCS	Rate of hydrolysis of small particle hemi-cellulose (kg d^{-1})
SPPIAA	Rate of hydrolysis of insoluble protein in small particles (kg d^{-1})
T3	Thyroid hormone expressed as multiple of basal/normal (arbitrary units)
THETA1	Steepness exponent for substrate (Ts) effects on lipolysis in adipose tissue (5.0)

THETA5	Steepness exponent for effect of retained milk on rate of udder enzyme degradation (10.0)
TsFaF1	Rate of body fat synthesis from acetate
Ucells	No of udder cells
Udeg	Degradation rate for udder enzymes
Uenz	Udder enzymes (arbitrary units)
Umave	Milk in udder - sort of rolling average (kg)
Umilk	Milk in udder (kg)
Usyn	Synthesis rate of udder enzymes
VAaPmV	Proportional constant for milk protein synthesis per unit Uenz
VAcTmV	Proportional constant for milk fat synthesis from acetate/unit of UENZ (0.00649 moles d^{-1})
WaMi	Mass of microbes in rumen fluid not attached to particles (kg)
VFaTmV	Proportional constant for blood lipid uptake for milk fat synthesis
VGlLmV	Proportional constant for glucose conversion to lactose/unit UENZ (default value 0.0039 moles d^{-1})
VRAaFv	Proportional constant for amino acid and peptide fermentation kg^{-1} microbes
VTsFaF	Proportional constant for lipolysis in adipose
Vusyn	Proportional constant for synthesis of udder enzymes (1.0 moles d^{-1})
YATP	Microbial growth yield coefficient (kg/mole)

II

PART II - DEVELOPMENT FOR EDUCATIONAL USE OF THE COW DIGESTION MODEL MOLLY

CONTENTS

1. BACKGROUND

Molly is the name of a cow digestion model developed by Baldwin et al (1987*a, b and c*). The purpose of this model was to describe how fodder is processed in a cow. As input, fodder of a certain composition was fed regularly to the cow. The digestion process is described by 20 constituents which can be followed in quantitative terms throughout the process. The conceptual form of Molly is presented in Figure 1.

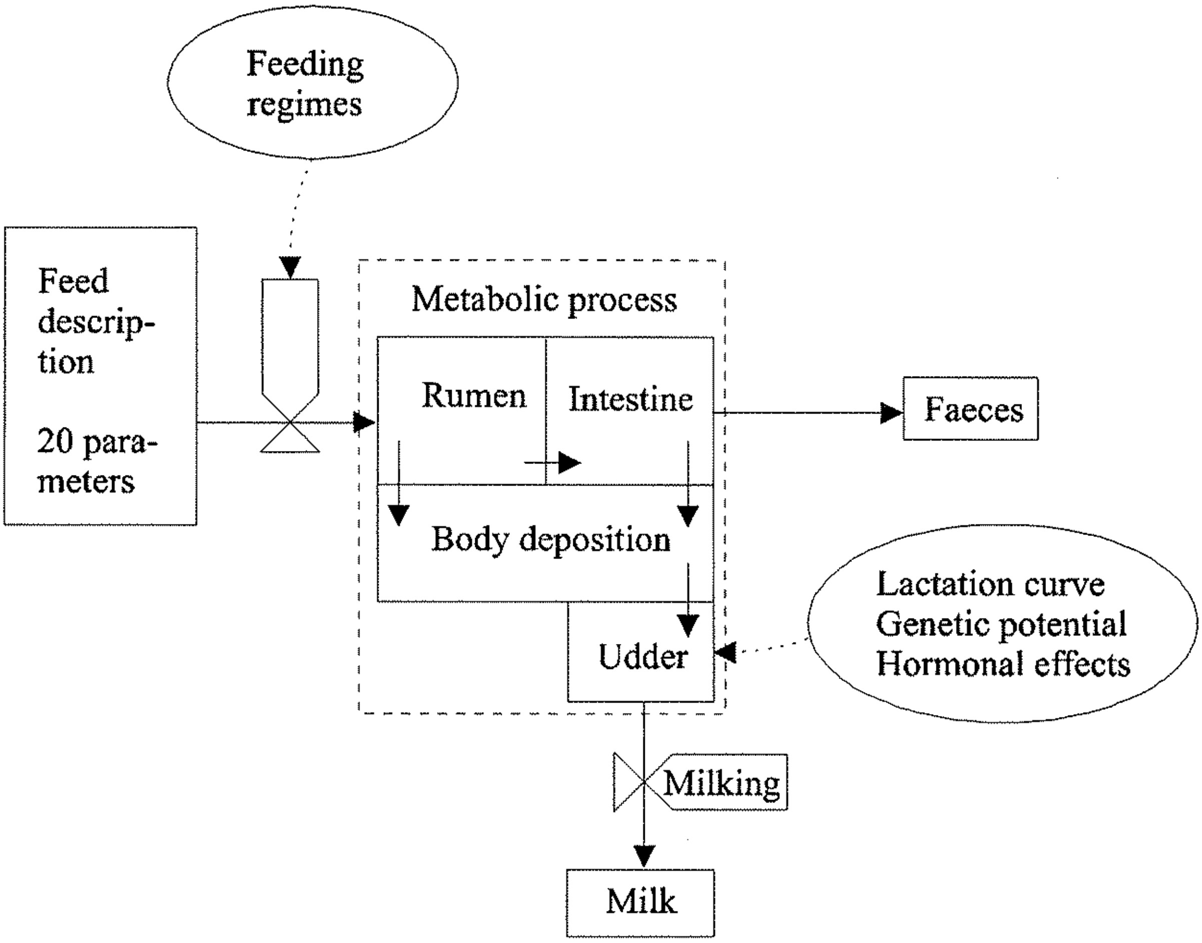

Figure 1. Conceptual description of Molly.

This model was implemented in the advanced continuous simulation language ACSL. In 1992 it was refined by Baldwin (Baldwin, personal communication). During 1993-94 the model, representing conditions in the United States, was adapted to Swedish conditions with respect to separate schedules for feeding of concentrate and forage. The digestion process in the rumen was also described in a refined model (Part I of this document). These extensions were all made in ACSL.

The Molly model is a powerful tool for research and experiments and for structuring and collecting knowledge on digestion in the cow. It is, however, complex and difficult to handle for non-experts. Therefore, it requires a person with expert knowledge in cow metabolism and good knowledge of the Molly model as well as skill in ACSL.

We judged the Molly model to be an excellent instrument for various teaching purposes provided the model could be reformed to become more easy to handle and overview. Therefore, a project to find a suitable educational form and to implement this in a suitable environment was carried out from 1995-99.

2 OBJECTIVES

This work was aimed at simplifying the use of the Molly cow feed digestion model for teaching at undergraduate or graduate levels and also to facilitate its use by interested nutritionists and researchers in related areas. The objectives were to implement the Molly model in a more modern computer environment and in a graphical and hierarchical form. The model would thus become more educational and more suited for demonstrations in the sense that it would become more transparent and easy to handle.

To translate these objectives into operational terms the following properties were considered desirable.

- The model must be *hierarchically structured* so that the whole process can be monitored and that the user can zoom into more detailed levels. This will improve overview as well as orientation.

- The model must be presented in a *graphical form based on states and flows* rather than in computer code.

- Experimenting with the model by *setting appropriate values to various parameters and initial conditions* must be made in a user-friendly way without the need for entering programme code and recompilation.

- The computer environment must support easy and *flexible presentation of results* in diagrams and figures.

- The computer environment must have *tools for handling pre-defined scenarios* by using macro capabilities.

3. CHOICE OF COMPUTER LANGUAGE

When this project started there were different types of computer languages for continuous simulation to choose between. One type of language follows the old CSSL convention (SCi Simulation, 1967). ACSL (Advanced Continuous Simulation Language, 1993) is a typical example of this. The structure of the simulation code into initial, dynamic, output blocks etc. is well specified. These languages are characterised by having a high speed of execution when the model is formulated, which can be of importance e.g. for parameter estimation or optimisation models. However the readability and overview of models written in this type of language suffers because the model representation is more close to machine code than the code for more modern languages. Above all, these languages do not have the integrated capabilities for graphical presentation of the model structure and consequently a hierarchical presentation of a model such as Molly in graphical terms is difficult to accomplish.

Another type of simulation language is that in the System Dynamics family consisting of languages like DYNAMO, Stella and Powersim (Richardsson and Pugh III, 1981). In these languages, dynamic modelling is performed by linking symbols of stocks and flows into a graphical structure without explicitly involving differential or integral equations. This was a revolutionary pedagogic invention first used by economists but which soon spread to biology, agriculture and many other sciences.

In the oldest of these languages, DYNAMO (Pugh III, 1983), the model is built as a graph by connecting stock, flow and a few other symbols on a piece of paper. Each symbol is then represented by one equation of code of a predescribed form. In the followers (Stella, 1990) and (Powersim, 1996) similar graphical symbols are used to build the structural diagram of the model directly on the computer screen using "click-and-draw" from a toolbar. From this diagram the simulation code is automatically generated, whereby the user can concentrate on the graphical representation of the model. Stella and Powersim are especially useful for teaching and for building small and medium scale models. However when building large models, structure and overview clearly become a problem because of the lack of hierarchical tools.

A third type of package for continuous system simulation was MATLAB/Simulink. MATLAB (Matrix Laboratory, 1997) is a general package for mathematical handling of matrices. To MATLAB there are a variety of toolboxes for different purposes. One of these toolboxes, (Simulink, 1990), is a package in which graphical symbols (associated to the theory of automatic control) are used to build the structure of the model. Simulink inherits the powerful tools for array handling from MATLAB and can handle different structural levels graphically. This gives the user the possibility to zoom in or out of the model. The model or one of its submodels represented by an icon can be opened by a mouse-click to display successively more detailed levels of a model. In this way the user can navigate through the large complicated structure to the details of interest without losing the orientation and overview. Furthermore, MATLAB also has powerful tools in the form of its M-files to handle scenarios, specify output etc. MATLAB also has strong and flexible facilities for result presentation in graphical and tabular form.

A disadvantage of MATLAB/Simulink is that the speed of execution is considerably slower than for the same models built in the CSSL type of languages. The main reason for this is that MATLAB works in an interpretative way - although the code can be transferred into c-code and compiled to execute 5 to 7 times faster. The compiled version of a Simulink model is still slower than the same model executed in ACSL. However, since the speed of computers is increasing with every new generation, the execution time becomes less and less of a problem although it may still be a problem when models are used for e.g. optimisation.

Judging pros and cons between different languages from a teaching point of view MATLAB/Simulink was chosen as the best platform for the Molly digestion model in the cow, mainly because of its graphical tools and its capacity to handle hierarchical structures. Its arrays-based philosophy was also an advantage in handling the large number of elements involved in the digestion process as well as its flexible graphical result presentation facilities.

4. THE IMPLEMENTATION OF MOLLY IN MATLAB/SIMULINK

In order to make the model more useful for teaching and demonstration purposes, a new architecture of the model was needed.

A direct transformation of the ACSL code into Simulink is presently not possible since the underlying philosophy, structure, tools and components differ completely.

The original structure of the model followed the ACSL standard with the sections Initial, Dynamic, Derivative, Discrete and Terminal. The Dynamic section normally consists of the two sub-sections derivative and discrete. However, the Discrete and Terminal sections are optional and not used in Molly. The initial section of Molly contains:

i) initialisation of the states (stocks), initialisation of physical and chemical constants,

ii) rescaling factors for some of the states for computational purposes associated with integration step length,

iii) definition of parameters describing feed composition and amounts of feed,

iv) parameters describing starting time and duration of consumption,

v) parameters defining hormonal status depending on and defining stage of lactation

vi) parameters defining genetic potential etc.

The derivative section contains statements to be integrated over time. Furthermore, the statements defining choice of algorithm for integration and integration step size and length of simulation time etc. are located in this section. The derivative section also contains various procedures to be calculated depending on feeding or milking strategy. For a more detailed description of the original model, the reader is referred to Chapter 16 in Baldwin (1995).

A direct transformation of the structure of the model as it is built in ACSL is not appropriate from a teaching perspective but is also impossible, since the structure and components in MATLAB/Simulink are completely different. Before describing the new design, therefore, we present the array handling and graphical tools in MATLAB/Simulink.

4.1 The graphical representation

Conceptually, MATLAB is built for handling matrices, i.e. vectors. The graphical representation of a model in Simulink is done by connecting various symbols on the computer screen by lines in form of arrows representing (arrays of) flows of information. There are many symbols available to generate a signal in Simulink. Figure 2 shows some symbols commonly used in Molly, namely a simple "constant", a "clock" representing the value of the simulation time and the "function" which returns the value of a mathematical expression including MATLAB functions and referring to variables or matrixes defined in the current MATLAB workspace or to the vector optionally connected to the input of the block.

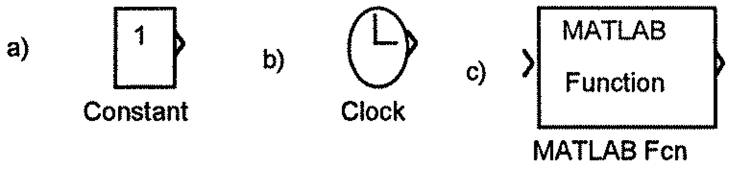

Figure 2. Icons of blocks commonly used as signal sources in the Simulink version of Molly. a) a simple constant, b) the simulation time as a signal, c) the MATLAB Fcn block which generates a signal or treats an input signal according to a mathematical expression referring to MATLAB functions or variables defined in MATLAB workspace or input signal(s) to the block.

Figure 3 shows some blocks in which the signals are treated mathematically in various ways. The Gain block simply amplifies the signal by a specified factor. There is a special Fcn block to combine several in-signals in form of an array **u** = {u(1), u(2)...u(n)} by any mathematical formula to a new signal. The value of an input signal, **u**, can be transferred into an output signal using a graphical Look-up table function. Values between input points are interpolated. A signal or an array of signals can also be differentiated or integrated over time by the use of a derivative or an integrator block.

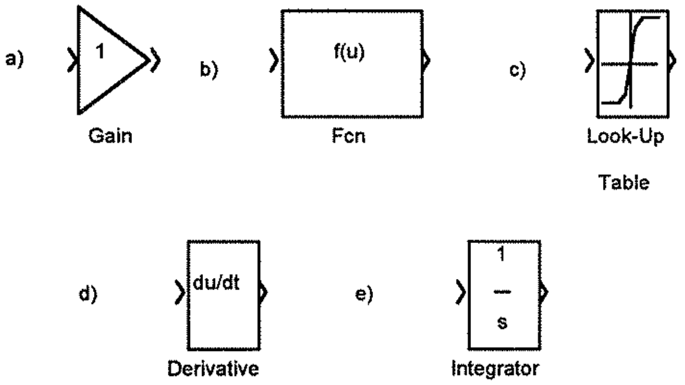

Figure 3. Examples of some commonly used icons of blocks in Molly in which the input signals are treated mathematically: a) the Gain block, b) the Fcn block, c) the Look-up table block, d) the Derivative block and e) the Integrator block.

The bottom layers of the model are composed of the symbols in figures 2 and 3. These components are grouped together in new symbols forming new components in subsequent higher layers of the model.

4.2 The hierarchical structure

The top layer of Molly implemented in Simulink consists of six blocks named; Feed Intake Generator, Rumen, Intestine, Absorption of Nutrients to Blood, Body Deposition and Udder. In Simulink it is possible to create special icons in the form of simple pictures of the graphical blocks containing the model. This feature has been used in the top level of Molly to give the model a pleasant appearance which associates to the organs in the body. See Fig. 4.

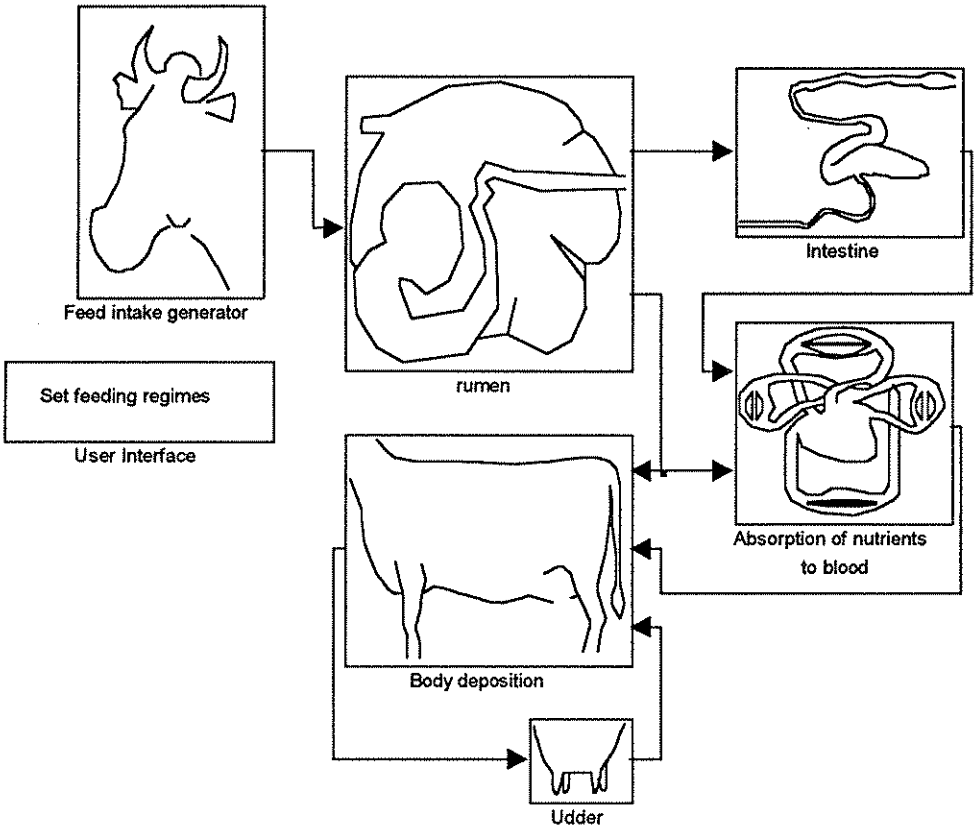

Figure 4. The top level of the Molly model in Simulink.

In Simulink Molly is hierarchically restructured into a number of layers depending on the complexity of different parts of the model. When a user clicks on a symbol a new window containing the subsequent lower layer of the component opens. The three layers of the ”Feed intake generator” is shown in Figure 5.

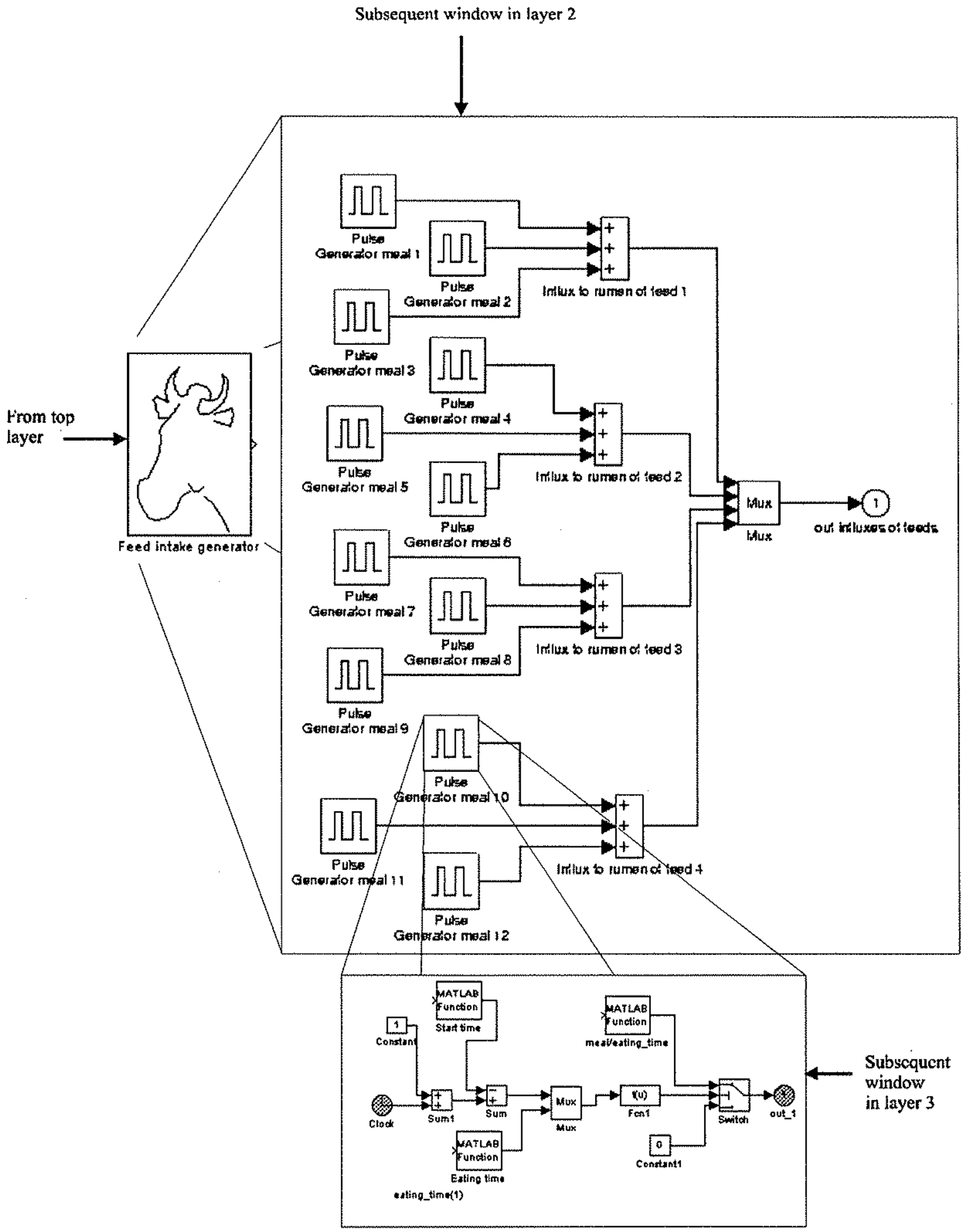

Figure 5. Clicking on the "Feed intake generator"opens the details at layer 2. Further, zoming in on "Pulse Generator meal 10" brings us to a third layer showing the details in Simulink symbols.

The windows containing the different layers of the model often cowers the whole screen. The hierarchical structure of the six parts of the model, not to be confused by their appearance on the screen, is shown in principle in figures 6 - 11.

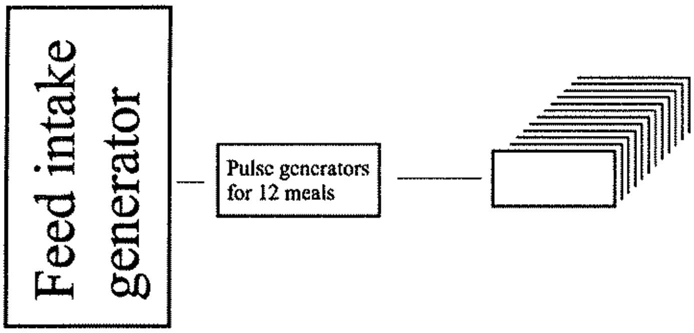

Figure 6. The structured hierarchy of the Feed intake generator part of Molly in Simulink.

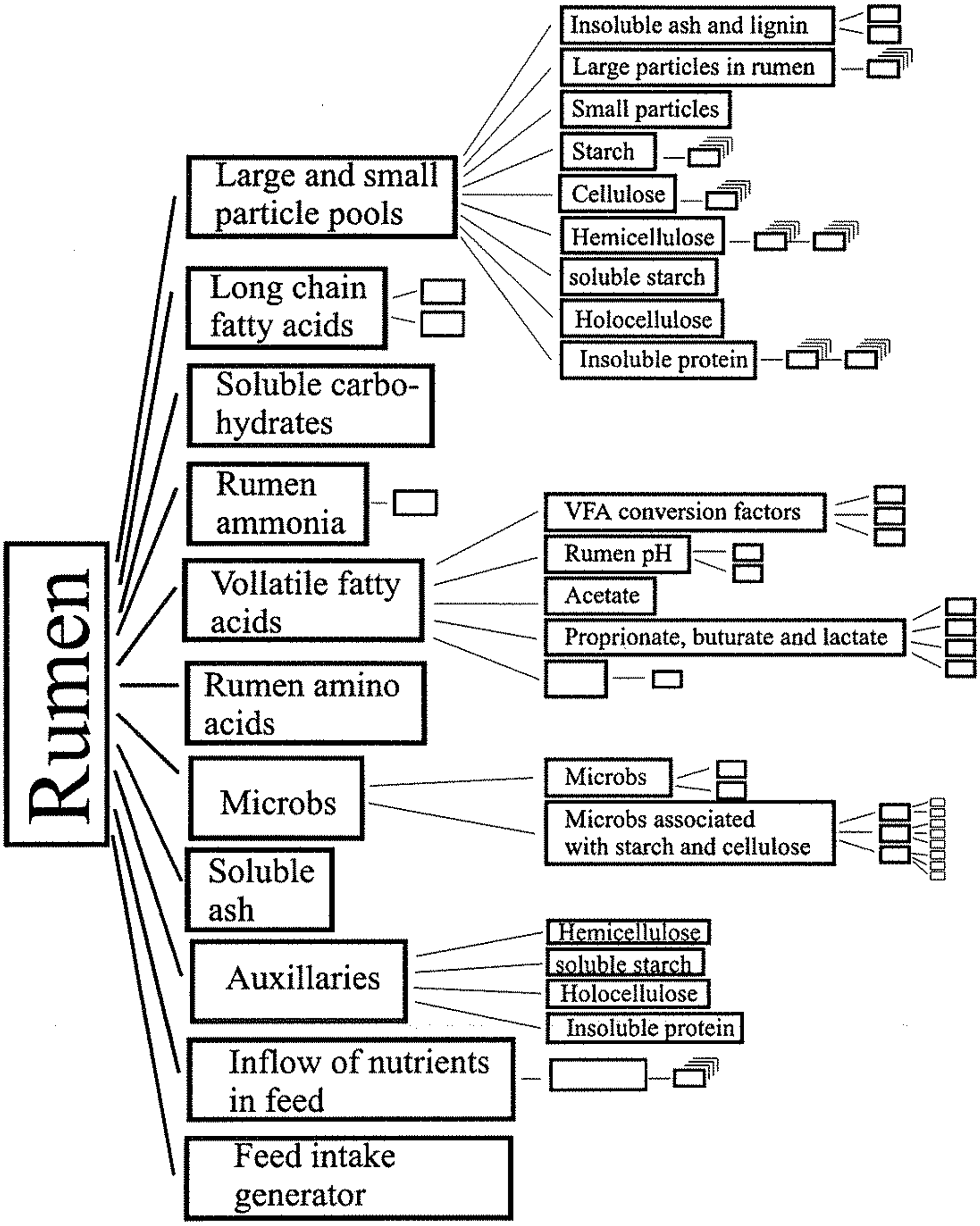

Figure 7. The structured hierarchy of the Rumen part ofMolly in Simulink.

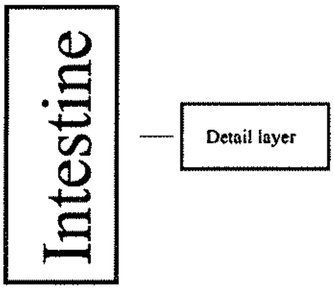

Figure 8. The structured hierarchy of the Intestine part of Molly in Simulink.

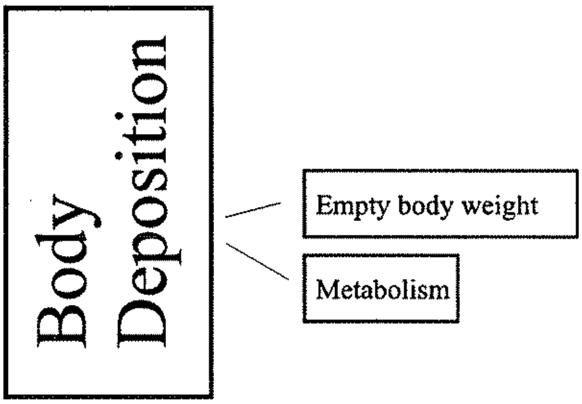

Figure 9. The structured hierarchy of the Body Deposition part of Molly in Simulink.

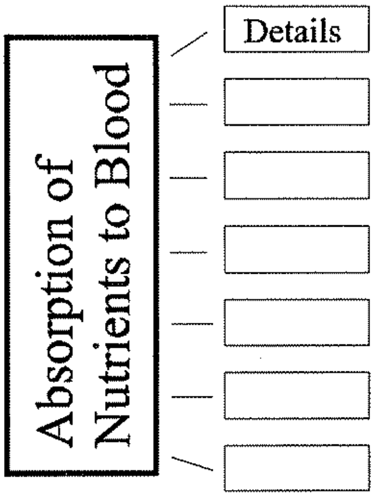

Figure 10. The structured hierarchy of the Absorption of Nutrients to Blood part of Molly in Simulink.

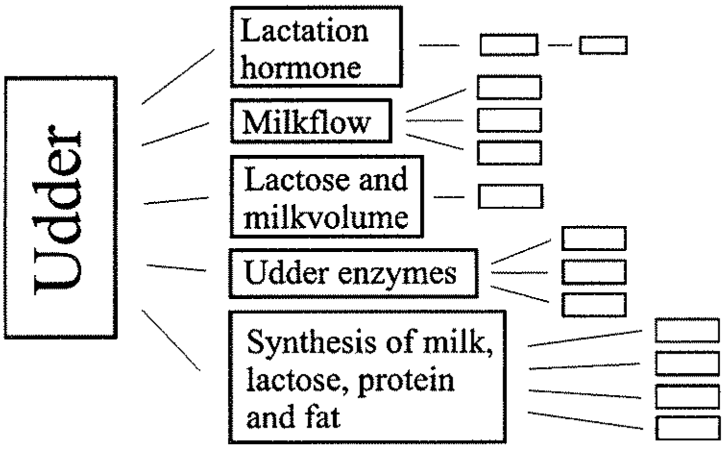

Figure 11. The structured hierarchy of the Udder part of Molly in Simulink.

In the more complicated parts of the top level, some effort has been made to give the user a rough overview of the structure of the respective part on the subsequent second level. Particularly for the Rumen part, the lay-out is designed to give the user an overview of the metabolic pathways in the rumen. Here, the concept of vectors was utilised since it was impractical and confusing to draw multiple lines between two components in the graphical presentation. It was, therefore, decided to represent the chemical components in a material flow by one vector (one line) rather than by many individual lines. For the sake of surveyability it was also decided that no flows of information other than those representing material or nutrient flows should be visible. Thus, the user of the model will have a clear view of the nutrient flow through the rumen model. For other flows of information the blocks "goto" and "from", which look like “address tags”, were used instead of lines to show their source and destination.

In other less complicated parts like the udder, arrows can represent any type of information flow and these layers have just been put together by the use of standard components. Below layer 2 of the model, there are one or several layers depending on model complexity, in which arrows can represent any type of information flow.

A comparison of the flows into the rumen as represented by Simulink and ACSL code is shown in Figure 12.

```
fDSc=
(fDSc1*DDMINmix1+fDSc2*DDMINmix2+fDSc3*DDMINmix3...
+fDSc4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDOa=
(fDOa1*DDMINmix1+fDOa2*DDMINmix2+fDOa3*DDMINmix3...
+fDOa4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDPe=
(fDPe1*DDMINmix1+fDPe2*DDMINmix2+fDPe3*DDMINmix3...
+fDPe4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDLa=
(fDLa1*DDMINmix1+fDLa2*DDMINmix2+fDLa3*DDMINmix3...
+fDLa4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDLi=
(fDLi1*DDMINmix1+fDLi2*DDMINmix2+fDLi3*DDMINmix3...
+fDLi4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDFat=
(fDFat1*DDMINmix1+fDFat2*DDMINmix2+fDFat3*DDMINmix3...
+fDFat4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDSt=
(fDSt1*DDMINmix1+fDSt2*DDMINmix2+fDSt3*DDMINmix3...
+fDSt4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDHc=
(fDHc1*DDMINmix1+fDHc2*DDMINmix2+fDHc3*DDMINmix3...
+fDHc4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDCe=
(fDCe1*DDMINmix1+fDCe2*DDMINmix2+fDCe3*DDMINmix3...
+fDCe4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDPs=(fDPs1*DDMINmix1+fDPs2*DDMINmix2+fDPs3*DDMINmix3.
..
+fDPs4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDPi= ( fDPi1*DDMINmix1+ fDPi2*DDMINmix2+ fDPi3*DDMINmix3...
+ fDPi4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
 fDNn=( fDNn1*DDMINmix1+ fDNn2*DDMINmix2+
fDNn3*DDMINmix3...
+ fDNn4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
 fDLg=(fDLg1*DDMINmix1+ fDLg2*DDMINmix2+ fDLg3*DDMINmix3...
+ fDLg4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
 fDAs= ( fDAs1*DDMINmix1+ fDAs2*DDMINmix2+
fDAs3*DDMINmix3...
+ fDAs4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDAi= (fDAi1*DDMINmix1+fDAi2*DDMINmix2+fDAi3*DDMINmix3...
+fDAi4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDAc= (fDAc1*DDMINmix1+fDAc2*DDMINmix2+fDAc3*DDMINmix3...
+fDAc4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDBu= (fDBu1*DDMINmix1+fDBu2*DDMINmix2+fDBu3*DDMINmix3...
+fDBu4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
FDUr= (FDUr1*DDMINmix1+FDUr2*DDMINmix2+FDUr3*DDMINmix3...
+FDUr4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
Fot= (Fot1*DDMINmix1+Fot2*DDMINmix2+Fot3*DDMINmix3...
+Fot4*DDMINmix4)/(DDMINmix1+DDMINmix2+DDMINmix3...
+DDMINmix4)
fDOM1=1.0-fDAi1-fDAs1
fDOM2=1.0-fDAi2-fDAs2
fDOM3=1.0-fDAi3-fDAs3
fDOM4=1.0-fDAi4-fDAs4
```

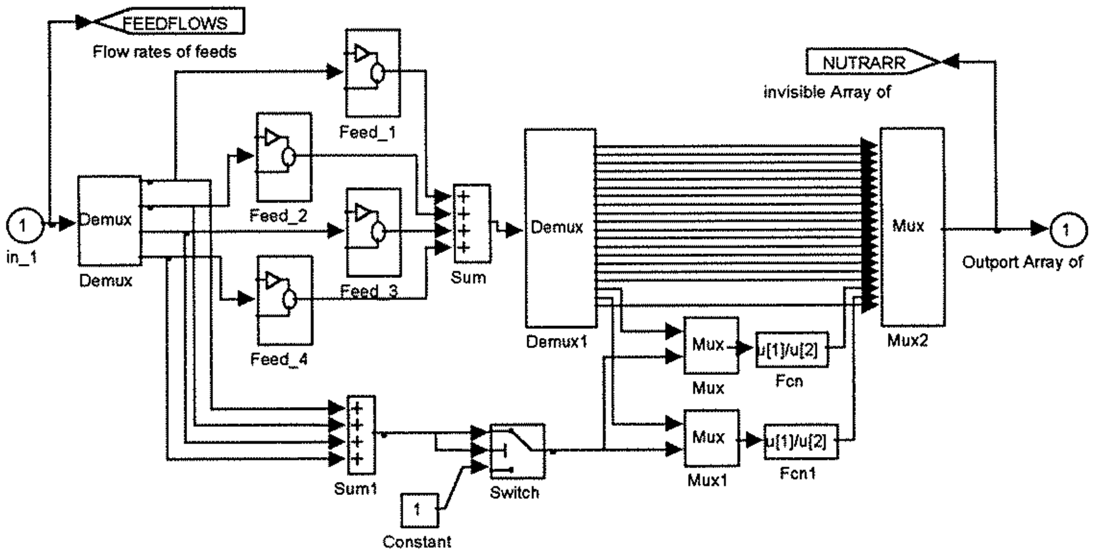

Figure 12. A comparison of coding in ACSL and Simulink. The flows of different nutrients into the rumen described in Simulink and the corresponding ACSL code.

4.3 Scaling and integration

The original model as it was received from Professor Baldwin in 1992 contained several optional feeding regimes and also optional equations for water dynamics etc. which the user of the model could turn on and off depending on interest. Several of these feeding regimes were related to questions concerning the dynamics of the model over a year, whereas others concerned the within-day dynamics. As a result of compromising between speed and accuracy of execution under these circumstances, the model was facilitated to be run with two different time scales (i.e. integration steps sizes). The model also contains a number of so called "stiff problems" as explained by France et al (1992) and Baldwin (1995). For this reason, some of the equations were rescaled in the ACSL model. These mechanisms were transformed into Simulink in a straight-forward way.

In ACSL there are a number of integration routines to use. A fourth order Runge-Kutta algorithm was used with fixed step size and worked properly. Similarly in Simulink there is a set of integration routines, including a fourth order Runge-Kutta with fixed step size now used as default in the Simulink model. However, the fourth order Runge-Kutta algorithm used in Simulink may not necessarily be the same as the one used in ACSL.

4.4 Parameter setting in ACSL and Simulink

Initialisation of necessary variables has to be performed before the actual simulation starts. This is done by defining all the necessary variables to default values in MATLAB workspace by an M-file which is automatically run when the model is opened and which assigns values to the parameters in the MATLAB workspace. Relevant parameters can be changed by the user from a user interface in form of a dialogue window which is implemented in Visual Basic and which communicates directly with MATLAB workspace through dynamic data exchange links. In ACSL the user had to change parameters one by one interactively by the use of the ACSL "set" variable command in the ACSL run window or by having the set parameter statements organised in command files which can be either pasted in the ACSL run window or executed directly from the ACSL environment. The command files into ACSL and the M-files in MATLAB work very much in the same way from the user's perspective and can be used to set up initial conditions before a run as well as to automatise any sequence of commands including running the model and presenting the results in plots or tables. In the ACSL environment, all commands during a runtime session are recorded in a file with the file extension ".log" and the same can be done during a MATLAB session by specifying a DIARY file. Both the log files in ACSL and the DIARY files in MATLAB can be edited to form a command or an M-file for later use.

4.5 Result presentation tools

The organisation of output from ACSL is done by the use of PREPARE statements executed before the execution of the model. Any variable which has been prepared can then be listed or plotted after execution of the model. MATLAB/Simulink has good tools for output handling. Here the outputs from a model can either be directed to matrixes in workspace or to files. In both languages it is possible to present the results as time plots, scatter plots or tabular outputs. It is also possible in both languages to use command files and M-files to automatise the generation of outputs from a model.

5. VERIFICATION

Since this work consists of a translation from one computer language to another for teaching purposes without changing the behaviour, the corresponding simulations of Molly in ACSL and in Simulink should give the same results. Hence it was possible (and necessary since bugs always occur in any programming work) to verify the Simulink version of Molly by checking that results were the same (within limits of the accuracy of integration algorithms etc.) as for the ACSL version. The input parameters to the part of Simulink Molly to be tested were recorded from the ACSL Molly. If the output from the tested part was the same as for the ACSL version of Molly then the tested part was verified. If the output was not the same, the tested part contained bugs and had to be debugged until the results from the two models agreed. So far most of the Simulink implementation of Molly, namely the parts contained in the Rumen block and the Feed Intake Generator block, has been successfully verified but the remaining parts have still to be tested further.

6. RUNNING MOLLY IN SIMULINK

6.1 Introduction

In order to run Molly in MATLAB/ Simulink, the Molly II model with its corresponding files first has to be correctly installed on a PC. Before opening Molly, the user has to start MATLAB from Windows and the toolbox Simulink by writing "simulink" in the MATLAB command window. The Model "Molly II" is opened by choosing **Open** from the **File** pull down menu and specifying the library according to Figures 13 and 14.

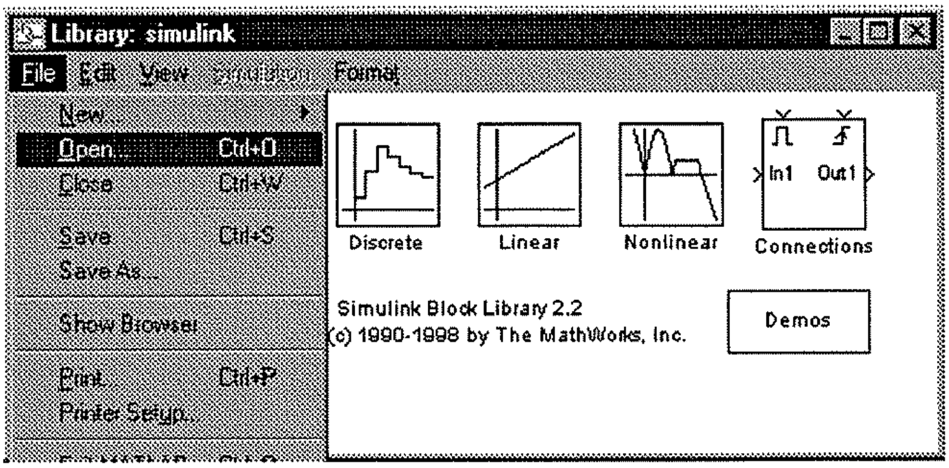

Figure 13. The file Open command in the Simulink window.

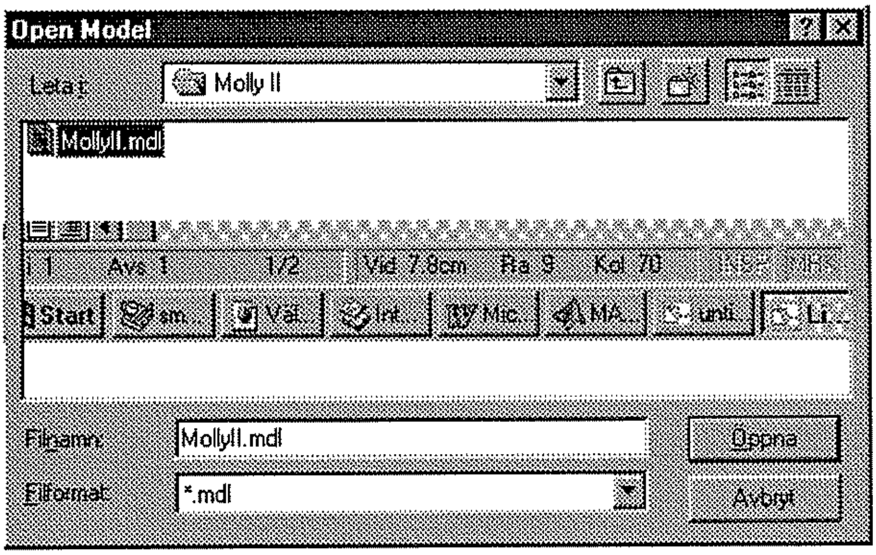

Figure 14. Opening the model MollyII.mdl from Simulink.
This opens the MollyII window as shown in Figure 15.

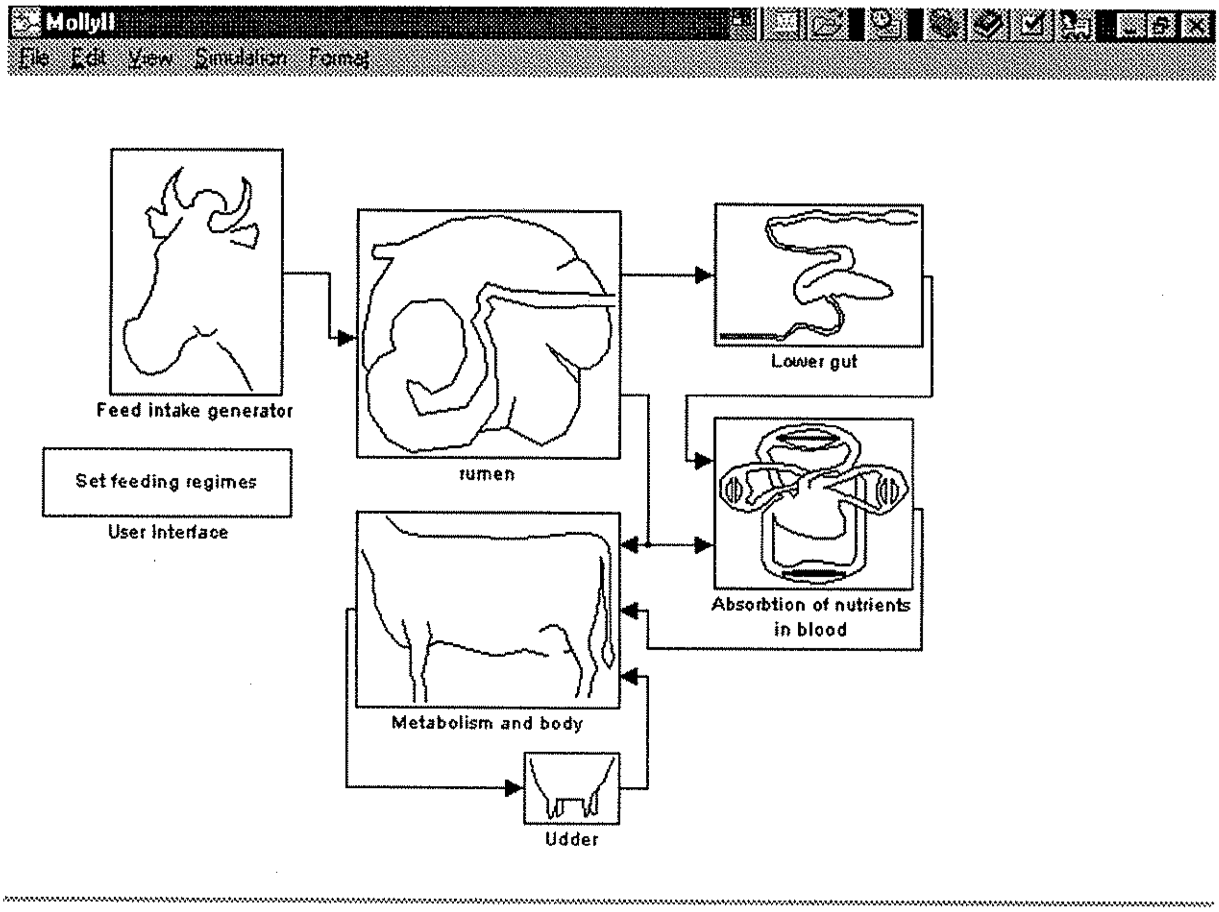

Figure 15. The screen after the model Molly II is opened and ready for use.

As the model is opened, an M-file containing the default values of the parameters in the MATLAB workspace for the base scenario of Molly II is automatically executed. If desired, the user can start the base scenario simulation immediately after the model is opened. If the user wants to specify his own scenario, this can be done in a number of ways. The relevant parameters in workspace can be changed directly from the MATLAB command window, an M-file can be edited and prepared for each scenario or user-interfaces can be used. As seen in Figure 15 there is a block or an icon labelled ”Set feeding regimes, User interface”. When this icon is double-clicked, a user interface in form of a pull down menu window appears on the screen. Se Figure 16.

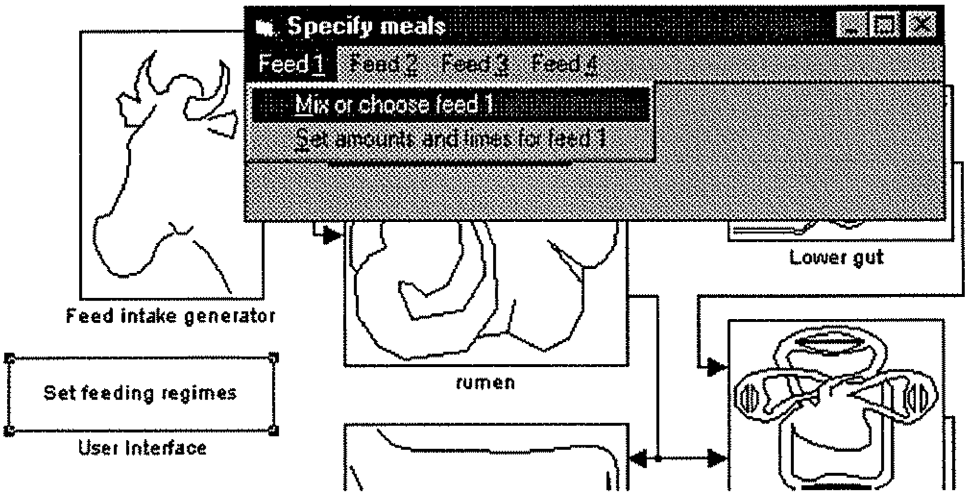

Figure 16. The top level window of the user interface "Set feeding regimes". From this window a user can compose or choose four different feeds, see Figure 17, which each can be fed at specified times and durations up to three times a day.

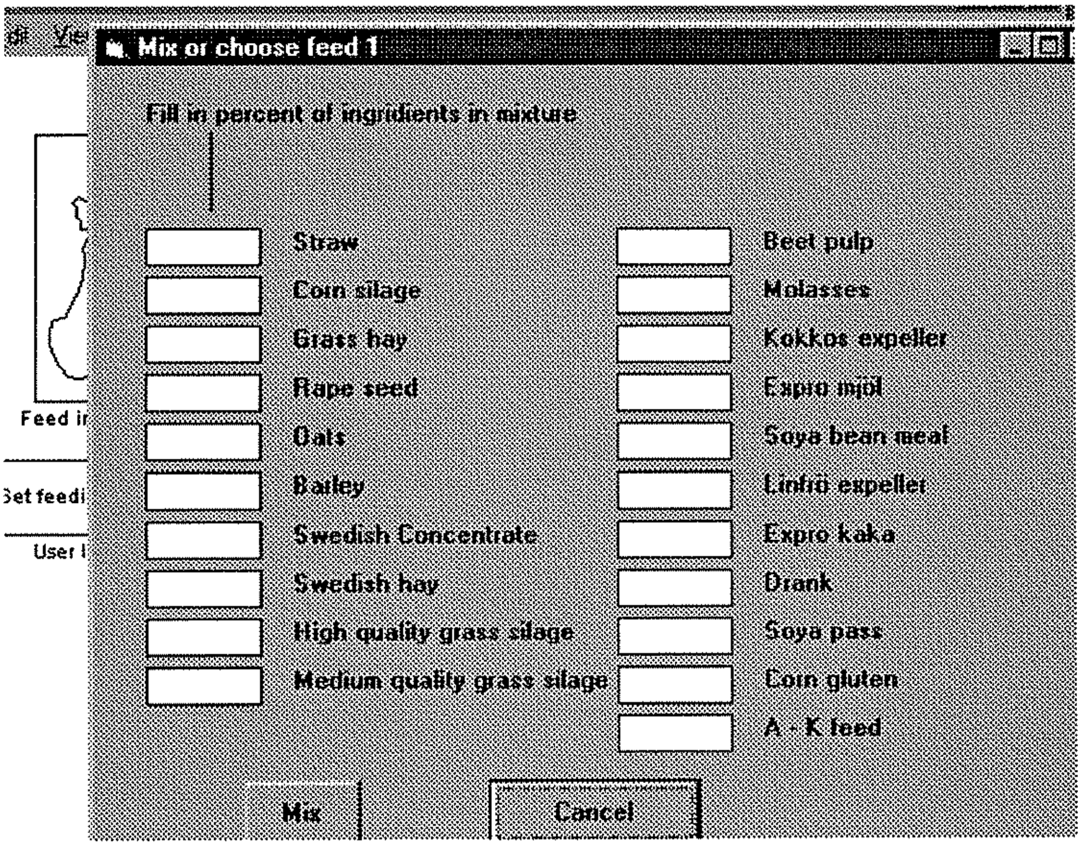
Fill in percent of ingridients in mixture

Figure 17. The sub-menu for mixing or choosing feeds.

There is also a similar interface for milking regimes. Here the user can choose to milk the model cow continuously at a flow rate proportional to the amount of milk in the udder, specify a milking pattern in which the udder is emptied except for a specified percentage of residual milk or choose to milk a fixed amount at each milking time. Time and duration for up to six milkings per day can be specified.

The user interfaces are programmed in Visual Basic and communicate, i. e. read and set the appropriate variables in the current MATLAB workspace, through dynamic data exchange links (DDE).

6.2 Initial conditions

Settings of initial values of the state variables (stocks) in Simulink are defined by clicking on the graphical symbol for the respective integrator and specifying a number or variable name in the field "initial conditions" in a dialogue box. This means that changing the settings of the initial values specified as numbers has to be done interactively with the graphical representation of the model, whereas initial conditions specified as variables have to be changed in MATLAB workspace in the same way as the other variables as described above.

6.3 Experimental conditions

Choosing the experimental conditions such as the length of simulation time, step size and choice of integration algorithm is done by choosing "Parameters" in the "Simulation" pull down menu in the Simulink window containing the graphical representation of the model. Default values in the model are a stop time of 14 days and a fixed stepped Runge-Kutta 4 algorithm with a step size 0.001 of the time unit one day.

6.4 Run

When all experimental conditions are set the model is executed by choosing "Start" in the "Simulation" pull down menu.

6.5 Output

The variables in a Simulink model to be plotted or tabulated have to be sent to MATLAB workspace or to a file. In the Simulink verson of Molly, the variables are sent to MATLAB workspace by use of the Simulink "To Workspace" block. Hence variables can be plotted against time, against each other in a phase plane or tabulated.

7. DISCUSSION

The model Molly has now been implemented in a *graphical form based on states and flows* and *hierarchically structured* so that a user can now get a good overview of the nutrient flow through the parts of the higher levels of the model and at the same time zoom way down to the details at the lower levels of the model without losing orientation. Interfaces to the model have been implemented (in Visual Basic) so that it is easy for a user to mix and choose from a menu of common Swedish feed ingredients and to experiment with different milking schedules and routines. For systematic experimentation, another possibility is to set the appropriate parameters by executing especially edited M-files and saving them by name for later review and modifications. MATLAB's graphical facilities are available for use with the outputs from the Simulink version of Molly. It is also easy to add on MATLAB routines for various experiments, presentations and post-calculations etc.

It should be stated, however, that this work is not a comparison between the two simulation languages MATLAB/ Simulink and ACSL. It is a report of an attempt to further develop the model Molly from a teaching perspective. A few comments on the similarities and differences between the two languages are possible:

- MATLAB Simulink is a tool for building graphical and executable dynamic models and is excellent for handling of hierarchical structures. ACSL has a graphic modeller of which the author have no experience.

- MATLAB is a programme package for handling and calculation of matrices. Simulink and ACSL both are tools for continuous simulation. ACSL offers a special programme which interfaces ACSL to MATLAB and which has not been used by the author.

- Both ACSL and MATLAB have macro capabilities i. e. command files and M-files which work in much the same way. They consist of a series of run time commands in text format which are executed batch-wise and can be used for similar purposes.

- MATLAB and Simulink are interpreted, whereas ACSL is translated to FORTRAN and compiled. Loops are slow in MATLAB. This greatly affects the execution speed and an ACSL model executes much faster than the same model formulated in Simulink.

The slow execution of the Simulink version of the model is a disadvantage. However, Simulink has an add on facility in which the formulated models can be compiled to machine-code. In the future when the model is updated to be compiled, the solution speed will become approximately 10 times faster. The speed will also become less and less of a problem because of the ever increasing speed in new generations of computers. Solution speed is of less importance in teaching use, for example in demonstrations and when finished results are presented.

The status of the work of translation of the model into Simulink is the following: The structure of the model has been formulated graphically and hierarchically in Simulink. The major part of the model i. e. the Feed Intake Generator and the Rumen part have been debugged and verified. Although the model is working, some intended facilities and extensions are not yet fully implemented. Thus the Feed Intake Generator so far only contains one feeding regime, namely the one with multiple feedings per day. Only the Feed Intake Generator and Rumen have been debugged and verified.

For further work it is suggested that all the parts of the model be debugged and verified. It is also suggested that arrows representing information flow other than chemical or physical flow are replaced by the "Go to" and "From" blocks, which makes them visible as address tags rather than arrows also in the lower levels of the model and not only in the higher levels. All the old building blocks from older versions of Simulink which prevent the model from being compiled should be replaced by their newer forms in order to speed up the execution of the model. It is also suggested that routines for more user-friendly automatic plotting and printing are prepared for inexperienced users. Exercises and scenarios for different teaching purposes have to be prepared before students can be allowed to do experiments on the model.

A real challenge would be to have a model like this presented on the Internet, with objectives and restrictions concerning the scope of the model as a map over the current knowledge in the field of feed digestion in the cow. Researchers all over the world could then send in their suggestions for improvements to the model and it could be changed in time with the progress of research.

8. REFERENCES

Advanced Continuous Simulation Language (ACSL). 1993. Reference Manual. Mitchell and Gauthier, Assoc. Inc.

Baldwin R. L. 1995. Modelling Ruminant Digestion and Metabolism. 1th Edn. Chapman and Hall, London.

Baldwin R. L., France J. and Gill M. 1987*a*. Metabolism of the lactating cow. I. Animal elements of a mechanistic model. Journal of Dairy Research 54, pp 77-105.

Baldwin R. L., Thornley J. H. M. and Beever, D. E. 1987*b*. Metabolism of the lactating cow. II. Digestive elements of a mechanistic model. Journal of Dairy Research 54, pp 107-131.

Baldwin R. L., France J., Beever D. E., Gill, M. and Thornley J. H. M. 1987*c*. Metabolism of the lactating cow. III. Properties of mechanistic models suitable for evaluation of energetic relationships and factors involved in the partition of nutrients. Journal of Dairy Research 54, pp 133-145.

France J., Thornley J. H. M., Baldwin R. L. and Crist K. A. 1992. On solving stiff equations with reference to simulating ruminant metabolism. Journal Theoretical Biology 156, pp 525.

Format Subcommittee of the Sci Simulation Software Committee. 1967. The Sci Continuous System Simulation Language (CSSL). Sci Simulation, Vol 9, No. 6, pp 281-303.

MATLAB. The Math Works Inc. 1997.

Powersim – Reference Manual. Powersim AS. 1996.

Pugh III A.L. 1983. DYNAMO User's Manual. The MIT Press. Cambridge, Massachusetts.

Richardson G.P. and Pugh III A.L. 1981. Introduction to System Dynamics Modelling with DYNAMO. The MIT Press. Cambridge, Massachusetts.

Simulink. Dynamic System Simulation for MATLAB. The Math Works Inc. 1990.

Stella. High Performance Systems. Inc. 1990.