Lehrstuhl für Maschinen- und Apparatekunde der Technischen Universität München

MODELLING AND SIMULATION OF COW LOCOMOTION FOR DYNAMIC WEIGHING IN MODERN DAIRY FARMING

Dragan Cveticanin

Vollständiger Abdruck der von der Fakultät Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt der Technischen Universität München zur Erlangung des akademischen Grades eines Doktor-Ingenieurs (Dr.-Ing.) genehmigten Dissertation.

Vorsitzender:

Univ.-Prof. Dr.agr. Dr.agr.habil. Hermann Auernhammer

Prüfer der Dissertation:

1. Univ.-Prof. Dr.-Ing. Karl Sommer
2. Univ.-Prof. Dr.-Ing. Dr.h.c. Karl-Theodor Renius, i.R.
3. Prof. Dr. Milos Tesic (Universität Novi Sad, Serbien und Montenegro)

Die Dissertation wurde am 16.06.2004 bei der Technischen Universität München eingereicht und durch die Fakultät Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt am 25.11.2004 angenommen.

CONTENT

Nomenclature

kg	Kilogram
m	Meter
s	Second
m^2	Meter square
m/s	Meter per second
ANFIS	Adaptive network based fuzzy inference system
RBNN	Radial basis neural network
PC	Personal computer
ID	Identification
exp	Exponential
i	Imaginary number
Hz	Hertz
dB	Decibel
MF	Membership function
Tc	Crossing time
MPE	Maximal percentage error
APE	Average percentage error

1 Introduction

Nowadays, the objective of the contemporary dairy farming is to produce high quantity of milk with the low production costs. It means that the production of milk pro day has to be maximized and the expenses need to be minimized. At the same time the quality of milk has to satisfy very high standard, which is expected by the consumers. To satisfy all of these requirements is not an easy task. The problem is quite complex and many researchers and engineers are dealing with this topic. To get the highest milk yield and quality of milk for each cow with minimal investment the farmers maximally exploit the animals and feed them economically by preserving good health condition of the animals. It means that a strict control of the quality and quantity of food intake is necessary as it influences the milk production and also the milk producing cost. The production is optimized by choosing the right food mixture and amount for each animal. The investigations show that the right individual nutrition regime may not be selected based on the measured daily milk yields as it might be insufficient and misleading. Therefore, another measurable performance parameter that helps in individual concentrates supplementation decision needs to be chosen. The body weight of the cattle is shown to be a reasonable and available parameter. Dairy cow body weight is a useful parameter for husbandry decisions along lactation. Additionally, the data of body weight is important information of health condition of the cow. It seems that the information on the milk yield and body mass may be the correct basis not only for feeding strategy but also on the health condition of the cattle. Namely, the loss of body weight and the changes in the milk yield indicate the early symptom of disease. Based on this statement it is concluded that the need of overlook of the daily milk yield and the body weight for each animal is necessary.

In order to gather all necessary data for modern dairy farming the individual identification of cows has to be introduced. The European Union countries are encouraging the farmers to mark the cattle. The marking of cows is made with electrical transponders attached on the collar or implants in the ear or rumen. To obtain the data of each individual cow the transponder antennas are positioned in the milking parlous, feeding stations and weighing devices.

To get the information of the milk yield of an individual cow, first, in the milking parlour the number from the transponder is read. Then the milking process begins. The electronic in the milking device records the amount of given milk and the cow number. Such identification systems and milking installations exist in commercial use. Many companies like Westfalia, DeLaval, etc., are widely recognised as major dairy equipment manufacturers and suppliers.

The data of cow weight is obtained by measuring on the weighing device. In modern farming where the cows are treated individually it is necessary to know the weight of each cow. To get the information which cattle is measured the number of the cow is recorded from the transponder.

Locations suitable for weighing cows are the return alley of the milking parlour and the concentrate feeding station. Weighing at the feeding station is not practical for large herds, since each feeder needs to have a weighing device and the body weight of the cow without milk is of interest. The weighing scales are frequently placed in the exit corridor of the milking parlour.

There are three types of scales: manual (mechanical), semi-automatic (with doors) and automatic weighing scales (commonly referred to as walk-through weighers). The decision on what kind of weigher would be preferable depends on the number of cows in the herd and milking parlour. The majority farms are equipped with stationary milking parlours and just a few are rotary. After leaving the stationary milking parlour cows are commonly moving as a tight bunch in the exit walkway. The manual and the semi-automatic weighers are usually there installed. The advantage of these weighers is in their simplicity but they have many disadvantages and do not satisfy the requirements of modern and economical farming. Namely, the manual weighing scale requires labour and a long measurement time. The semi automatic weigher is an improved one and it consumes less time than the static measuring and requires less human assistance. The both weighers stop the advance of the cows and eventually block the milking process. In spite of that, for rotary parlours the manual or the semi-automatic weighing scale could be a good weighing solution due to the separate exit of the animals.

For the modern milk production where the goal is to automate all the processes on dairy farm, an automatic weigher is the suggested solution. It eliminates the lacks of manual and semi-automatic weighing. The automatic weigher works without worker help and stopping of animal on the scale. The weighing process is done in a very short time without disturbing normal animal movement. The measured data is stored in a memory for the certain cow due to recording of the transponder number. In this work a new construction of automatic weigher based on the previous ones is suggested. Also, three methods dealing with body weight recognition from measured data are proposed. In the next chapter the most important automatic weighers are reviewed.

2 Stand of science and technique

In this part the previous research in the field of dynamic weighing is presented. The achievements in the field of modelling of human being locomotion are reviewed. The fuzzy logic usage in function approximation is analysed, as well as, the possibilities of the neural networks in classification.

2.1 Walk-through weighing scale development

The first weighing scales used were manual weighing scales that weighted animas while standing on the platform. At least one man was needed to do the measurements and to organize animal movement. As the dairy farming was developing into a complex milk production the need for faster and simpler procedure of body weight measurement was increasing. Consequently, semi-automatic weighing scale with doors and walk-through weighing scale were designed. The weighers with doors were similar to manual scales just that the procedure of weighing was partially automated. The doors of the scale were automatically closed as the animal stepped on the weighing platform. The animal was kept on the scale long enough to perform good reading of the body weight. Due to the long time needed for body weight measurement and often door failure the semi-automatic weighing was not preferred. Consequently, walk-through weighing scales were developed.

The first automatic weighers appeared at the end of seventies of the twentieth century. D.E. Filby, M.J.B. Turner and M.J. Street gave the initial investigation in this field in 1979. The paper (FILBY et al., 1979 [19]) discussed the novelty in automatic walk-through weighing. Since that time many researchers were and are working in developing an accurate walk-through weighing scale. The innovations in automatic weighers are explained in a number of scientific papers. All of them are following the same route in creating a better weighing device and all the following researches are the improvement of the previous results and represent the alternative solutions. Some of the main results will be shown in this chapter.

In the work of FILBY et al. (1979) [19] the weighing platform had the length of 3 meters. The measured force during cow stride over the weigher was averaged and the peak of the signal was recorded as new measured body mass. However, if the new measured mass was bigger than allowed deviation of +/- 30 kilograms compared to reference body weight, the results were rejected. The useful capture rate was 60.5 %. The method gave better results for big and slow than for young and fast cows. The method was applicable only for the case when cow was alone on the scale, but for fast crossing and crowding on the weigher it could not give satisfactory results.

IPEMA et al. (1987) [35] showed that the weighing platform could be mounted on the floor of concentrate feeding station on dairy farm. Namely, the cow is mostly motionless during feeding so the running mean is sufficient in signal processing. The authors suggested that for commercial purposes on large dairy farms all the concentrate feeding stations would have to be equipped with such weighing scales. This would lead to higher investments but to an improvement in cow body weight measurements.

The measuring of the weight during feeding is also applied for pigs. WILLIAMS et al. (1996) [86] considered the body weight measurement at a single space feeder. A small electronic weighing platform was positioned in front of the feeder. The weigher measured the weight of pig front legs during eating. The real time method with Kalman filter was used for signal processing. The standard deviation of the error between the whole weight of the pig and the daily weight derived from a linear interpolation between first and fifth day was 1.55 kilograms. The disadvantage of the front legs weighing was in estimation of the whole body weight. There was time dependence in proportion between front legs and whole weight of the animal. Consequently, measuring the whole weight of the pig periodically (e.g. every 14 days) was required.

The main disadvantage of measuring of animal during feeding is that it requires the measuring to be at the same time every day. Namely, the body weight can oscillate considerably during the day. The weight of the cattle fluctuates between 30 and 40 kilograms during the day due to large intakes of food and water and outputs of milk, urine and faeces (PEIPER et al., 1993 [58], REN et al., 1992 [62], MALTZ et al., 1990 [45], MALTZ et al., 1992 [46]). If the measuring is in different time the obtained results may be interpreted falsely. As the body weight does not vary much from day to day at the same time of the day the weighing of the animals at the same time every day reduces this lack. In order to eliminate this problem and to interpret the changes in body weight correctly, it was suggested that the weights should be compared at the same time of the day. To ensure the cow to be fed at the same time every day and to be measured at the same time is not an easy task. Due to this disadvantage the measuring during the feeding had to be eliminated.

Since it was known that the weighing after milking, which occurs at approximately the same time each day, was more reliable than the cattle weighing at the feeding station the new methods of automatic weighing were introduced (PEIPER et al., 1993 [58], REN et al., 1992 [62]).

REN et al. (1992) [62] created an automatic weigher for mass measurement after milking of dairy cows. The walk-through weigher consisted of a 2.1 meters long platform. The platform was shorter than the previous mentioned to obtain the lower possibility of cow crowding on the scale. The measured weighing force signal was compared with reference weight. There were two reference weights for every cow: one for morning and one for evening milking, as the weights of an animal vary during the day much more than from day to day for the same time of the day. If the measured data points were within the 5% of reference value, they were recorded and averaged. Each recorded weight was impaired with the identification number of the animal coming from the identification system. After each milking, 78% of measured cows data was acceptable. For weekly averaging of measured weights, the achieved accuracy was 2% compared to statically measured data. The weighing after milking gave better results than those obtained during feeding in the terms of efficiency and constancy of results, but problems for fast crossing and crowding on the weighing device occurred.

To eliminate the previous problems PEIPER et al. (1993) [57] designed an automatic walk-through weighing scale with a slow down step and modified the signal processing. The platform had the dimensions of 2.5 meters in length and 1.2 meters in width. A 1.5 x 1.0 m^2 slow down step plate was positioned in front of the main scale in order to reduce high variations of the signal caused by quick steps and to separate the animals crossing the scale. The identification system antenna was placed on the two-third of the way toward the end of the scale. The identification signals were received just before the cow stepped on the scale with its full weight. Measured reaction force between the platform and cow feet was averaged. Using a special algorithm the real weight of the cow from a vector of averaged weight values was selected. It was based on the comparison of weighted average to the reference weight of the cow for allowed difference of 30 kilograms. For three daily measurements an average of 76.5% successful weights was obtained at least once a day. The accuracy of 1.5% was achieved for monthly comparison of the computer and manual weighing. The main disadvantage was that the weighing device was inadequate for daily weighing of the dairy cows since the high inaccuracy of the calculated weights. Also, the physical barrier which was placed to slow down the animal movement on the scale required additional measurements time. The weighing process could not be implemented as desired on automated dairy farms.

Based on the previous results a new conception of automatic weigher is developed and described in this paper.

2.2 Modelling of human locomotion

To understand the connection between the weight of the cattle and the force in the foot the process of walking has to be modelled.
To understand fully the effects on produced force between animal feet and ground during walking, the locomotion of cattle need to be investigated. The accent is on the walking mechanism. There are only a few books and papers dealing with the animal locomotion (see ALEXANDER, 1992 [4]). Most of the research is concentrated on producing robots that imitate bipedal and quadrupedal motion (TODD, 1985 [77], RAIBERT, 1986 [61]). They are not directly specialized in the mechanics and modelling of cattle walk. Much more investigation is done for bipedal locomotion (MCGEER, 1990 [50], VUKOBRATOVIC et al., 1990 [79], ALEXANDER, 1995 [5], GARCIA et al., 1998 [23]). Therefore, the review of works dealing with human movement and similarities with cow movement are here presented.
The simplest of all models of walking is explained in the work of ALEXANDER (1976) [3]. It consists of point mass moving on rigid, massless legs (Figure 2.1a). While a foot is on the ground, the mass moves along an arc of circle with centre on the foot. The mass rises and falls as the model walks. This is similar to the motion of centre of mass of real people (MARGARIA, 1976 [47]). The minimal biped gives some understanding of the energy changes during walking. While one foot is on the ground, the model behaves as inverted pendulum. The potential energy rises as the point mass rises while the kinetic energy drops. The energy is restored as the mass falls again. The work has to be performed during changeover of active and passive legs. Kinetic energy is lost in inelastic impact between foot and ground (MCGEER, 1990a [49]). The muscles restore the energy loss and do the positive and negative work, degrading mechanical energy to heat. According to ALEXANDER (1995) [5] during human walking each foot is on the ground for more than half the time. There are stages when both feet are touching the ground simultaneously. In walking, the knee remains more or less straight while the foot of the same leg is on the ground. To obtain the time period of a stride it is necessary to know the velocity of walking. THORSTENSSON et al. (1987) [76] denoted that for the adults walk the

maximal speed is up to 3 m/s. For an extremely slow walk at the velocity of 0.4 m/s the stride period for adult human being is about 2 seconds. For the brisk walking at the velocity of 1.6 m/s the stride period is 1 second (ZARRUGH et al., 1974 [88]). These values represent the periods of vibrations of the inverse pendulums which are the models of bipedal motion.

The model, represented by MCGEER (1990a) [49] have legs that are sufficiently massive for the changes of momentum that occur as they swing to affect the velocity of the trunk (Figure 2.1b). MCGEER (1992) [50] developed a walker model with knee joints (Figure 2.1c). The stride period for this model a little differs from those models without knees. The energy costs are bigger, but the gait is more realistic.

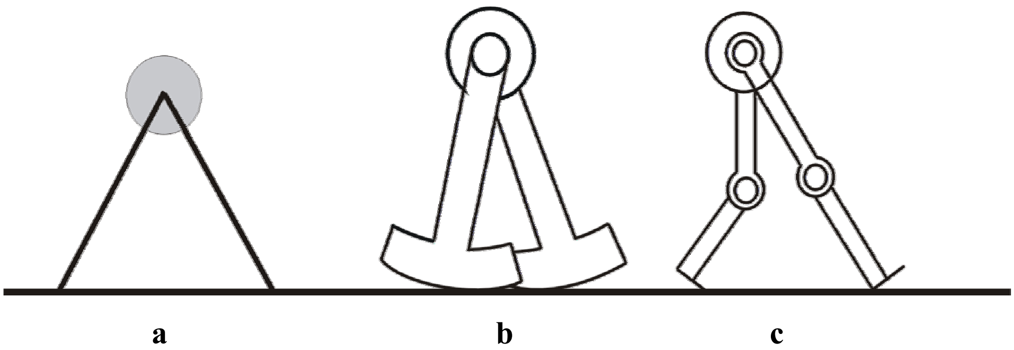


Figure 2.1: Models of walker: a) Minimal biped (Alexander, 1976), b) Two-dimensional biped (McGeer, 1990a), c) Knee-jointed biped (McGeer, 1992).

ALEXANDER (1995) [5] measured the force that the foot exerts on the ground and compared with the force obtained using his model. The gait in the model was adjusted in order to achieve same pattern force as measured for people. The vertical component of force must equal body weight, when averaged over a complete stride. The force is greater than body weight at the stage of a walking stride when both feet are on the ground and the centre of mass has an upward acceleration. The force is smaller than body weight as the centre of mass passes over the supporting foot and accelerates downwards. According to measured pattern of vertical force during human walk an equation of vertical force dependent on stride period (T), duty factor of each leg (β), body mass (m) and shape factor (q) is developed. The equation has the following form: $F_y = \frac{3\pi mg}{4\beta(3+q)}\left[\cos\left(\frac{\pi t}{\beta T}\right) - q\cos\left(\frac{3\pi t}{\beta T}\right)\right]$. The force is presented by means of truncated Fourier series. The only parameter with physical meaning is body mass (m). A good matching between measured and modelled force pattern is achieved adjusting parameters β and q in the model.

MCGEER (1990) [49] analysed the passive dynamic walking with use of simple, pure mechanical models. It is suggested that the mechanical parameters of the human body (e.g., leg length, mass distribution) have a greater effect on the existence and quality of gait than predicted. GARCIA et al. (1998) [23] give the accent to the mechanics rather than control of human walk. A simple bipedal model with two rigid legs connected to point mass through a frictionless hinge is analysed. The model of motion is governed by the laws of classical rigid body mechanics. The dynamics of the uncontrolled systems are based on mass distribution and length characteristics. According to above presented the simple walking model of human movement can nicely represent the mechanics and forces of the real system.

The cattle walk, as a four legged animal, is more complicated than two legged being motion modelling. From the previous results it is concluded that the passive dynamic models might be a good starting point for understanding some aspects of animal motion. It may be that many animal motions are largely natural or quasi-passive and not heavily controlled. In this dissertation a model of cow movement is presented. It is planed to be simple to avoid difficulties but at the same time to give some positive advantages due its simplicity. HUBBARD (1993) [34] wrote, “The most fundamental understanding often comes from the simplest models”. The simpler the model, the easier is to find the essential influences on the observed effect.

2.3 System modelling

There are two types of modelling: analytical and experimental (LJUNG et al., 1994 [43]). Analytical modelling involves physical modelling of the system, which results in derivation of mathematical equations describing various processes within the plant based on first principles of physics. Physical parameter models are often called white-box models. In order to develop such a model, the understanding of processes and physical knowledge is required. Most of physical modelling includes certain assumptions and simplifications since most systems in real world are non-linear and the knowledge about many physical phenomena are poor. In many cases non-linear systems are linearised. Such a model mostly denotes an approximate representation of the real system that gives fairly accurate estimation of real system behaviour. However, the simplifications and approximations of real systems can be valid in some system cases, but sometimes the construction of complex non-linear model cannot be escaped. Modelling of complex systems and systems with uncertainties with conventional mathematical tools is in many cases not possible. Another tool for describing such systems is needed. When the analytic approach cannot solve the modelling of some complex and/or uncertain system the experimental concept is employed (LJUNG, 1987 [42], SÖDERSTRÖM et al., 1989 [71]).

2.3.1 Physical and mathematical modelling

Physical modelling of a system is the first step in describing physical behaviour of the system. To do so some elementary understandings of the system need to be known. The physical model of a system is described as a spring-mass-damper system. It may involve motions, translation and rotation, of the mass. Dependent on number of springs, masses and dampers, and their connections the physical meaning of the model change. Once the physical model is described mathematical model is derived. Mathematical model of the system is a set of differential equations, which describe motion of the physical model. Behaviour of the model and, therefore, the real system can be predicted solving the mathematical equations for particular cases. The model represents correctly the real system, if the results from the model match the output of the real system; otherwise the model has to be changed.

2.3.2 Experimental system modelling and function approximation

Experimentally based models are developed from experimentally measured input-output data. Experimental analysis can produce two models: parametric and non-parametric. Parametric models can be designed as differential equations, frequency transfer functions or state space representation. The main feature of parametric models is the known model structure. Non-parametric models include: look-up tables, impulse response functions, convolution sums and frequency response functions. Non-parametric models do not possess a given structure and use a large set of parameters. Experimental models describe the process in an input-output sense and the parameters generally do not have any physical meaning. Simplified, experimental modelling is basically, finding a curve that fits the measured input-output data set. In other words, it can be described as function approximation or function fitting.

Function approximation is performed whenever a function needs to be constructed to connect experimentally measured input-output data. Function *f* fits input-output data set (*x*, *y*) when *y*=*f*(*x*). Function approximation is the same as system identification, since the objective of the theories is to find the optimal model that estimates the output of the real system for certain input.

The procedure of system identification is as follows. An identification experiment is performed by exciting the system and observing its input and output over the interval of time. These signals are normally recorded in a personal computer for subsequent information processing. The model representation is assumed. Typically a linear difference equation of a certain order is selected. The chosen parametric model is then fitted to the recorded input and output sequences. Some statically based method is used to estimate the unknown parameters of the model. In practice, the estimation of structure and parameters are often done iteratively. This means that the model structure is chosen and the corresponding parameters are estimated. The obtained model is then tested to see whether it is an appropriate representation of the system. If not, some more complex model structure is considered, its parameter estimated and the new model validated. This procedure is repeated till the goal is reached.

The black-box models or ready-made models are often used when no physical knowledge about a system is known. The parameters in such models have no direct physical interpretation, but are used as tools to describe the properties of the input-output relationship of the system. Normally, ready-made models are described in discrete time, since data are collected in sampled form. It is always possible to transform the time discrete model into a time continuous model, if required. The parameter black-box models are polynomial regression models. These models are analysed and explained in many books and papers (LJUNG, 1987, SÖDERSTRÖM et al., 1989 [71], LJUNG at al., 1994 [43]). The general model structure for linear time-invariant system is:

$$A(q)y(t) = \frac{B(q)}{F(q)}u(t) + \frac{C(q)}{D(q)}e(q),$$

with

$$A(q) = 1 + a_1 q^{-1} + \ldots + a_{na} q^{-na},$$

$$B(q) = b_1 q^{-nk} + b_2 q^{-nk-1} + \ldots + b_{nb} q^{-nk-nb-1},$$

$C(q) = 1 + c_1 q^{-1} + ... + c_{nc} q^{-nc}$,

$D(q) = 1 + d_1 q^{-1} + ... + d_{nd} q^{-nd}$,

$F(q) = 1 + f_1 q^{-1} + ... + f_{nf} q^{-nf}$,

where *na, nb, nc, nd, nf* and *nk* are the structural parameters, and a_i, b_i, c_i, d_i and f_i are the adjustable parameters to input-output data. Parameter *q* is the shift operator.

From the general model four developed models are often used:

Box-Jenkins, $y(t) = \dfrac{B(q)}{F(q)} u(t) + \dfrac{C(q)}{D(q)} e(q)$,

Output error, $y(t) = \dfrac{B(q)}{F(q)} u(t) + e(q)$,

ARMAX, $A(q)y(t) = B(q)u(t) + C(q)e(q)$,

ARX, $A(q)y(t) = B(q)u(t) + e(q)$

where AR- auto regressive (*A(q)y(t)*), MA- moving average (*C(q)e(q)*) and X- extra input (*B(q)u(q)*).

The above presented models are good only for linear dynamic systems. Yet, as suggested the majority of systems are non-linear and so the modelling needs to be non-linear. A number of papers are published dealing with input-output parametric models for non-linear systems (PARKER et al., 1981 [55], LEONARITIS et al., 1985 [40], [41], HABER et al., 1990 [26], BILLINGS et al., 1988, 1989 [10,11], AGUIRRE et al., 1995 [2], SJÖBERG et al., 1995 [70], JUDITSKY et al., 1995 [37], MAO et al., 1997 [48]). The non-linear polynomial regression models consist of linear parts (as in linear polynomial regression models) and non-linear parts that are not included in linear models. Among the frequently used models are those described by LEONARITIS et al. (1985) [40], [41]. The non-linear autoregressive moving average with exogenous inputs (NARMAX) is the non-linear extension of ARMAX model. The NARMAX model is represented as the function of previous inputs, outputs and disturbance:

$$y(t) = F^l \left[y(t-1),...,y(t-n_y),u(t-d),...,u(t-d-n_u+1),e(t),...,e(t-n_e) \right]$$

where F^l is some non-linear function of $u(t)$, $y(t)$ and $e(t)$ with non-linearity degree l, $u(t)$ and $y(t)$ are, respectively, input and output time series obtained by sampling the continuous input-output data. Uncertainties, noise, unmodelled dynamics, etc, figurate in $e(t)$. The deterministic part of NARMAX model is a NARX model. NARX model can be expanded as the summation of terms with degrees of non-linearity up to the defined degree l. Even such a simple model structure as NARX has a lot of unknown parameters and coefficients. Different concepts of solving the minimal model structure, as well as, coefficient estimation from measured input-output data are suggested (BILLINGS et al., 1988, 1989 [10,11], HABER et al., 1990 [26], AGUIRRE et al., 1995 [2], MAO et al., 1997 [48]), but exhaustive computation of unknown parameters cannot be avoided. The mathematical function structure describing a real world process might be very complex and its exact form is usually unknown. Therefore, another tools for function fitting and so system identification described by neural networks and fuzzy logics need to be introduced.

2.3.3 Fuzzy theory and applications

The practical success of fuzzy control systems in commercial products and industrial process control caused the increase of investigations in theoretical fuzzy systems and control. Researchers are trying to understand its possibilities and develop more powerful tools for modelling, function approximation and control. In this section the achievements in field fuzzy logic are presented. The first part is a short review of fuzzy theory, while the second explains its possibilities in modelling and function approximation.

The word “fuzzy” stands for blurred, imprecisely defined, indistinct, confused. However, fuzzy theory is precisely defined. The meaning is mostly related to the phenomena that fuzzy systems theory characterise (exp.: low speed is between 0 and 60 km/h, high speed is between 50 and 120 km/h). Fuzzy systems theory is justified by complexity of the real world to be precisely described and the need of formulating human knowledge in systematic manner and put it in engineering systems. Fuzzy systems are knowledge or rule based systems. The heart of the system is the IF-THEN rule. Fuzzy systems have been applied to a variety of fields ranging from control, signal processing, communication, integrated circuits manufacturing, and expert systems in business, medicine, etc.

There are two major types of fuzzy systems: Mamdani fuzzy systems and Takagi-Sugeno (TS) fuzzy systems (TAKAGI et al., 1985 [74], SUGENO et al., 1988 [73]). The main difference lies in the consequent of fuzzy rules. Mamdani fuzzy systems use fuzzy sets as rule consequent, while Takagi-Sugeno fuzzy systems employ linear functions of input variables as rule consequent. The example for Mamdani IF-THEN rule is: “IF speed is low THEN more force to accelerator”. The example for Takagi-Sugeno IF-THEN rule is: “IF speed is low THEN force to accelerator is $F=cx$”. The product of constant c and input x has the same meaning as words in consequent part of Mamdani example. The main problem of Takagi-Sugeno fuzzy system is that the mathematical formula cannot be used in representing human knowledge. On the other hand, a great advantage of Takagi-Sugeno fuzzy models is its representative power. It is capable of describing a non-linear system using sufficient rules and training data.

Every fuzzy system consists of: fuzzifier, fuzzy rule base, inference engine and defuzzifier. The fuzzifier is defined as a mapping from real valued point $x \in \Re$ to fuzzy set A. For example the word "high speed" is a linguistic variable for physical value between 50 and 120 km/h. Fuzzifiers differ because of membership functions used in fuzzification. Here are some fuzzifiers commonly used. Singleton fuzzifier maps a real valued point x into membership value 1 at $\bar{x}$ and value 0 at all other points. Gaussian fuzzifier maps x into a fuzzy set, which has the following membership function:

$$\mu(x) = e^{-\left(\frac{x_1-\bar{x}_1}{\sigma_1}\right)^2} * \ldots * e^{-\left(\frac{x_n-\bar{x}_n}{\sigma_n}\right)^2},$$

where $\bar{x}_i$ and σ_i are positive parameters and the t-norm * is usually chosen as algebraic product or *min*. Triangular fuzzifier maps x into fuzzy set, which has the following triangular membership function:

$$\mu(x) = \left(1 - \frac{|x_1 - \bar{x}_1|}{b_1} * \ldots * 1 - \frac{|x_n - \bar{x}_n|}{b_n}\right) \text{ if } |x_i - \bar{x}_i| \le b_i,\ i = 1,2,\ldots n$$

$$\mu(x) = 0, \text{ otherwise}$$

where b_i are positive parameters and the t-norm * is usually chosen as algebraic product or *min*. The singleton fuzzifier greatly simplifies the computation involved in the fuzzy inference engine. The Gaussian and triangular fuzzifiers can suppress noise in the input, but the singleton fuzzifier cannot.

Fuzzy rule base consists of a set of fuzzy IF-THEN rules. Fuzzy rule base has the following form: *IF (x_1 is A_1) and ... and (x_n is A_n), THEN (y is B)*. The elementary logic operations such as conjunction, disjunction and negation can be used in fuzzy rule base. In a fuzzy inference engine fuzzy logic principles are used to combine the fuzzy rule base into a mapping from fuzzy set A to a set B. There are two ways to infer with a set of rules. In composition based inference all rules in fuzzy rule base are combined into single fuzzy relation. In individual rule base inference each rule in the fuzzy rule base determines an output fuzzy set, and the output of the whole fuzzy inference is the combination of the individual fuzzy sets. Most commonly used fuzzy inference engines are: product and minimum inference engine. In product inference engine algebraic product is used for t-norms and maximum for s-norms operators. In minimum inference engine minimum is used for t-norms and maximum for s-norms operators.

Deffuzifier is defined as the mapping from fuzzy set, which is the output of the fuzzy inference engine to crisp point value. Commonly used defuzzifiers are: centre average and maximum defuzzifier. Centre average defuzzification is computed according the equation:

$$y = \frac{\sum \bar{y}_i \omega_i}{\sum \omega_i},$$

where $\bar{y}_i$ is the centre of the i-th fuzzy set and ω_i is the height of i-th fuzzy set. Maximum deffuzifier chooses the crisp output at which the fuzzy set has the maximum value. Since the maximum might be achieved at more than one point, different criterions can be used. Among them are: the smallest of maxima, the largest of maxima, mean of maxima, centroid of area, bisector of area, etc.

Fuzzy system composed of the mentioned elements is presented in Figure 2.2.

The process of fuzzy reasoning is presented in the following steps:

1) Compare the input variables with the membership functions on the premise part to obtain membership values of each linguistic label (fuzzification).
2) Combine through a specific t-norm operator, usually multiplication or minimum, the membership values on the premise part to get firing strength (weight) of each rule.
3) Generate the qualified consequent (fuzzy or crisp) of each rule depending on the firing strength.
4) Gather the qualified consequent to produce a crisp output (defuzzification).

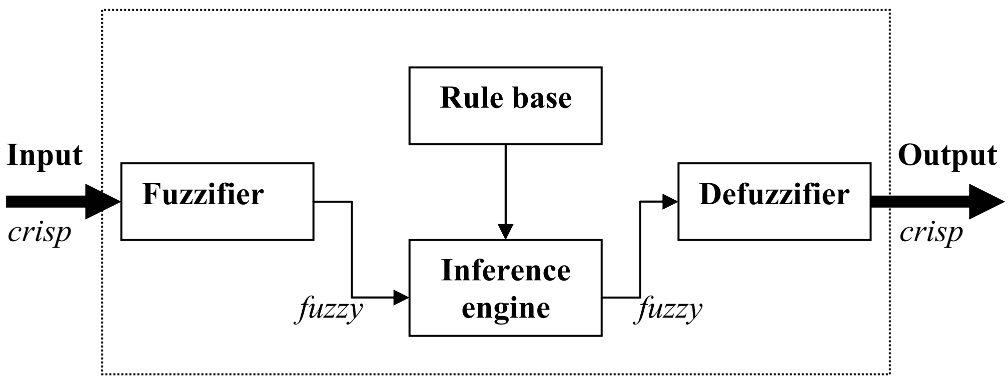


Figure 2.2: Basic configuration of fuzzy system.

Depending on the types of fuzzy reasoning and fuzzy IF-THEN rules employed, most of the fuzzy systems can be classified into three types:

a) The overall fuzzy output is the weighed average of each rule crisp output. The crisp outputs are generated by the rules firing strength, which is the product or minimum of membership values from the premise part.

b) Applying maximum defuzzification to the qualified fuzzy outputs derives the overall fuzzy output. Minimum inference operation is performed on firing strength of each rule.

c) Takagi and Sugeno's fuzzy IF-THEN rules are used. The output of each rule is the linear combination of input variables plus a constant term. The final output is the weighted average of each rule output.

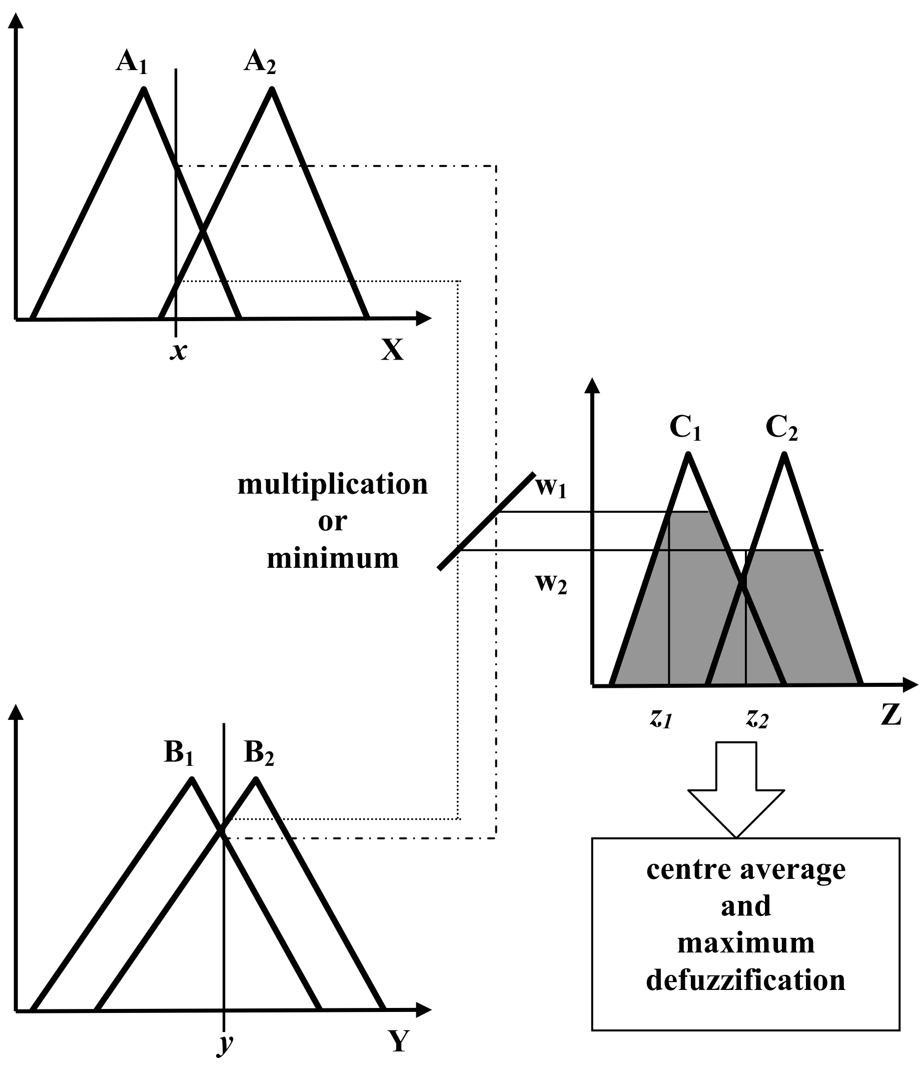


Figure 2.3: Fuzzification, inference operation and defuzzification.

The most commonly used membership functions in fuzzification are triangular and Gaussian. Gaussian function has an advantage since it never reaches zero. Algebraic multiplication or minimum is applied in inference engine. Defuzzification is obtained with centre average or maximum method.

$$z_1 = ax + by + c$$
$$z_2 = px + qy + r$$
$$z = \frac{w_1 z_1 + w_2 z_2}{w_1 + w_2}$$

Figure 2.4: Takagi-Sugeno fuzzy rule consequents and centre average defuzzification.

In Figure 2.3, a two rule and two input single output fuzzy system is presented. The Takagi-Sugeno fuzzy rules and centre average defuzzification for this system are shown in Figure 2.4. Since the most common fuzzy systems are being explained, it is to present the possibilities when these systems are applied. In this work the accent is on the Takagi-Sugeno fuzzy system. In the next section the application of Takagi-Sugeno systems in function approximation and system identification.

In fuzzy logic theory many effort is done exploring the Mamdani fuzzy systems in system modelling and function approximation (WANG et al., 1992 [80], ZENG et al., 1994 [88], YING, 1994 [86], CASTRO, 1995 [13], ABE et al., 1995 [1], DICKERSON et al., 1996 [18], CASTRO et al., 1996 [14], RUNKLER et al., 1999 [67], THAWONMAS, 1999 [75], WANG et al., 2000 [81], ROJAS et al., 2000 [63], SALMERI et al., 2001 [69], LANDAJO et al., 2001 [39], POMERAS et al., 2002 [59], GONZALEZ et al., 2002 [25]). Mamdani fuzzy systems employing fuzzy singleton as rule consequent are equivalent to linear Takagi-Sugeno fuzzy system rule consequent with coefficients set to zero. In other words, the simplest Takagi-Sugeno rule consequent is same as Mamdani singleton rule consequent. Almost all the Mamdani fuzzy systems used in function approximation discussed in above references use singletons as rule consequent. This, however, points out the importance of investigations of Takagi-Sugeno fuzzy systems as universal approximators. Recent investigations in Takagi-Sugeno fuzzy systems show the good capability in function approximation and modelling (YING, 1998 [87], ZENG et al., 2000 89]). The Takagi-Sugeno type of fuzzy models has attracted great attention due to their good performance in various applications. Takagi-Sugeno fuzzy models have a great advantage compared to traditional Mamdani fuzzy models since their functional type fuzzy rules consequences enable the fuzzy system to represent any non-linear system using sufficient rules and training data. In both, Mamdani and Takagi-Sugeno, fuzzy systems when modelling a real system or approximating a function the adjusting parameters are: fuzzification parameters, fuzzy rules, defuzzificatin parameters. Fuzzification parameters are the number of membership functions and its characteristics. The number of membership functions covering the input region need to be optimal, such that unnecessary computation or impreciseness is minimized. The characteristics of the membership function are the width, height and centre of the membership function. Same as for fuzzification, defuzzification membership functions has to be defined to produce desired output. Specially, for Takagi-Sugeno fuzzy system the coefficients of the fuzzy rules consequences are adjustable. The connection between fuzzification and defuzzification is organized with fuzzy IF-THEN rules. Number of rules as well as the link between input and output fuzzy values has to be determined for the particular system.

The design of fuzzy system using a table look-up scheme (input-output pairs) as a simple heuristic method is described by WANG (1997) [82]. First the fuzzy sets are defined to cover the input-output spaces. All membership functions in fuzzification of crisp input have the same width and height, and the centre is moved for the same interval. The identical principle is applied in output space defuzzification. Function fitting accuracy increases as the number of membership functions in input-output space enlarges. One fuzzy rule is generated from each input-output pair. Conflicting rules are expelled and the rule with highest degree is kept. So created fuzzy system may not be complete, since the number of rules in the fuzzy rule base may be much less than all possible combinations of the fuzzy sets defined for input variables and number of input-output pairs. For this reason the method of look-up scheme is not commonly used. The concept of designing a fuzzy system by first specifying its structure and then adjusting its parameters using some training algorithm is widely investigated and used. With specifying the structure the elements of the fuzzy system are denoted. In order to do so, following obscurities have to be enlightened. First, the dimensions of input and output spaces are established. It depends on the ability to experimentally measure inputs and outputs of the real system. Yet, the inputs and outputs that have minor influence on the system behaviour can be neglected and so simplify the fuzzy model. Further on, the number of fuzzy rules is determined. The number of rules in a fuzzy model corresponds to the order in a conventional model. Higher order model minimizes the conventional criterion, i.e. output error, but might result in over fitting. In many cases the input-output data is contaminated by noise so the lower order model gives better fitting on the average. The fuzzy model parameter identification is done by denoting the parameters in the membership functions in fuzzy sets and the coefficients in Takagi-Sugeno fuzzy rules consequent.

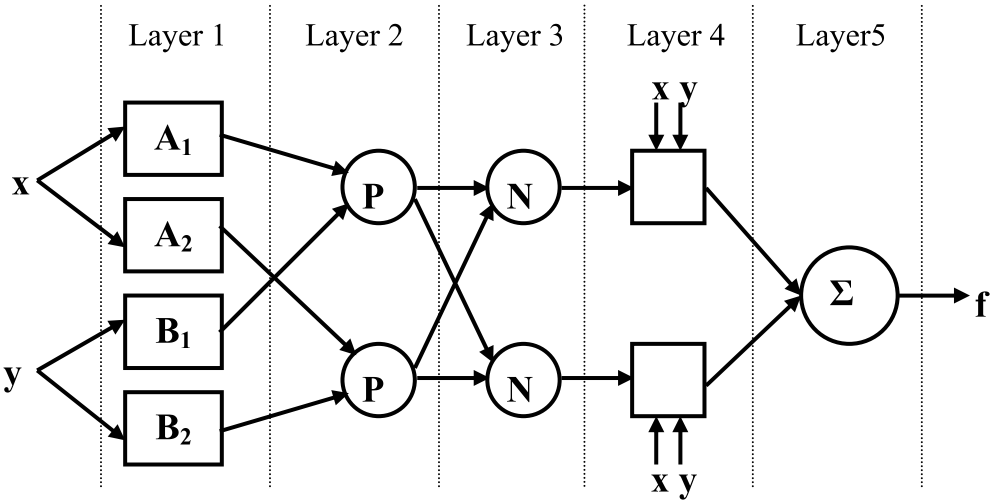


Figure 2.5: ANFIS structure for two input single output with two fuzzy rules.

One of the most influencing works in the input-output data mapping is done by JANG (1993) [36]. Based on Takagi-Sugeno fuzzy system an adaptive-network-based fuzzy inference system (ANFIS) was created. ANFIS is the presentation of fuzzy system through feedforward neural network with supervised learning capability. The adaptive network representing Takagi-Sugeno fuzzy system consists of five layers as shown in Figure 2.5.

The square and circle nodes are for adaptive nodes with parameters and fixed nodes without parameters, respectively. The first layer consists of square nodes that perform fuzzification with chosen membership function, which is usual a bell-shaped function, i.e. Gaussian membership function. The parameters in this layer are called premise parameters. In the second layer the t-norm operation (algebraic product) is performed to produce firing strength of each rule. The ratio of i-th rule firing strength to the sum of all rules' firing strength is calculated in the third layer generating the normalized firing strengths. The fourth layer consists of square nodes that perform multiplication of each normalized firing strength with corresponding rule. The parameters in this layer are called consequent parameters. The overall output is calculated by summation of all incoming signals in the fifth layer. The adjustable parameters are the premise and consequent parameters. JANG (1993) [36] proposed the hybrid learning method that integrates gradient descent and least square estimation. Each epoch of hybrid learning procedure is composed of forward and backward pass. In the forward pass, for supplied input data the consequent parameters are calculated with sequential least squares formulas. After identifying the consequent parameters, the output error is obtained comparing the created output and measured output. In backward pass, the error rates propagate from the output towards input updating the parameters in premise part by the gradient descent method. As proposed, the structure of the fuzzy system is fixed and the parameter identification is solved through the hybrid training. The possibilities of this method are broad and can be used in modelling non-linear functions, predict chaotic time series, signal processing, etc. ANFIS can replace almost all neural networks in control systems.

The presented ANFIS model can be simplified by constraining some adjustable parameters. It depends on the knowledge about the system that is being modelled. On the other hand, ANFIS model can be easily completed by adding the part for structure identification, which concerns with the selection of an appropriate input space partition style and the number of membership function for each input, etc. It is proved that ANFIS can achieve high non-linear mapping and reproduction of non-linear time series. For many cases of non-linear mapping ANFIS uses less parameters and time adaptation then neural network. ANFIS architecture is functionally equivalent to radial basis function neural network (RBNN). Training methods used in RBNN can be applied in ANFIS and vice versa.

2.3.4 Neural networks structure and application

Neural networks theory is developed as a result of necessity to create a system that can be capable of behaving as human being brain. The brain is a highly complex, non-linear and parallel computer, which is capable to organize its structural constituents and perform certain computations much faster than the fastest digital computer. Neural network carries out a function by means of training instead of algorithmic programming.

Because of its properties and capabilities of non-linearity, generalization, adaptivity, input-output mapping, etc. neural networks are widely used in engineering, mathematics, physics, biology, medicine, etc. Neural networks are based on the human nervous system structure.

The elementary part of the neural network is a neuron. Neuron is an information-processing unit that is fundamental to the operation of a neural network. It consists of: source nodes, synapses, adder, bias and activation function. A basic structure of a neural network is presented in Figure 2.6. Input signal is entering the system through the source nodes. Synapses are connecting links between input signals and adder, characterised by a

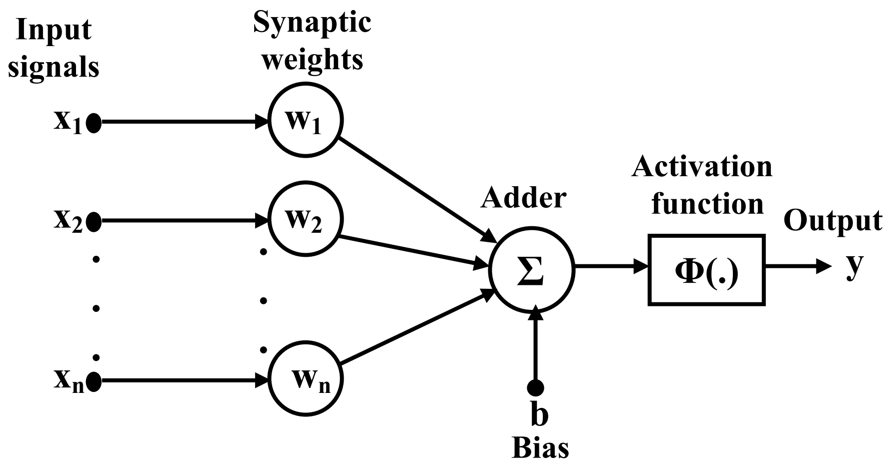


Figure 2.6: Neuron structure.

weight. Adder performs summation of incoming signals from neurons. Bias has the effect of increasing or lowering the net input of the activation function. Activation function limits the amplitude of the output. There are various activation functions used for creation of neuron output. Three basic groups activation functions are as follows: threshold, piecewise-linear and sigmoid functions (HAYKIN, 1999 [30]). Threshold activation function is often called hard limiter, since the output can be zero or one, dependent on input. Threshold function is of form:

$$\Phi(x) = \begin{cases} 1, & x \geq a \\ 0, & x < a \end{cases}$$

where $\Phi(x)$ is activation function, and a is a constant number.

Piecewise-linear activation function may be viewed as an approximation to a non-linear amplifier and is presented as:

$$\Phi(x) = \begin{cases} 1, & x \geq a \\ x, & a > x > -a \\ 0, & x \leq -a \end{cases}$$

Sigmoid activation function is by far the most common used activation function in neural networks. These functions are s-shaped and the widely implemented are: $\Phi(x) = \tanh(x) = \dfrac{e^x - e^{-x}}{e^x + e^{-x}}$, the hyperbolic tangent function and

$\Phi(x) = sig(x) = \dfrac{1}{1 + e^{-ax}}$, the sigmoidal function.

Beside mentioned activation functions, Gaussian bell is often applied as output activation in radial basis neural networks.

In neural networks neurons as processing elements are organized in parallel arrangement called layers. The network topology is mainly determined by the layer order and the connection direction. First layer in the system is called input layer, while the last is called output layer. Between the input and output layer are the hidden layers. Single layer network is the network where input layer is at the same time the output layer. Multi-layer networks consist of more than one layer. The connection direction controls the information flow from the networks input to the output. Possible connections are: feed-forward, feedback, recurrent and lateral. In Figure 2.7, multi-layer network with various connection directions is shown.

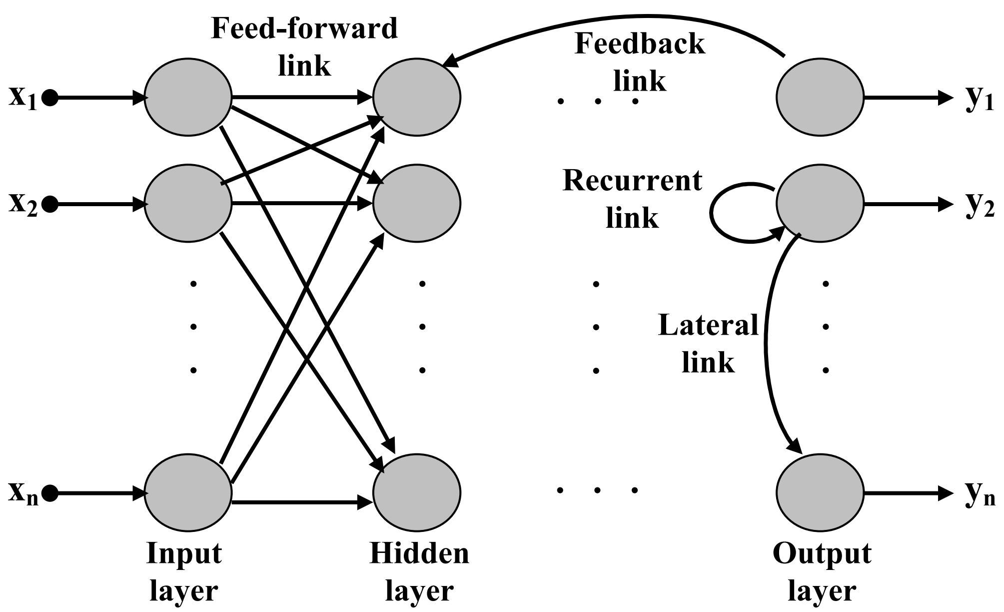


Figure 2.7: General neural network architecture.

Artificial neural networks can be grouped into four categories: stochastic, associative, hierarchical and mapping neural nets. Stochastic networks typically have a cost function and employ noise processes to reach global minimum of this function. This network utilizes the procedure called simulated annealing (METROPOLIS et al., 1953 [51]) in Boltzmann machine network (HILTON et al., 1984 [32]). Stochastic networks are being used for solving combinatorial optimisation problems.

Associative networks are used for storing and recalling patterns. They consist of only one functional layer with both feed-forward (ANDERSON et al., 1972 [6], KOHONEN, 1972 [37]) and recurrent (ANDERSON, et al., 1977 [6], HOPFIELD, 1982 [33]) connections. Associative networks are widely applied to pattern recognition tasks. In hierarchical neural networks the processing units within a layer are connected to a group of processing elements in the previous layer. This type of network is not so commonly used. It is investigated in pattern recognition of static images (FUKUSHIMA, 1988 [21]). The fourth category of neural networks is mapping networks. Mapping networks are capable of approximating any mathematical function in an input-output sense. Multi-layer perceptron and radial basis function networks are commonly applied in classification problems and identification tasks. Self-organized maps (KOHONEN, 1987 [38]) and counter-propagation networks belong to the mapping networks group.

The significant property of neural networks is its ability to learn. The improvement in performance of the network occurs by adjustments of synaptic weights and biases over a period of time in which the network and environment interacts. There are two types of learning: learning with a teacher and learning without a teacher. Learning with a teacher is also referred as supervised learning. With this sort of system learning, knowledge of the environment is being presented by a set of input-output data. Network is producing output for certain input, and is compared to desired output. The adjustment of the network is performed with parameter estimation procedures in order to minimize a performance function. Adaptation algorithms deal with least mean square error between produced and desired output of the system. Among these algorithms are the delta rule (Widrow-Hoff law) and error backpropagation algorithm. Learning without a teacher can be divided in the two subdivisions: reinforcement and unsupervised learning. Reinforcement learning is similar to supervised learning except that instead of known value of desired output for particular input the network receives a score on how it behaved over a sequence of training. This is achieved with cost function evaluation. The advantage of reinforcement learning is in possibility of network adjustment without correct input-output trial. In self-

organized (unsupervised) learning there is no external teacher or critic to watch over the learning process. The most famous learning algorithm in unsupervised learning is the competitive learning rule. Such a network contains a competitive layer in which neurons compete with each other according to learning rule for the opportunity to respond to input. However, beside the mentioned learning methods there are some other learning rules that are not so common (coincidence, filter, spatiotemporal learning).
In the following section the commonly used networks in classification, function approximation and system identification are reviewed.

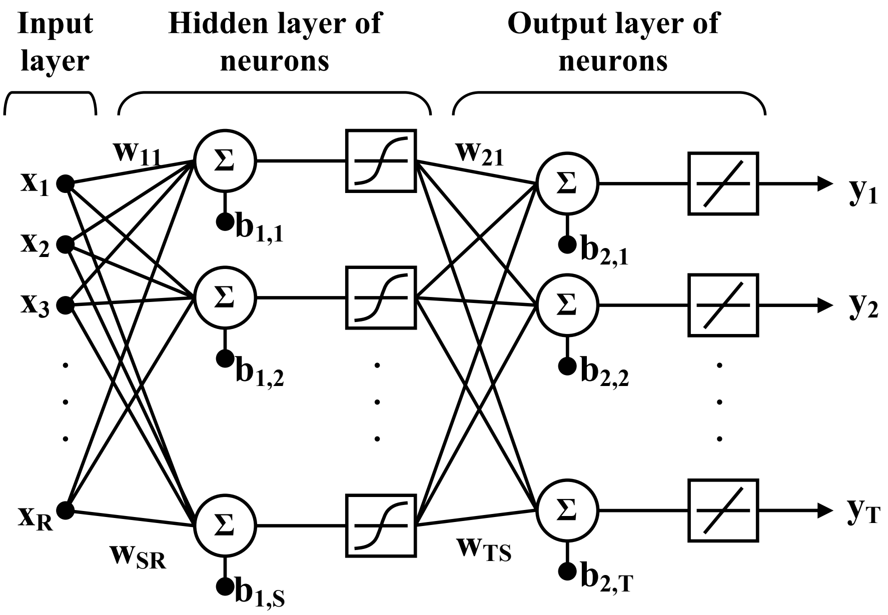


Figure 2.8: Multi-layer preceptron network structure with one hidden layer.

Multi-layer perceptron and radial basis function networks are the basic constituents of feed-forward neural network. They are structurally equivalent consisting of one hidden layer with non-linear activation function and an output layer with linear activation function. Multi-layer perceptron network is a powerful tool for a variety of problems ranging from function approximation to image processing. The elementary processing unit called neuron is here referred as perceptron. A perceptron structure is similar to that shown in Figure 2.6, with activation function chosen as non-linear function (i.e. sigmoid, hyperbolic tangent function). ROSENBLATT (1962) [64] invented the perceptron structure as known today. Single-layer perceptron network is commonly used with threshold activation function and so-called Widrow-Hoff learning algorithm, which is the least square error technique for weight estimation by lowering the cost function

(WIDROW et al., 1960, 1985 [84], [85]). The parallel arrangement of perceptrons within layers and feed-forward connection between layers characterize multi-layer perceptron (Figure 2.8). The hidden layer includes sigmoid transfer (activation) function (exp. tangent-sigmoid, logarithm-sigmoid), while linear activation function is specified in the output layer.

Multi-layer perceptron networks are, generally, trained with error backpropagation learning rule. Error backpropagation rule was created by generalizing Widrow-Hoff (least mean square) algorithm. Standard error backpropagation algorithm is, as well as Widrrow-Hoff algorithm, gradient descent training method performed for known input-output data set. The equation of Widrow-Hoff algorithm is:

$$w_i(n+1) = w_i(n) + \alpha\delta(n)x_i(n),$$

where w_i is the ith weight, α is the learning rate, and δ is the current gradient defined as:

$$\underline{\delta}(n) = \Phi' e(n),$$

with activation function Φ and error at nth iteration

$$\underline{e}(n) = d(n) - y(n),$$

$$y(n) = \Phi\left(\sum_{i=1}^{m} x_i(n) w_i(n)\right),$$

for ith input x_i, computed output y, and desired output d. Total number of inputs into the network is m. Φ' denotes the derivative of the activation function with respect to activation (input). Error backpropagation algorithm was essentially derived to provide an approximation of the gradient term for hidden layers (RUMELHART et al., 1986 [65]). Error backpropagation rule adjust the weights according to the mathematical formula:

$$w_{ji}(n+1) = w_{ji}(n) + \alpha\delta_j(n)x_i(n),$$

where x_i is the ith input of the jth neuron, w_{ji} is the ith weight of the jth neuron, α is the learning rate (correction factor) and δ_j is the local gradient,

$$\delta_j(n) = e_j(n)\Phi'_j,$$

for Φ'_j being the derivative of the jth activation function, and error

$$e_j(n) = d_j(n) - y_j(n),$$

output of the jth neuron

$$y_j(n) = \Phi_j\left(\sum w_{ji}(n)x_i(n)\right).$$

There are two different ways of gradient descent algorithm implementation to the network: incremental mode and batch mode. In the incremental mode the gradient is computed and the weights are updated after each input-output pair is applied to the network. In the batch mode, weights are updated after all input-output data sets are employed to the network. In both cases, synaptic weights are adjusted backwards starting with the last layer and ending in the first layer. A primary difficulty of error backpropagation method is the proper choice of correction factor. Large learning rates lead to faster adaptation of the network, but too large rates lead to instability. On the other hand, small learning rates cause long training time. Many work have been done on improving error backpropagation rule. Some are dealing with better choice of initial weights (NGUYEN et al., 1990 [54]). Faster algorithms can generally be classified into two categories. The first category uses heuristic techniques such as momentum technique, which decreases the possibility of trapping in the local minima. The second category uses numerical optimisation methods. Among them are the conjugate gradient, quasi-Newton and Levenberg-Marquardt training methods, which utilize second order derivative information (BATTITI, 1992 [8], BEALE, 1972 [9], CHARALAMBOUS, 1992 [15], FLETCHER et al., 1964 [20], HAGAN et al., 1994, 1996 [27], [28], MOLLER, 1993 [52], POWELL, 1977 [60], VOGL, 1988 [78]).

Multi-layer perceptron networks with three layers have been proved to be universal approximators (MOORE et al., 1988 [53], FUNAHASHI, 1989 [22]) satisfying the universal approximation theorem. It was confirmed that even a two layer feed-forward network could approximate arbitrarily well any continuous function (CYBENKO, 1989 [17]). Hecht-Nielsen in 1990 developed a backpropagation theorem to prove approximation ability of backpropagation networks (feed-forward network with error backpropagation learning).

Radial basis function networks gained a considerable attention as an alternative to multi-layer perceptron networks trained by error backpropagation algorithm. They are structurally equivalent. Radial basis function neuron uses distance (Euclidian norm) between input vector and weight vector

$$\|x - w\| = \sqrt{\sum (x_i - w_i)^2} \ ,$$

and a radial basis function as the activation function of form

$$\Phi(u) = e^{-u^2} \ .$$

The Euclidian distance with radial basis activation function is similar to the Gaussian bell-shaped function. A multi-layer radial basis function network structure with one hidden layer is shown in Figure 2.9. The hidden layer contains radial basis function neurons. In the output layer, linear activation function is applied for function approximation and s-shaped activation function is applied for classification problems.

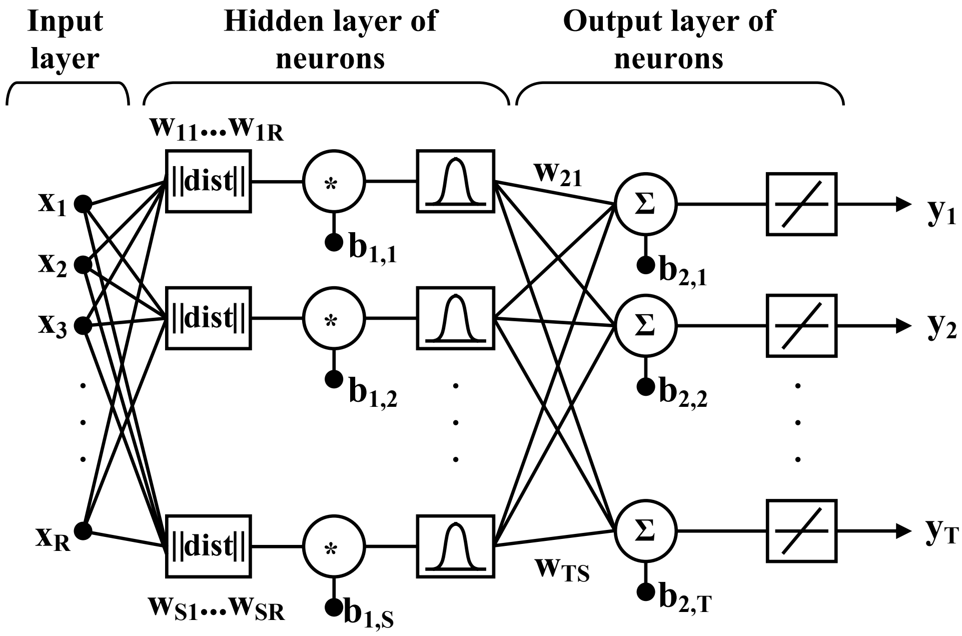


Figure 2.9: Radial basis function network structure with one hidden layer.

Like multi-layer perceptron networks, radial basis networks have universal approximation ability (HARTMAN et al., 1990 [29], PARK et al., 1991 [56]). Radial basis function networks have been proved to have best approximation property among the artificial neural networks (GIROSI et al., 1990 [24]). Radial basis networks may require more neurons than multi-layer perceptron networks, but often can be trained much faster. There are two variations of the standard radial basis network: regression and probabilistic neural networks (WASSERMAN, 1993 [83]). Regression networks are often used for function approximation, while probabilistic networks can be used for classification problems. Regression network is similar to radial basis network, but has a slight difference in the linear layer. The normalized dot product operation is performed on coupling the input into the layer and its weights. In probabilistic network output layer is so-called competitive layer that sums the contributions for each class of input to produce output and form a vector of probabilities. Compete activation function picks the maximum of the probabilities and assign 1, but for other classes 0. Radial bases function networks are not suggested to be used for the systems with high dimensional input space, because of exponential increase of parameters. Radial bases function approximate locally, while multi-layer preceptron does it globally. Multi layer perceptron can approximate only smooth non-linearities. Radial bases function is able to approximate both smooth and nonsmooth non-linearities.

2.3.5 Summary

To model a real system both analytical and experimental methods can be used. The mathematical equations of the model can be derived from physical model with some assumptions and simplifications. However, constrains of the model might cause inaccuracy of the model output. Consequently the experimental modelling methods can be applied. The most powerful black-box models, which connect input and output values of the real system, are based on fuzzy logic and neural networks theory. According to earlier exposed system modelling theory there are no strict rules to create fuzzy or neural network model. Each system has to be modelled independently and all cases have to be examined. Therefore, there is no final answer that suggests what type of modelling method to use in a particular case.

3 Goals of the work

The aim of this work is to adapt an existing weighing scale for the purpose of walk-through weighing of cattle and to create a method of denoting cattle body weight from the measured force. Based on the previous research results, it is obvious that the process of automatic weighing involves measuring of weighing force produced by feet of cow during its walk over the weighing platform. The measured force is processed and the body weight of the animal is estimated. The accent is on achieving precise and secure body weight recognition. The major inaccuracies of existing dynamic weighers are caused by crowding of animal on the weighing scale and fast crossing of the scale. The errors are produced due to the simplifications introduced to the real system. Simple averaging of the stochastic non-linear system does not necessarily lead to accurate result. Consequently, in this work the dynamic weighing of the dairy cows is analysed as a stochastic non-linear system.

This work represents an extension of the previous knowledge on automatic weighing scales and gives some new approaches and solutions, which avoid the previous lacks, like the problem of crowding on the weighing device and fast crossing of cattle.

The work consists of six main parts:

1) walk-through weighing scale design
2) signal processing
3) mathematical model design
4) fuzzy logic model design
5) neural network classifier design

An existing weighing platform is adjusted for dynamic weighing of cattle. The platform used was manufactured by DeLaval as a standard weighing device. The signal from the load cell is recorded in a PC and later processed. Recorded force signal is filtered from noise and then the valuable part of the signal is selected. The chosen signal part is processed with a body weight recognition method. Three methods are developed for body weight recognition. The first is based on mathematical model of cattle movement. The second performs function approximation of measured force signal based on fuzzy logic theory. The third body weight recognition method involves neural network classification. The body weight estimations for dairy cows weighted dynamically and processed with the three methods are presented and discussed.

4 Materials and methods for measurement of body weight data

4.1 Walk-through weighing scale description

The experiments were performed on the experimental dairy farm of Technical University Munich in Freising, Germany. The experiments were accomplished on dairy cows. The cattle were measured each time they were milked. The weighing system was designed and built to fit into the walking path of the cows as they leave the milking parlour. The milking parlour was a herringbone with six cows on each side. The maximum of 12 cows could be released at once. The weighing scale was positioned in the return alley of the milking parlour. This was chosen since the body weight without milk is of interest. Another important factor was that the animals were measured in the same interval of the day. Consequently, the daily oscillations in the animals body weights were minimized. The plan of the position of the milking parlour and the walk-through weighing device are shown in Figure 4.1.

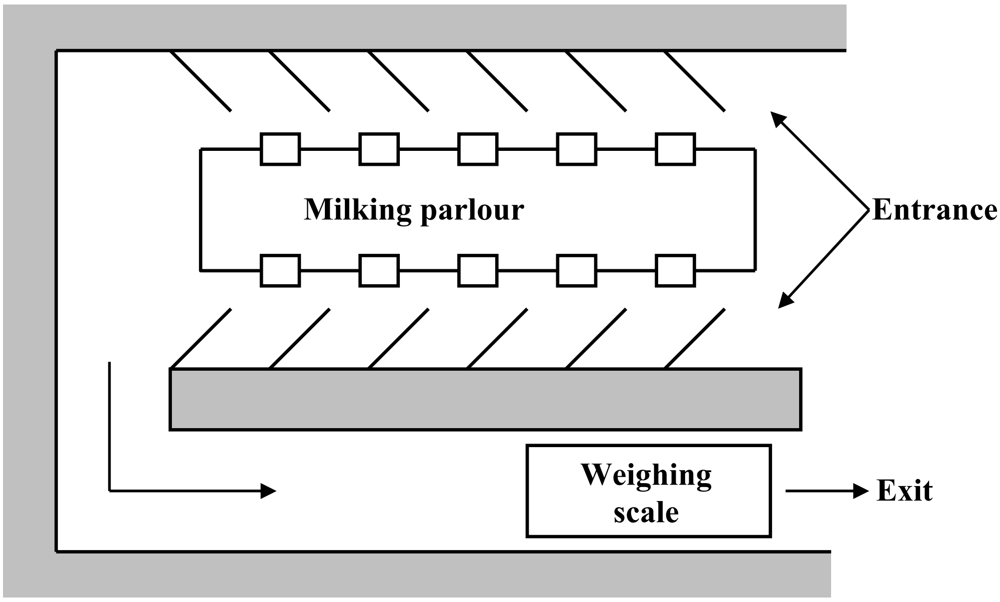


Figure 4.1: Plan of weighing scale position.

The corridor connecting milking parlour and weighing scale was 1.2 meters wide enabling cows to advance one by one in a row. The floor was smooth and sometimes slippery, which caused the animals to move with caution. However, younger animals walked faster than older and heavier. The weighing platform was made by DeLaval as a commercial weighing device. The weighing scale is made of thick steel plate and had a width of 0.8 meters and length of 2.3 meters. The width of the scale did not allow more than one cow to fit side by side on the weigher. The scale was designed to enable one cow to fit on the scale with its all four legs. The scale platform was elevated above the floor for 10 centimetres. The mount was influencing the walking speed of some animals, but for some it did not change their pace. On the height of 1.9 meters above the platform a horizontal bar was mounted on the entrance of the weigher. This obstruction was aimed to disable the cows to jump on each other hindquarters and advance through the weigher. The photo of the weighing scale is presented in Figure 4.2.

Figure 4.2: Photo of walk-through weigher.

The apparatus was waterproof for cleaning purposes. Load cells were positioned under the left and right end of the platform. The load cells sent the signal that was amplified in SOEMER LAC universal signal amplifier. The analogue signal was transformed in digital with an analogue to digital converter. Data were recorded with data acquisition software DIADEM installed on industrial personal computer on dairy farm. Calibration parameters were adjusted such that the measured force data are expressed in kilograms. Data were recorded every 0.01 seconds. The rate of hundred samples per second was necessary since the quick changes of the signal during animal crossing over the weighing platform. Yet, this sampling rate was more than required for slow crossing of the animal, the extremely fast walking case produced less than hundred recorded points. To match weights with cows, a complete electronic identification (ID) system was installed. The identification system manufactured by DeLaval consisted of: ear transponder, antenna and control device. The antenna was positioned approximately at a middle of the platform length such that the cows were identified the time they stepped on the scale with front legs. The signals from the identification system were recorded with the same acquisition software as force signals. The synchronized recording of force and cow ID was of great importance for later signal filtering and processing. When cow transponder was in the region of antenna reading, the number was read and recorded. All other recorded values of ID were zero. The walk-through weigher experimental set-up is shown in Figure 4.3.

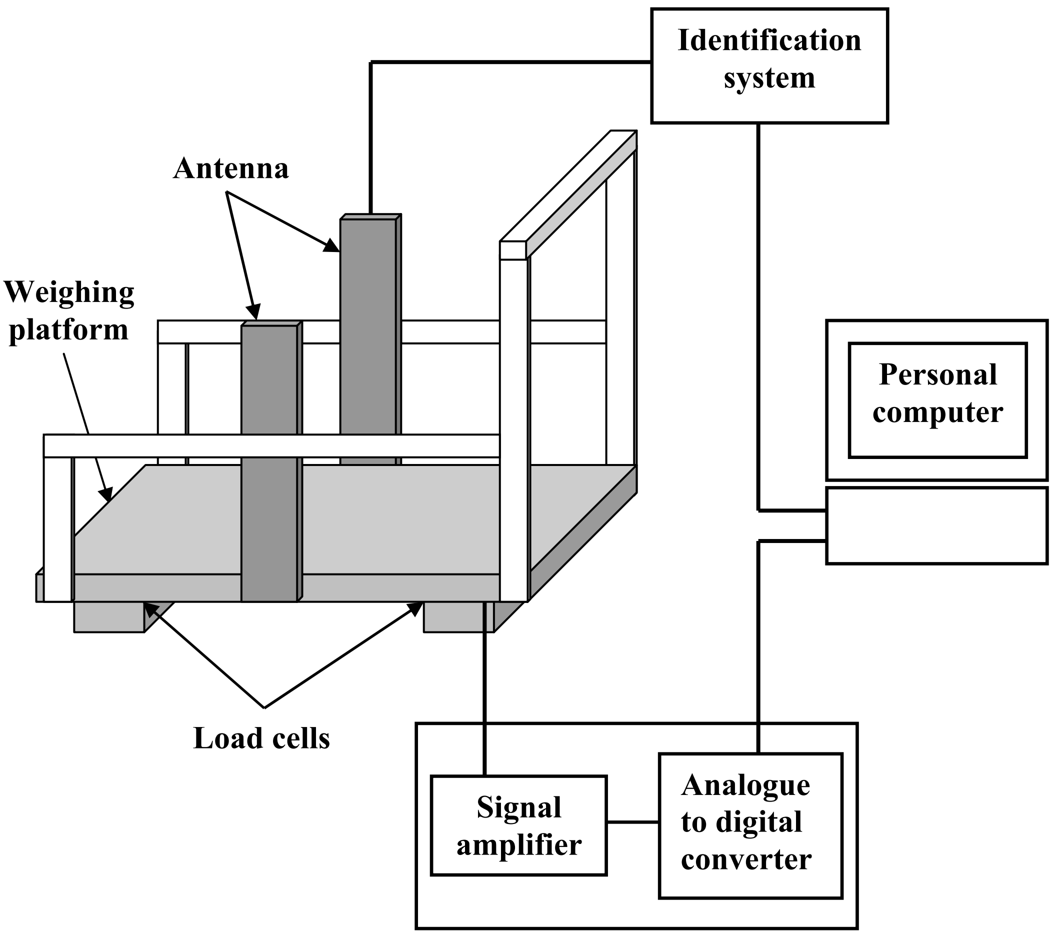


Figure 4.3: Schematic drawing of the identification and weighing system.

The walk-through weigher construction enabled that:

- cows could not fit side by side on the weighing platform
- cows could not ride on each other hindquarter and so advanced through the weigher
- there was always a moment when each cow was alone on the weighing platform with its whole body weight (all four legs on and/or above the platform)
- more than two cows could not act on the weighing platform at the same time

The cattle were measured twice a day after milking. A number of data were gathered for various pace situations. The recorded data was coupled with identification signal for animal. There were two cases of animal crossing the weighing scale. The first was a straightforward situation when the animal was alone on the weigher. The second was a more complicated case, which included two cows touching the weighing platform at the same time. The crossing situations are explained in the next chapter.

4.2 Sequence of weighing on the walk-through weighing scale

Before introducing the signal processing and body weight recognition methods in dynamic weighing the animal crossing manners of the weihger and the adequate measured force need to be exhibited.

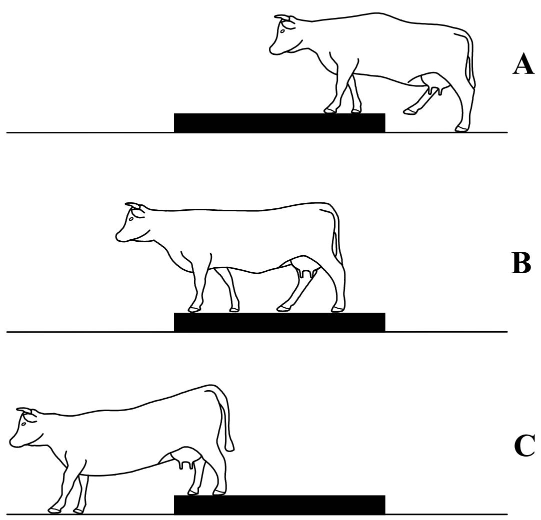


Figure 4.4: The sketch of dairy cow walking over the weighing scale.

The animal crossing cases over the weighing scale can be grouped into two categories:

1) single crossing case
2) crowded crossing case.

The procedure of walk over the weighing scale for single cow is shown in Figure 4.4. The weighing force versus time of a single cow crossing over the weighing scale with 558 kg body weight is presented in Figure 4.5.

As presented in Figure 4.4 and 4.5, the crossing of the weighing platform for single cow consists of three consecutive elements. The first marked with **A** is when forelegs are on the scale (Figure 4.4). Approximately half of the body weight is then measured (see Figure 4.5). The part of crossing when all four legs are on the scale and so the whole body mass is measured is indicated with **B**. The **C** point out the moment when hind legs are touching the scale. This part is similar to the first part (**A**) so the measured weight is approximately the half of the total weight of the cow.

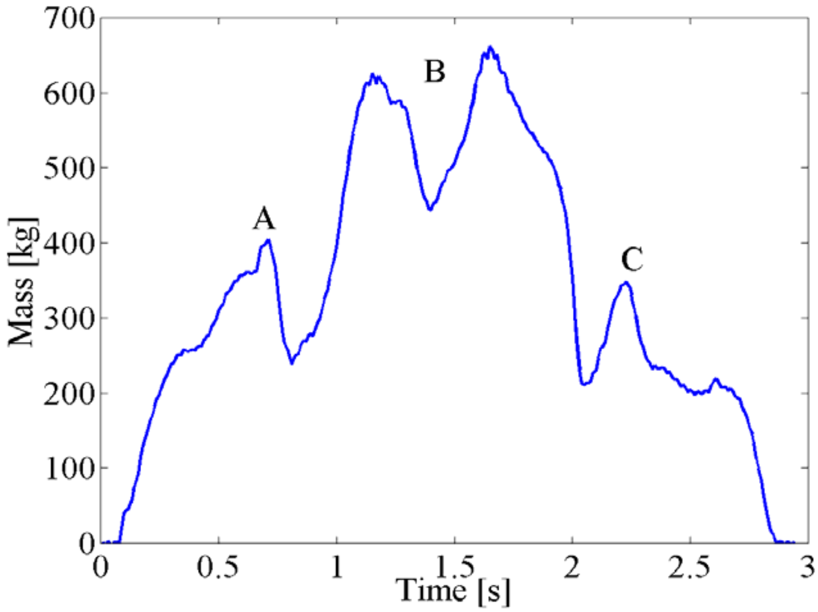


Figure 4.5: Plots of weighing force variation for single cow crossing the weigher. Time of crossing 0.97 s.

After milking in the milking parlour (exp: herringbone) cows leave the parlour in a bunch. The number of animals in the leaving group depends on the size of the parlour. Since the automatic walk-through weigher is placed in the exit corridor of the milking parlour the crowding of the animals occur. The cows are commonly following each other in quick succession over the scale. Due to the construction of the weighing scale the crowding on the scale is always formed with two animals. An example of two cows crowding on the platform is presented in Figure 4.6. The body masses of the cows were 624 and 600 kilograms, respectively. Figure 4.7, gives a plot of measured weighing force during the time when the two cows crossed the scale as sketched in Figure 4.6.

In the case a when one cow with forelegs is on the scale (Figure 4.6) approximately the half of the body weight of the first cow is measured (Figure 4.7). For b is the moment when all four legs are on the scale (Figure 4.6) and so the whole body mass is measured (Figure 4.7). In part c the first cow is on the scale while the following cow steps with forelegs on the scale. The measured force in d is sum of two "half" of the first and second cow. In parts e, f and g the second cow is first with its front, with all four and then with back legs alone on the platform.

Yet, this is only one of four possible crowding cases of two animals on the weighing device. Other cases are presented in the signal processing chapter.

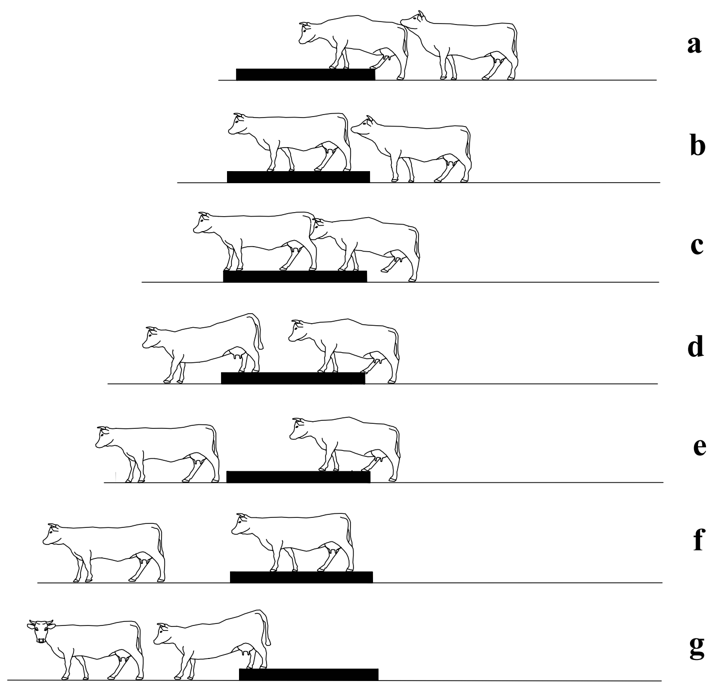


Figure 4.6: Sketch of two cows crowding on the weighing platform.

As presented in the previous plots, there are three types of intervals during crossing of scale that are not of interest. Those intervals are when: only the fore or hind legs of a cattle, the whole body plus fore or hind legs of the other animal and fore and hind legs of different cows are measured. The part of the signal when the animal is alone on the scale with its whole body is of relevance in denoting the correct body weight.

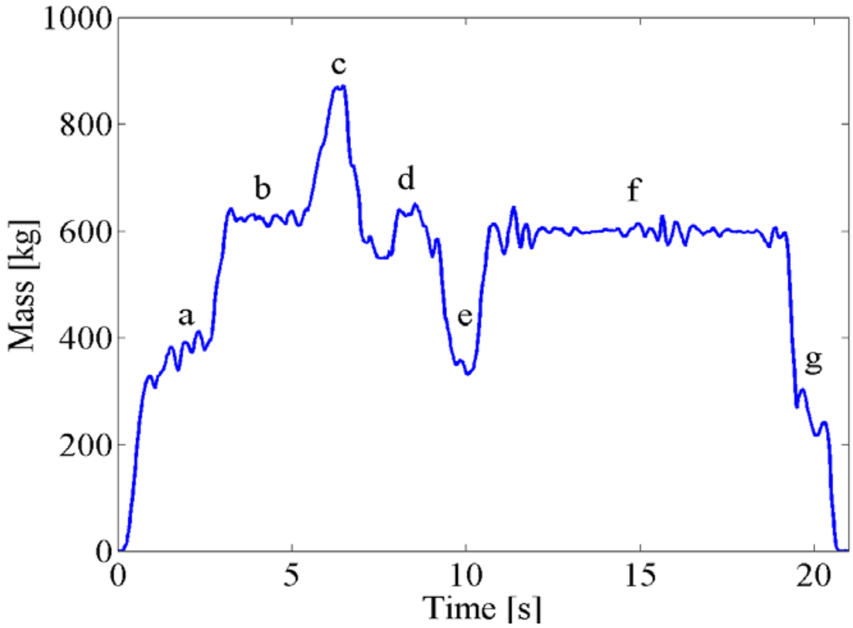
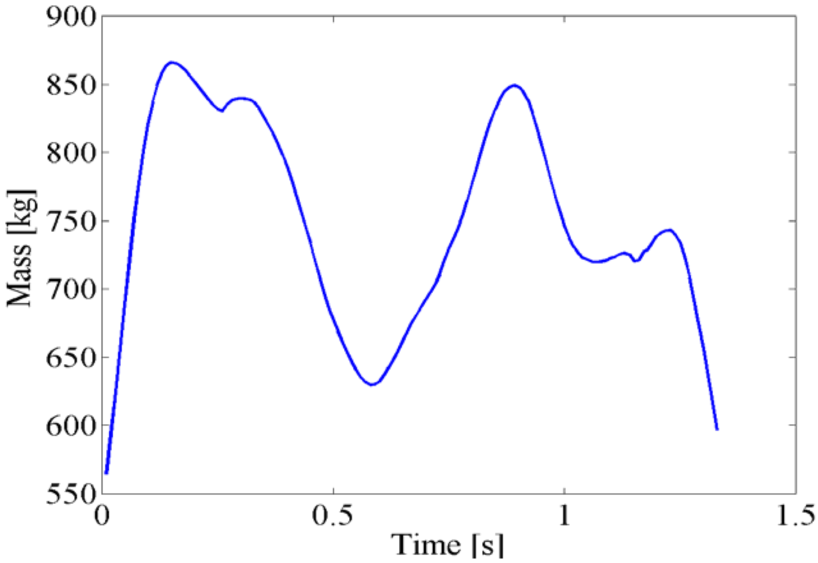


Figure 4.7: Plots of weighing force variation during crowding on the scale. Two consecutive crossings for cows with body weights 624 and 600 kg.

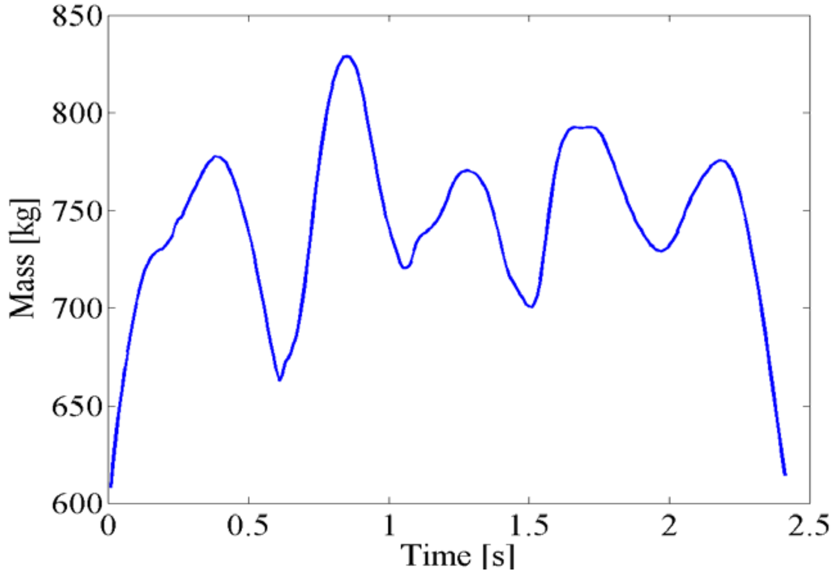
4.3 The influence of velocity of cow movement on the recorded force signal

The cow walk, as well as, any other life being motion is a non-linear and stochastic problem. Consequently, the forces between the scale plate and the animal feet are of varying periodic nature. The produced force depends on many parameters. Some of them are: body weight, walking velocity, step length, body swinging, etc. The most significant are the body weight and walking speed. The walking pace is the major frequency component that influences the reaction force among feet and platform. In order to have better view on the influence of the velocity of cow walk over the weigher on the measured weighing force, three different situations of stride for the same cow are recorded. A cow of 762 kilograms body weight is let over the weigher. Walking velocities vary from slow to fast. The part of the measured signal when animal is with its whole body on the scale is selected and plotted (Figure 4.8). The time that the cow spent on the scale are: 1.32, 2.43 and 4.24 seconds. As shown in Figure 4.8, there is obvious difference in plotted forces for the three paces. The impact between feet and platform increases as walking speed increases. For fast crossing the scale force peaks are larger than those created for medium and slow pace. The force varies between 630 and 865 kilograms for fast stride, while it only fluctuates among 725 and 775 kilograms in slow crossing situation. As the walking speed over the weigher decreases, the easier is to estimate the correct body mass of the cow.

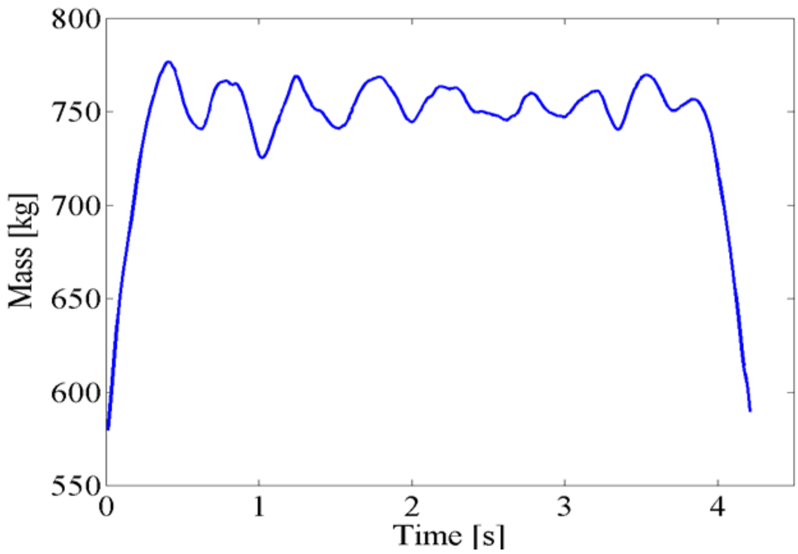
According to above presented, the best case in automatic walk-through weighing would be if the cows would cross the scale separately and with low velocity. Out of the rotary milking parlours animals are separated because of the specific milking method. Unfortunately the majority of milking parlours are stationary and the cows are let to cross the scale as a bunch after milking. Consequently, the methods of denoting the valuable part of the signal and the body weight recognition needed to be applied. In the next chapters the signal filtering methods are explained followed by the body weight recognition techniques.



(1)



(2)



(3)

Figure 4.8: Plots of weighing force variation for single cow crossing the weigher with its full body weight. Times of crossing: (1) 1.32 s, (2) 2.43 s and (3) 4.24 s.

5 Signal processing of recorded raw data

Signal processing is an important part of body weight recognition process of dynamic livestock weighing. Filtering out the signal from noise, as well as, correct separation of valuable part of the signal, influences the accuracy of the body weight estimation. As mentioned earlier, there are two cases of cattle crossing the weighing scale: single and crowded case.

Single case is the case when only one animal alone crosses the weighing scale (Figure 4.1, 4.2). In single case crossing, force signal is characterized by the zero force measurement between two consecutive crossings.

Crowded case is when animals are following each other in quick succession over the weighing scale (Figure 4.3). In this case, without identification information it could not be possible to distinguish the valuable part of the signal (Figure 4.4). The valuable part of the force versus time signal is measured when an animal was alone with its whole body weight on the weighing platform.

The aim of the signal processing is to separate the valuable part of the force signal. The method is explained for single and crowded crossing cases with examples.

5.1 Single crossing case

To explain the method of signal filtering and separation an example is presented. A cow with 618 kilograms body weight was let across the weighing scale and force was measured. Plot of measured force versus time is shown in Figure 5.1. Performing discrete Fourier transformation on measured force signal with mathematical formula

$$X(k) = \sum_{n=1}^{N} x(n) \exp\left(\frac{-2\pi i (k-1)(n-1)}{N} \right),$$

a signal in frequency domain is created, where N is number of measured points, k is a coefficient between 1 and N, and $x(n)$ is the measured force in the time sample n. As shown in Figure 5.2, applying absolute value of Fourier transformation values for measured signal the frequency domain of the force signal is formed.

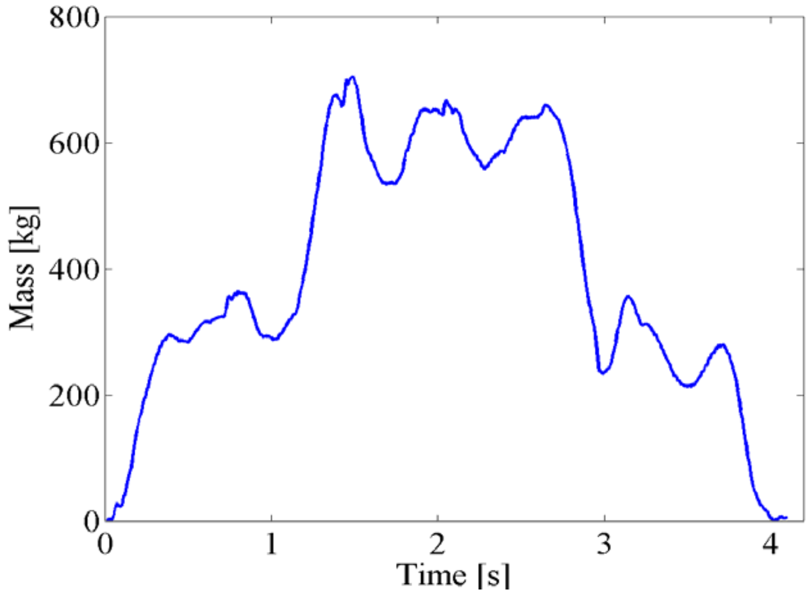


Figure 5.1: Measured force signal of 618 kg cow.

The signal is transformed excluding all frequencies except zero frequency, which carries the information of the body weight and walking velocity. The signal was filtered with the use of rectangular window of length L ($L = L_2 - L_1$):

$$w(n) = \begin{cases} 1, & L_1 \leq n \leq L_2 \\ 0, & elsewhere \end{cases},$$

which was multiplied with the measured force signal. Consequently, only a part of the signal was transformed in frequency domain. All values related to higher than zero

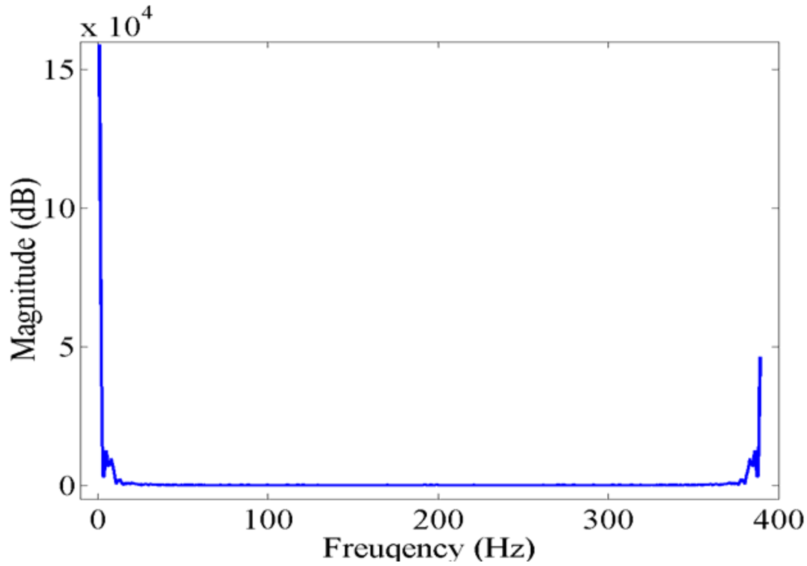


Figure 5.2: Frequency domain presentation of the force signal.

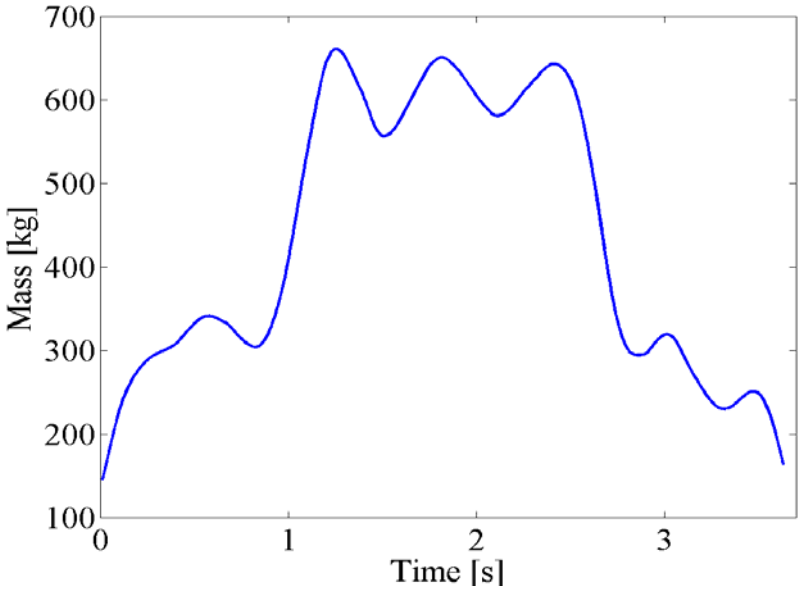


Figure 5.3: Plot of force in time after filtering.

frequency were set to zero. Inverse Fourier transformation turned the filtered signal in time domain. However, only the first value in the row of returned values was selected and the position in a new processed signal was assigned. The window was then moved from beginning toward the end of the signal such that the L_1 and L_2 are increased by one. The window filtering method was performed on the whole signal length. The created signal is presented in Figure 5.3. So obtained signal was much smoother than initially measured. The filtered signal was still containing the part where the animal was touching the scale not only with its four legs but also the transient parts of crossing (step on and off). To eliminate the transient parts of the signal another filtering needed to take place. This process was not as simple as the previous one, since there were no existing rules for the filter, but they had to be determined from experiments and practice.

Discrete Fourier transform was performed on the signal and the frequency domain representation was obtained. The rectangular window method was performed in this process. The length of the window was determined according to the signal length. The absolute value at the first frequency was assigned for each window as it progressed through the signal. A new signal was created as shown with dashed line in Figure 5.4. The abrupt changes in the force signal were manifested by jumps in the new signal (dashed line). The transient points in the signal were recognised and the signal valuable part was detected. As

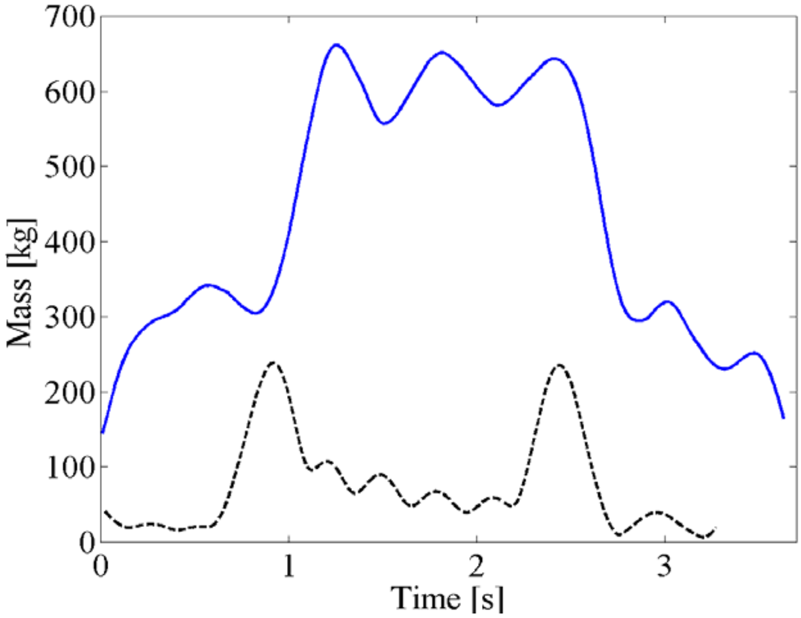


Figure 5.4: Transient points recognition plot. Dashed line has the magnitude of Fourier transform.

shown in Figure 5.4, transient stages were at 0.91 and 2.53 second. Cutting out the valuable part of the force signal a new force signal measured during the stay of the animal with its whole body weight was formed. The selected part of force signal is plotted in Figure 5.5.

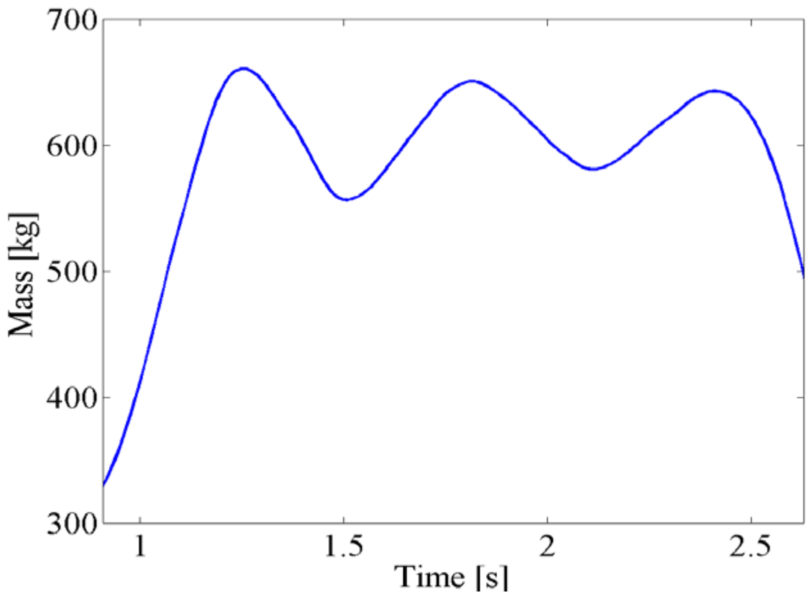


Figure 5.5: Selected valuable part of the measured force signal.

For the case of single animal crossing the weigher the process of finding the transient parts of the force signal is completely defined. In other words, an algorithm was formed to find all maximums of the signal created by second proposed filtering method. The two biggest maximums pointed out the moment when an animal stepped on and off the weighing scale.

5.2 Crowded crossing case

As mentioned earlier, the most common case of crossing the weigher on dairy farm is the crowded case. An example of six cows weighed during consecutive crossing the weighing platform is here analysed. The plot of force signal combined with identification signal is presented in Figure 5.6. The identification signal is presented as a discrete unit pulse for each time when a cow was detected by the identification system. As explained in previous section, the identification detection occurred approximately at the moment when the cow stepped on the weighing platform. Filtering the force signal from noises was done in the same way as explained for single crossing situation. The major difficulty was the second filtering that selected the valuable part of the force signal. The construction of the weigher enabled that the maximum of two cows simultaneously touch the platform. Consequently, analysing all crossing situations for two animals was enough to understand even the most complex cases of weighing scale crossing. In Figures 5.7, 5.8, 5.9 and 5.10 possible situations of force signal measurement for two cows are presented.

In Figures 5.7, 5.8, 5.9 and 5.10 idealised force signals of two cows with body weights of 600 and 800 kilograms are drawn. These Figures consist of three windows. The first cow force signal is plotted in upper window, while the second force signal is presented in middle window. The two signals in upper and middle plots are summed and produced lower plot, which is, in fact, the kind of signal that is measured when two cows are following each other over the weighing scale. The plots in Figures 5.7, 5.8, 5.9 and 5.10 are showing the decomposed force signal. They are basically showing how the final force signal is formed. Consecutive crossings over the weigher are characterised by the fact that the previous animal still acts on the weighing platform when the following animal stepped on the scale. The construction of the weighing device does not allow two cows fit side by side on the scale. In other words, maximum of three pair (two pair of one animal and one pair of the other) of legs can fit on the scale.

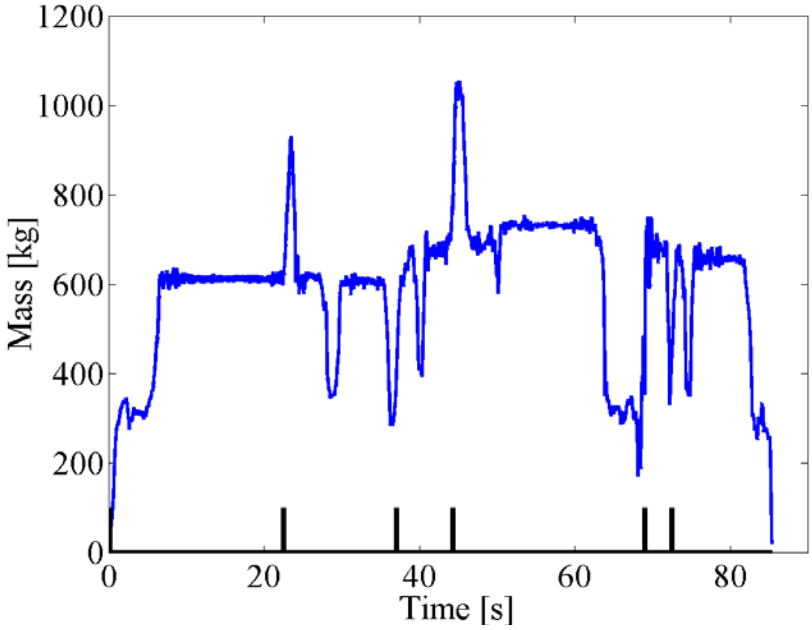


Figure 5.6: Plot of force signal (upper solid line) coupled with identification signal (bottom solid line).

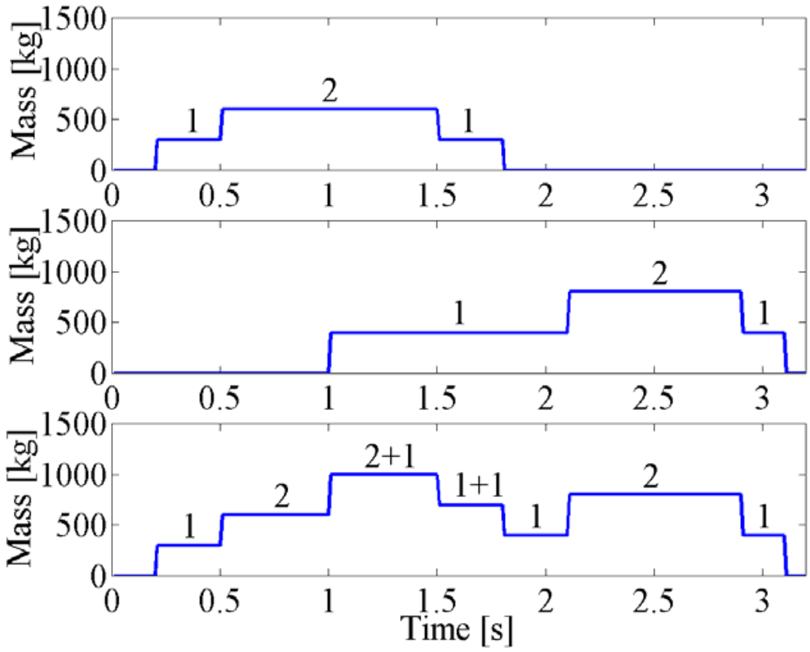


Figure 5.7: Plots of force signals for two cows consecutively crossing a weigher.

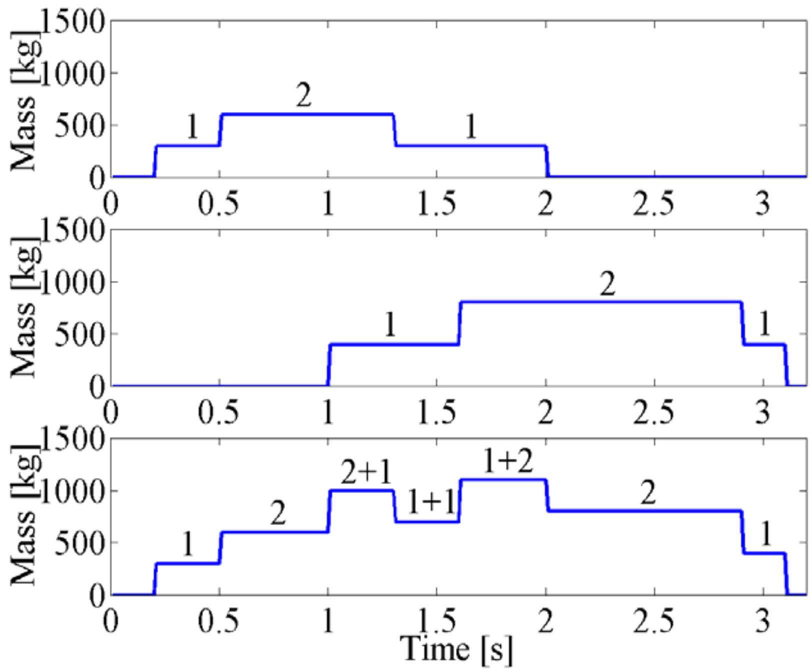


Figure 5.8: Plots of force signals for two cows consecutively crossing a weigher.

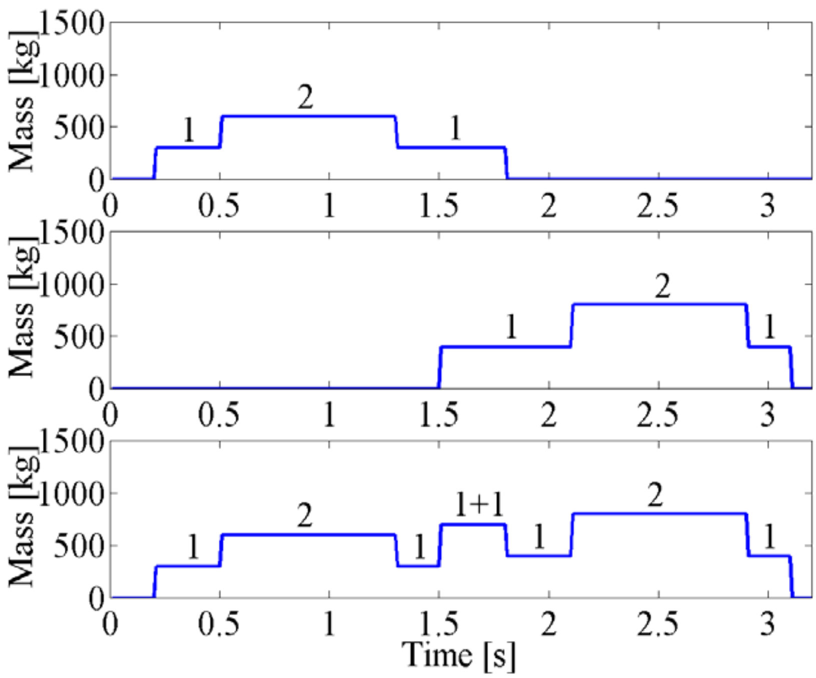


Figure 5.9: Plots of force signals for two cows consecutively crossing a weigher.

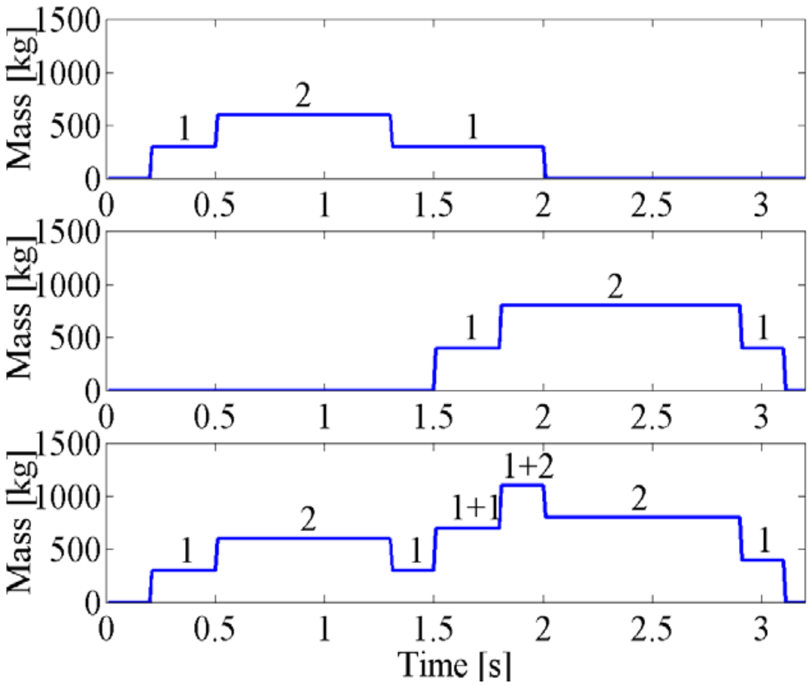


Figure 5.10: Plots of force signals for two cows consecutively crossing a weigher.

In Figures 5.7, 5.8, 5.9 and 5.10, one pair of legs is symbolised by “1”, two pair of legs of the same animal by “2”, one pair from one animal and one pair from the other by “1+1”, and two pairs from one animal and one pair from the other by “2+1”. Yet, when the first animal touches the scale with all four legs and the second animal affects the scale with its front pair of legs, the symbol is “2+1”. Vice versa, when the first animal touches the scale with hind pair of legs and the second animal affects the scale with all four legs, the symbol is “1+2”. In Figure 5.7 the second animal stepped on the scale with its front legs, while the first was still with all four legs on the scale (case **c** in Figure 4.3). The front legs of the second animal continued to affect the scale after the first animal stepped off with its front legs (case **d** in Figure 4.3) and later with its hind legs (case **e** in Figure 4.3). The second animal moved across the scale after the first animal disengaged completely (cases **f** and **g** in Figure 4.3). In Figure 5.8, the second animal moved forward after the first stepped off with its front legs and created space for the second to apply fully. In Figure 5.9, the second animal moved on the scale some times after the first animal stayed on the scale with its back legs producing a jump in the force signal symbolised by “1+1”. The second animal did not step on the scale with its back legs until the scale did not clear. In Figure 5.10 a similar situation of pace as in Figure 5.9 is presented. The difference is that the second cow fitted on the platform while the first had its hind legs on the platform, marked by “1+2”.

The four previous figures represent the possible cases of consecutive crossings over the weighing scale. Still, length of the stages “1”, “2”, “1+1”, “2+1”, and “1+2” vary from case to case. According to previously proposed, there are four main situations of crowded crossing over the weighing scale. Between the two consecutive full body weight (“2”) measurements in crowded crossing case, the following situations may occur:

1) “2“ – “2+1” – “1+1” – “1” – “2”
2) “2“ – “2+1” – “1+1” – “1+2” – “2”
3) “2“ – “1” – “1+1” – “1” – “2”
4) “2“ – “1” – “1+1” – “1+2” – “2”

In reality the length of some crossing stages “1”, “1+1”, “2+1”, and “1+2”, might be short or zero and so the four main situations can be simplified. The real situations of crowded crossing for two animals over the scale were measured and the force signals are plotted in Figures 5.11, 5.12, 5.13 and 5.14.

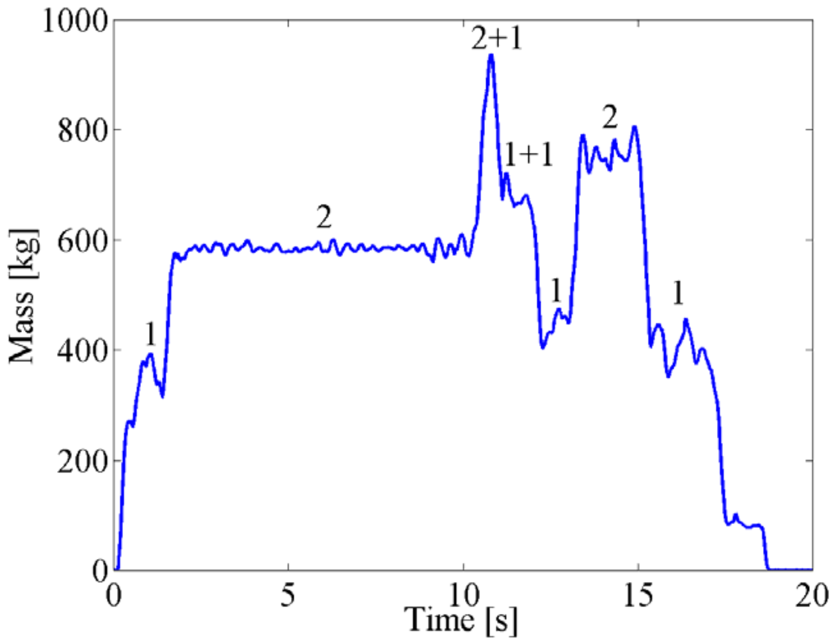


Figure 5.11: Force signal plot for two consecutive crossings (585 & 762 kg).

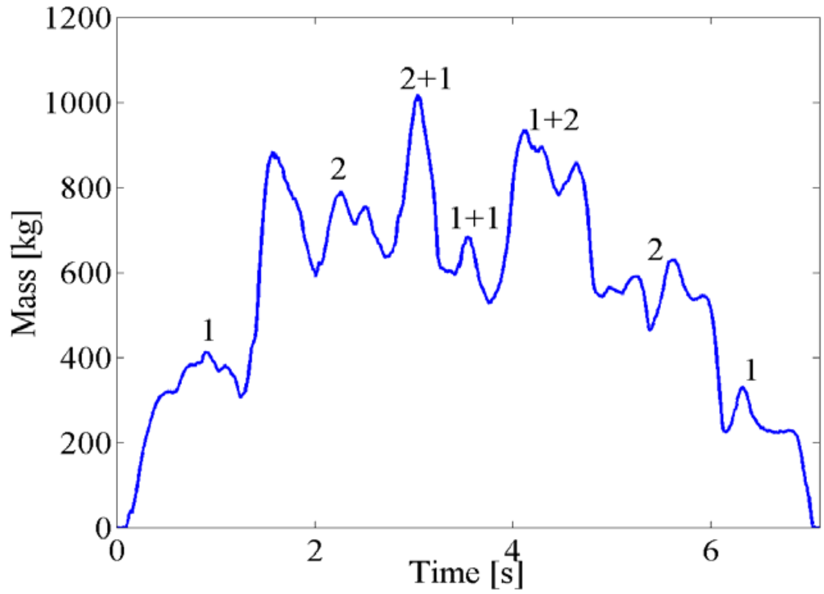


Figure 5.12: Force signal plot for two consecutive crossings (723 & 558 kg).

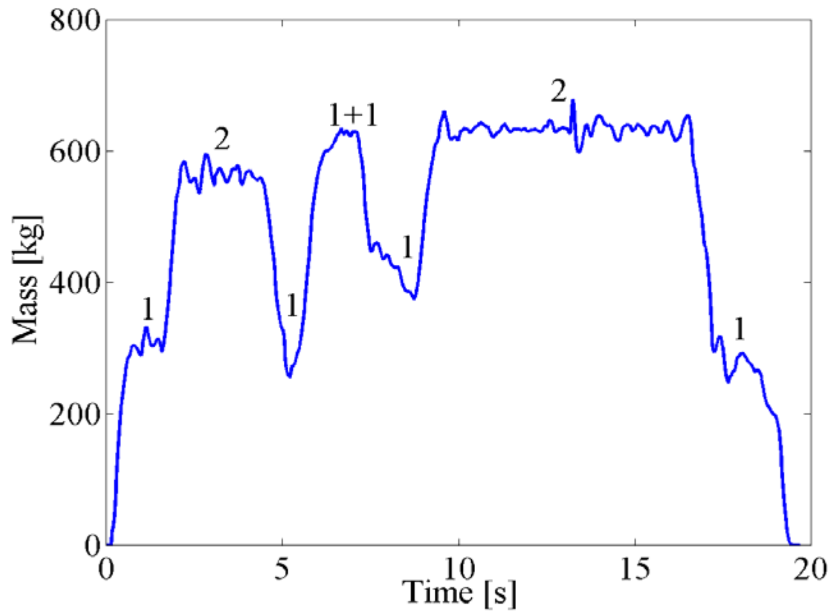


Figure 5.13: Force signal plot for two consecutive crossings (567 & 633 kg).

The four figures represent four proposed crowded crossing situations. As shown in Figures 5.11, 5.12, 5.13, and 5.14 the transient parts are not so visible as in ideal crossing cases shown in Figures 5.7, 5.8, 5.9 and 5.10. Especially, the lasting of transient parts of case 3 (“2“ – “1” – “1+1” – “1” – “2”) when only the front or back pair of legs was touching the scale (“1”) were almost zero and so invisible on force signal plot as presented in Figure 5.15.

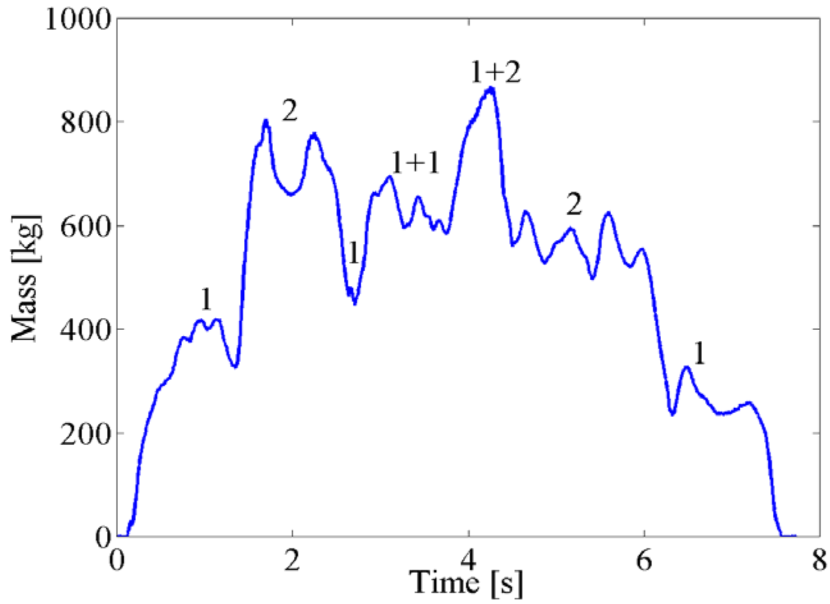


Figure 5.14: Force signal plot for two consecutive crossings (723 & 558 kg).

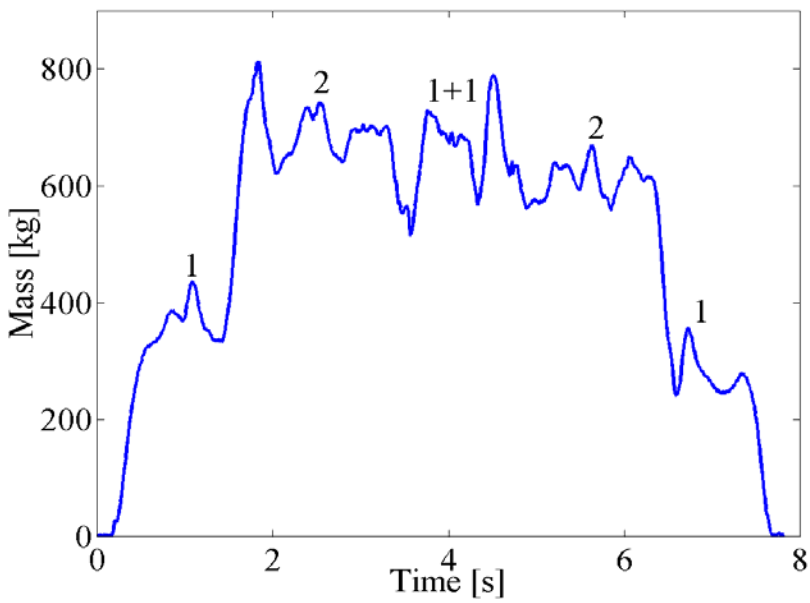


Figure 5.15: Force signal plot for two consecutive crossings (700 & 618 kg).

In the case of animals crowding on the weighing scale one of the four explained situations of two animals crowding occurred. The filtering of crowded force signal was done in the same way as proposed for single crossing case. The sampling window was selected according to recorded signal length and the discrete Fourier transform was applied. The absolute value of the first frequency value was calculated and the position in the new signal was given. The new created curve is presented together with force signal in Figure

5.16. The dashed line curve is the new created curve to denote transient points in the force signal. Vertical lines are drawn to mark the transient points. Dash-dotted vertical line is presenting the moment an animal stepped on the scale. This moment coincides with identification signal record.

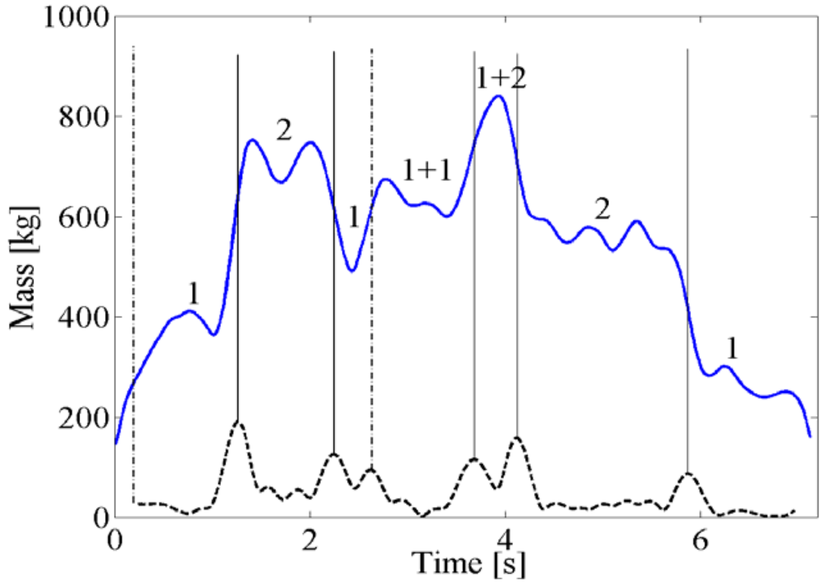


Figure 5.16: Transient points recognition plot.
(Dashed line has the magnitude of Fourier transform)

Unlike single crossing case where two maximum jumps of the new signal denoted the transient parts of the force signal, in crowded crossing case the situation was slightly trickier. The procedure of selecting the valuable part of the force signal for crowded crossing case is here explained. The force signal was separated into sections. Beginning of each section was the step on recognition from the identification signal and the end was the next recognition of the following animal. Six maximums of the first frequency curve are found for each section. This was chosen since the maximum of six transition points could occur, as previously explained. These five subsections were tested for body weight and time match. The construction of the weigher and the physical characteristics of the cattle determined the minimum of time spent on the scale when animal was alone with its whole body weight on the scale. Consequently, a boundary was stated for subsection time inspection. The time boundary was one second. Force signal corresponding to subsections were averaged. The mean value was compared to previously measured weight of the identified animal. The results from subsection body weight and time match were analysed. An algorithm inspecting all possible cases and combinations was developed. In the case when some transition points did not exist the algorithm lowered down the number of subsections combining the neighbouring subsections that had close average values of force signal. Another significant problem caused by close values of the average force signal of not neighbouring subsections with previously measured body weight was resolved by choosing the most right subsection as the valuable part of the signal. This situation occurred when previous and following animals had similar body weights. It caused the body weight inspection to produce similar results for subsection “1+1” and subsection “2”. As previously proposed, subsection “2” came as last before the section ending.

Finally, flow charts presenting the complete procedure of force signal noise filtering and valuable part filtering, as well as, algorithm for different situation inspection are shown in Figure 5.17, 5.18 and 5.19.

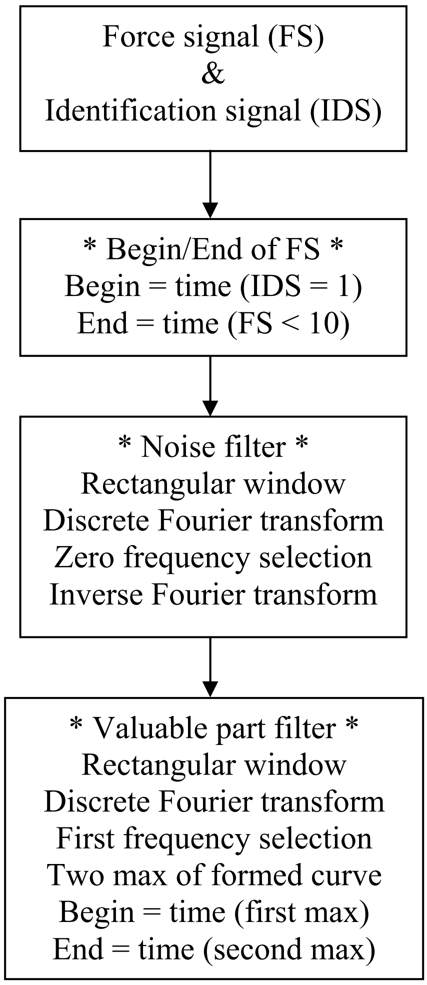


Figure 5.17: Flow chart of force signal filtering for single animal crossing case.

After force signal filtering was successfully done, it was possible to continue with body weight recognition procedure. There are three different methods of body weight recognition explained in the following sections.

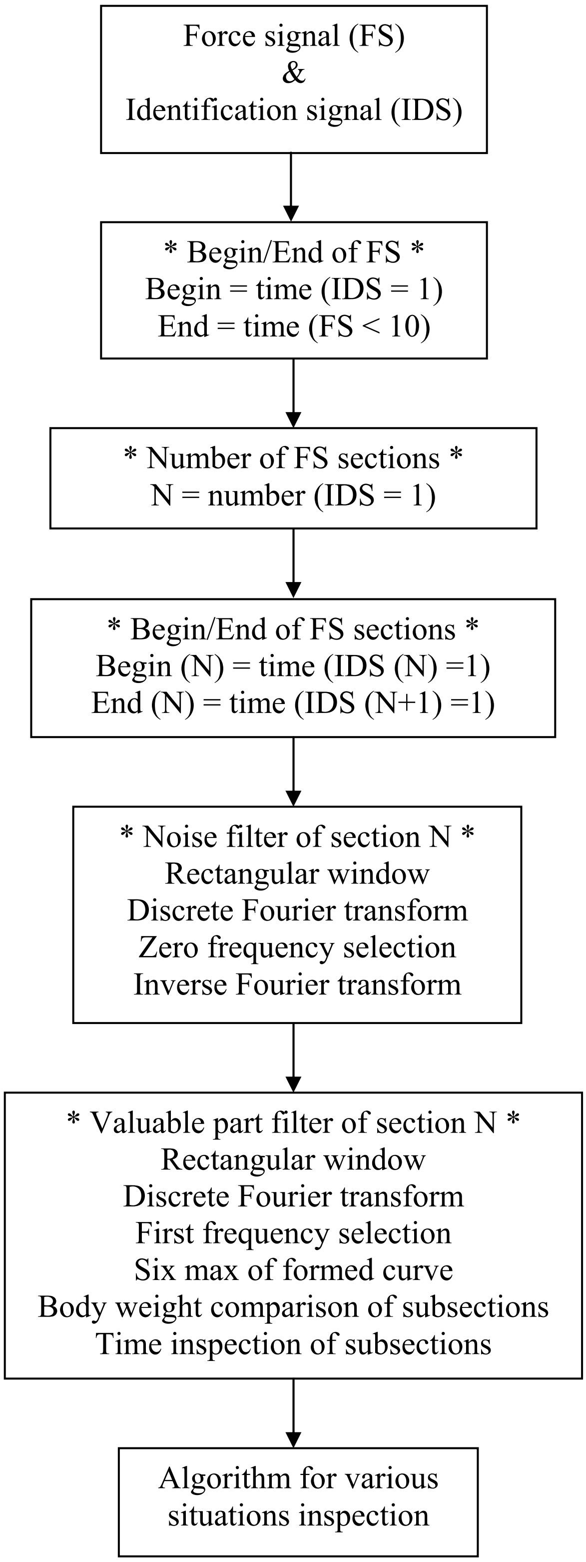


Figure 5.18: Flow chart of force signal filtering for crowded crossing case.

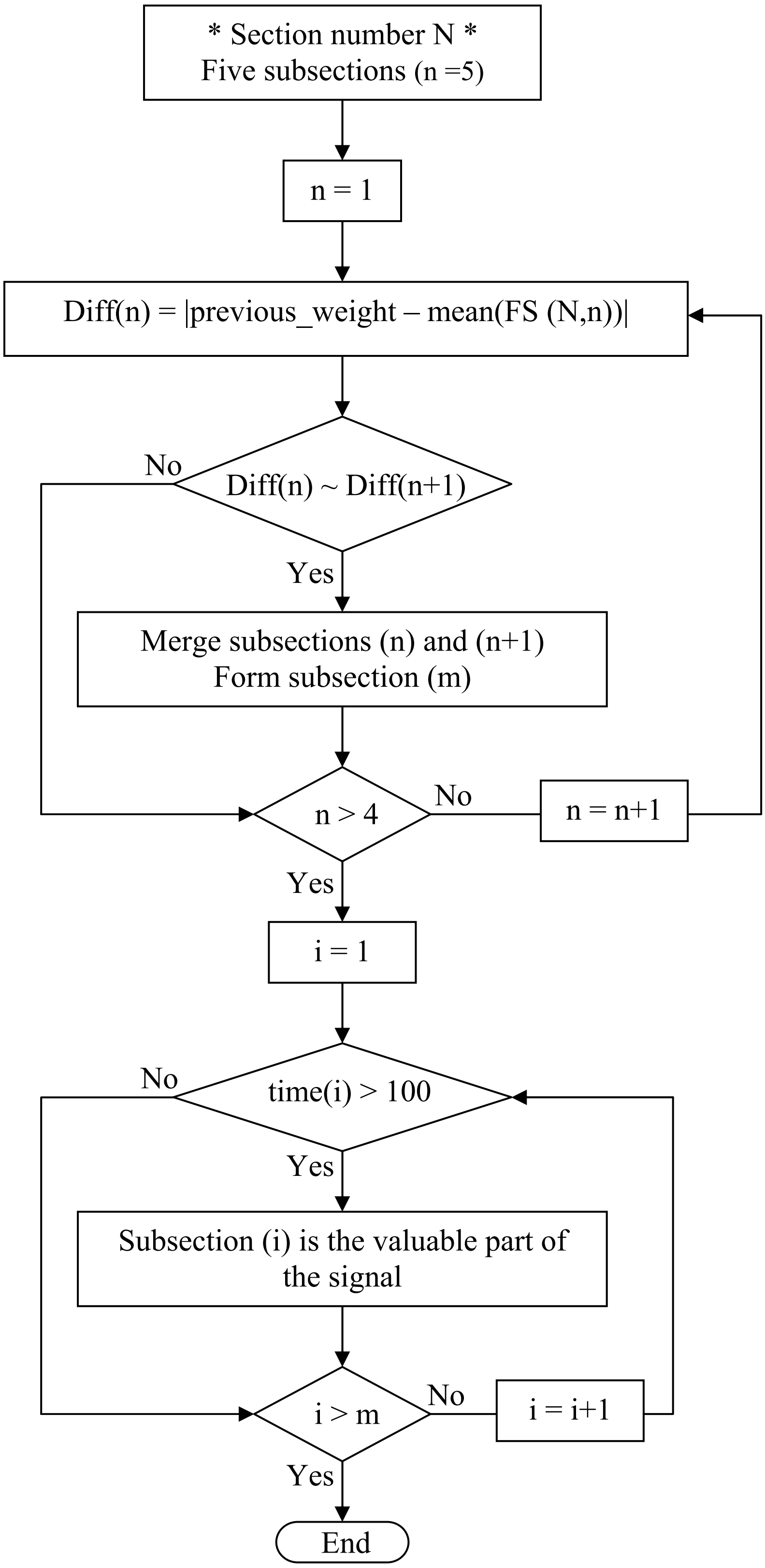


Figure 5.19: Block diagram of algorithm for inspection various crossing situations.

6 Development of body weight recognition methods

Three methods of denoting correct body weight of the animal weighted on the walk-through weighing scale are presented in the following sections. Mathematical modelling method, neuro-fuzzy method and neural networks method are explained.

6.1 Mathematical model of cow locomotion

6.1.1 Model creation

The model created and presented in this section was inspired from previous research in the field of human and animal locomotion. The aim was to develop a physical and mathematical model that could simulate cow walk. However, the model had to be applicable for the use in body weight recognition from the measured force signal.

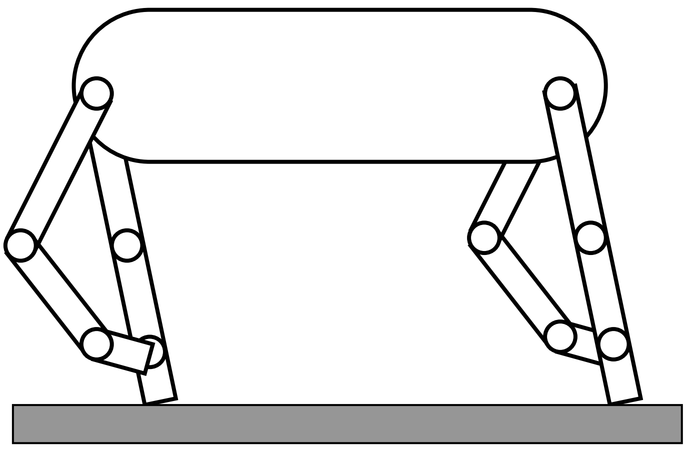


Figure 6.1: Physical model of cow locomotion.

The force signal gave information on intensity of vertical force produced between animal feet and weighing platform and time of crossing lasting. Lasting of crossing determined the walking velocity, since the known platform length. Therefore, the model had to combine three parameters: walking velocity, body weight and vertical force produced by feet. On the other hand, systems like four legged animals are highly sophisticated. All four legs are moved independently. Each leg is composed of three parts that perform rotation independently. This leads to the number of twenty-four parameters to control. Twelve are the torques produced by muscles to move each leg segment and the other twelve are the angle position of each leg segment. Additionally, there are seven parameters characterising masses of different body parts (1 – body mass, 3 – front legs, 3 – hind legs) and six parameters symbolising lengths of leg segments to adjust (3 – front legs, 3 – hind legs). Such a model produced directly from real system is presented in Figure 6.1. Due to its complexity it is not convenient for study of locomotion of the cattle. Some simplifications has to be introduced. Once the foot touches the ground the leg stays straight until the end of interaction with the ground. As a result, the leg can be examined not as a three segments element but as a single rigid one. This lowers down the parameters of the system almost three times.

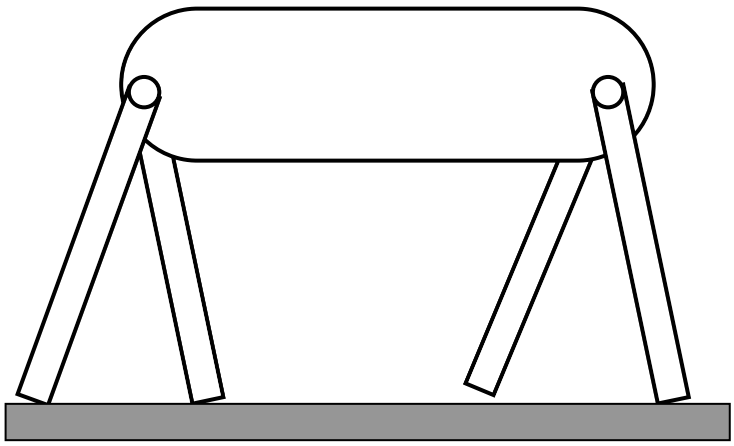


Figure 6.2: Simplified model of cow locomotion.

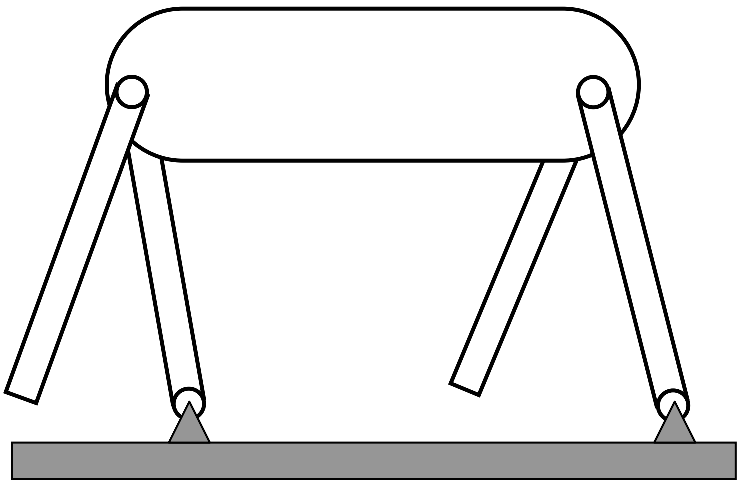


Figure 6.3: Pendulum type model of cow locomotion.

The simplified model is shown in Figure 6.2. An important notice, at the transition stage when the leg is without contact to the ground, the leg have to shorten otherwise it would touch the ground when it is not desired.

The model shown in Figure 6.2 symbolises a mass with four pendulums attached to it. Yet, the legs that are in contact with the ground perform as inverted pendulum moving the mass on arc. The free legs behave as pendulums moving from the position when they lost contact with ground to the position when they touch the ground again and become inverted pendulums. To achieve body motion during the inverted pendulum stage one front and one hind leg need to touch the ground simultaneously. Cattle move their diagonal legs roughly at the same time.

In Figure 6.3 the pendulum - inverted pendulum model is shown. In reality cattle do not move their diagonal legs precisely simultaneously. There is always some time shift. The position angles of diagonal legs change in approximately same manner. Simultaneous movement of diagonal legs with exactly same angles is the ideal case of cattle walk. The model of ideal walk is similar to that in Figure 6.3. Ideal model has the legs with same length and masses. The diagonal legs move at the same time for the same angle interval. Consequently, the ideal model can be unified in a simpler one. The assumption that the four-legged animal simultaneously moves its diagonal legs caused the four-legged being to be seen as a two legged being. The simplified ideal walking model is shown in Figure 6.4.

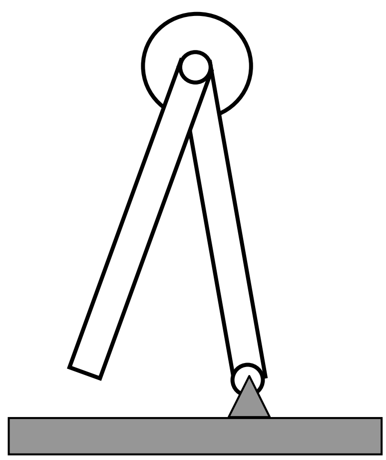


Figure 6.4: Simplified pendulum model of cow locomotion.

In this model the mass of each leg is double the mass of one leg of four legged pendulum model. The simple model of cow walk consists of: link bar and a point mass at its end, a free bar connected to the point mass and a motor that drives the rigid bars (Figure 6.5). The model is similar to those proposed as human walk model. Simplifications introduced to four legged model are:

1) simultaneous movement of diagonal legs
2) straight legs during interaction with ground

The mass and the length of the bars are *m* and *l*, respectively. The point mass is marked with *M*. The point mass stands for body mass of the animal, while the two bars represent legs of the cattle. The torques T_1 and T_2 which move the bars are in reality produced by muscles in order to move the legs. The angle positions of the bars are θ and φ.

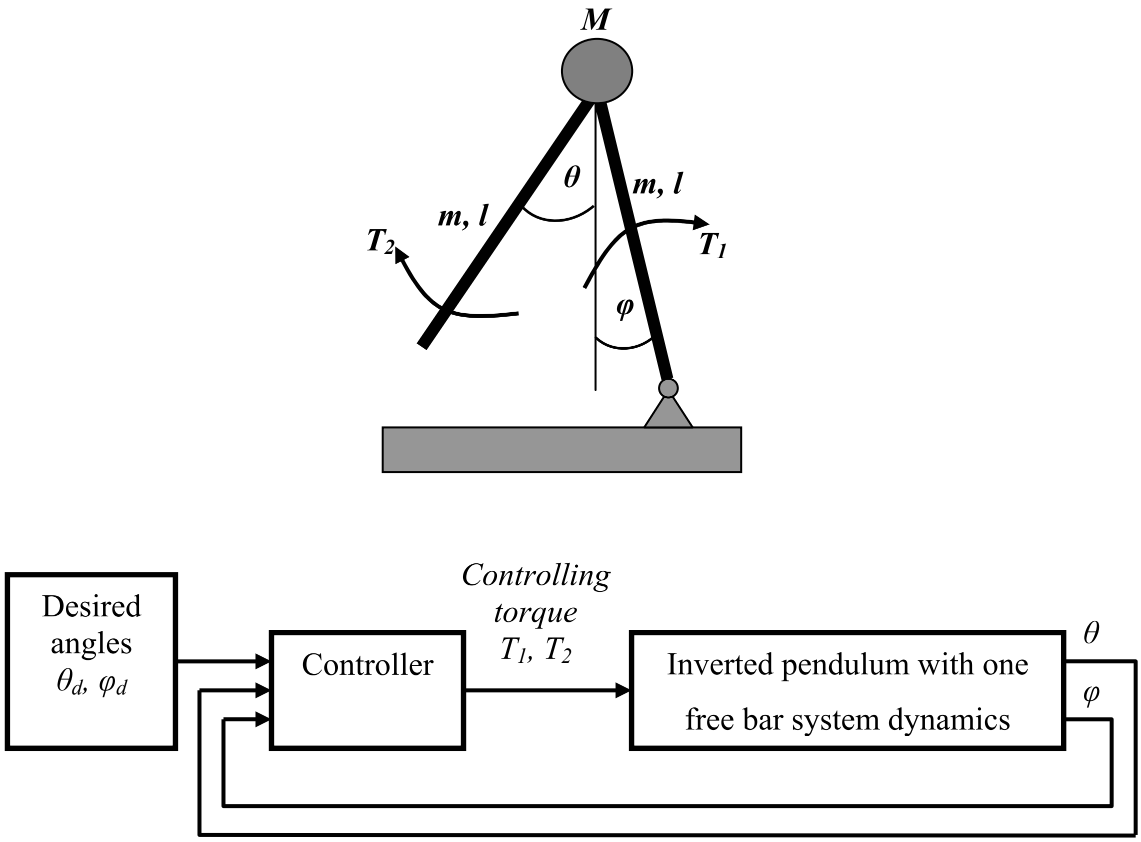


Figure 6.5: Physical model of cow walk and control schematic.

The walking of two legged being is the rotation of the active leg around the foot and the rotation of the passive leg around the hip. The active leg changes into passive leg and vice versa during walk. The active leg starts off with a negative value of angle to the vertical and ends approximately at the same angle value but with positive sign. During the rotation of an active leg, the passive leg rotates around the hip until the ground contact when it becomes the active leg and the active becomes the passive one. In ideal case the walking begin and end angles of active and passive leg are equal and same for both legs, as well as the angular velocities of the legs. Muscles move the legs according to the command from brain. Similar to that concept the bars in the model are moved with torques. The torques are applied to the bars dependent on the information from the controller. The controller tracks the angle positions of the bars, θ and φ, and compares it to the desired angle values, θ_d and φ_d. The simple pendulum model of cow movement with controller is shown in Figure 6.5.

Walk simulation with simple pendulum model is done analogue to that explained for two legged being. It is chosen that instead of moving the whole system like during walking the rotation around two joints, one at ground and the other at the point mass, produce the same result as real motion. The walking process is presented as vibration of inverted pendulum and pendulum attached to the end of the inverted pendulum. The oscillation of the model is shown in Figure 6.6.

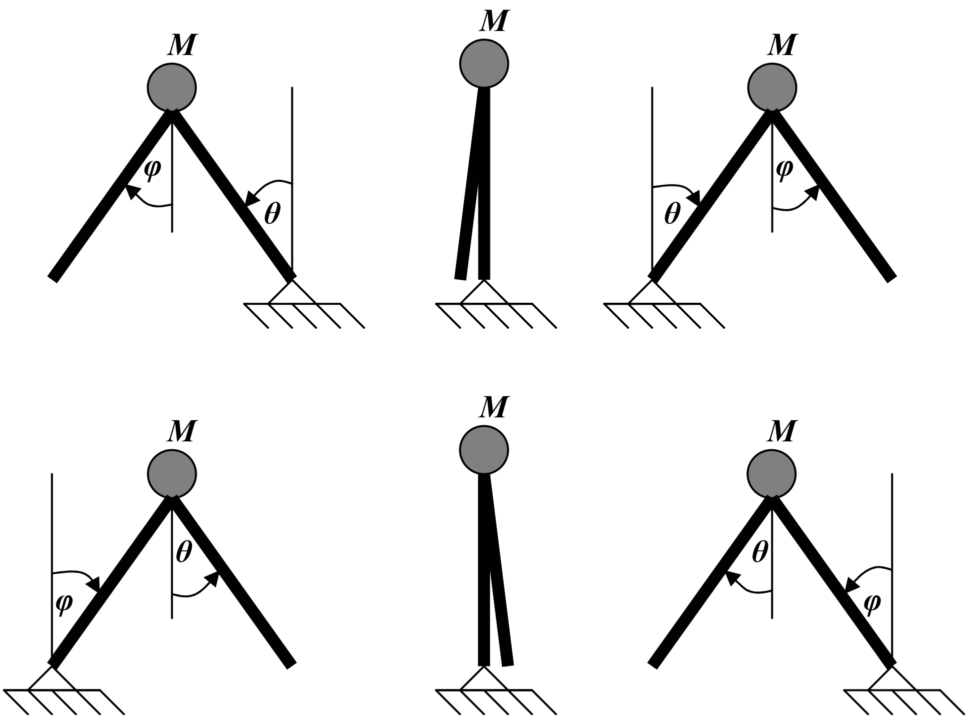


Figure 6.6: Simulation of walking process.

As in walking when legs switch at every step, the bars switch such that active becomes passive and passive becomes active. In simulation it is done by switching the angles. The angles θ and φ change their names and θ becomes φ, and vice versa. The fool step is accomplished when the bars complete one oscillation from left most to right most or from right most to left most position. The controller tracks the desired position angles of bars and applies amount of torque to each bar so it minimize the tracking error. During that procedure the torques change their direction. The desired angles change their values as sinusoidal function. Angle values are shifted for 180 degrees, such that the angles have the same amount just different signs.

The equation of motion of simple pendulum model is here derived. A drawing of physical model according to which the equations of motion are derived is shown in Figure 6.7. In Figure 6.7 the forces and torques acting upon the system, as well as, the relevant angles and points are presented.

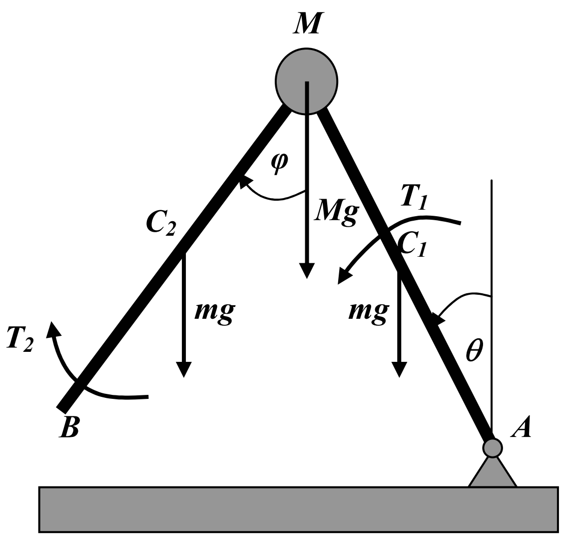


Figure 6.7: Forces and torques applied on the system.

The kinetic energy of the model plotted in Figure 6.7 is

$$E_K = E_{K1} + E_{K2} + E_{K3}, \tag{1}$$

where E_{K1} is the kinetic energy of rotating bar (inverted pendulum), E_{K2} is the kinetic energy of the point mass at the end of the bar, and E_{K3} is the kinetic energy of the bar with plain motion (free pendulum). Kinetic energies are

$$E_{K1} = \frac{1}{2} I_1 \dot{\theta}^2, \tag{2}$$

$$E_{K2} = \frac{1}{2} M l^2 \dot{\theta}^2, \tag{3}$$

$$E_{K3} = \frac{1}{2} m v_{C2}^2 + \frac{1}{2} I_{C2} \dot{\varphi}^2, \tag{4}$$

where I_1 and I_{C2} are the moments of inertia of the bars at the centre of rotation *A* and centre of bar C_2 for rotating bar and planar motion bar, respectively. Velocity at the mass centre C_2 is v_{C2} and is calculated from position of mass centre on the x and y axes

$$x_{C2} = l\sin\theta + \frac{l}{2}\sin\varphi, \tag{5}$$

$$y_{C2} = l\cos\theta - \frac{l}{2}\cos\varphi. \tag{6}$$

Velocity at the middle of free bar is calculated as

$$v_{C2} = \left(x_{C2}^2 + y_{C2}^2\right)^{1/2}. \tag{7}$$

Substituting (5) and (6) in (7) denotes

$$v_{C2} = l\left(\dot{\theta}^2 + \frac{\dot{\varphi}^2}{4} + \dot{\theta}\dot{\varphi}\cos(\theta+\varphi)\right)^{1/2}. \tag{8}$$

The moments of inertia of bars are

$$I_1 = \frac{1}{3}ml^2, \tag{9}$$

$$I_{C2} = \frac{1}{12}ml^2. \tag{10}$$

Substituting equations (9) in (2), and (10) and (8) in (4) the kinetic energy equations are formed

$$E_{K1} = \frac{1}{6}ml^2\dot{\theta}^2, \tag{11}$$

$$E_{K3} = \frac{1}{2}ml^2\left(\dot{\theta}^2 + \frac{\dot{\varphi}^2}{4} + \dot{\theta}\dot{\varphi}\cos(\theta+\varphi)\right) + \frac{1}{24}ml^2\dot{\varphi}^2. \tag{12}$$

Therefore, substituting (11), (3) and (12) in (1) the kinetic energy of the whole system is

$$E_K = \left(\frac{2}{3}ml^2 + \frac{1}{2}Ml^2\right)\dot{\theta}^2 + \frac{1}{6}ml^2\dot{\varphi}^2 + \frac{1}{2}ml^2\cos(\theta+\varphi)\dot{\theta}\dot{\varphi}. \tag{13}$$

Potential energy of the system is produced by potential energy of bars and point mass,

$$E_P = E_{P1} + E_{P2} + E_{P3}. \tag{14}$$

Potential energy of the rotating bar (inverted pendulum) is

$$E_{P1} = mg\frac{l}{2}\cos\theta, \tag{15}$$

potential energy of the point mass is

$$E_{P2} = Mgl\cos\theta, \tag{16}$$

potential energy of planar moving bar (free pendulum) is

$$E_{P3} = mg\left(l\cos\theta - \frac{l}{2}\cos\varphi\right). \tag{17}$$

Finally, substituting (15), (16) and (17) in (14) the total potential energy of the system is

$$E_P = mg\frac{l}{2}\cos\theta + Mgl\cos\theta + mg\left(l\cos\theta - \frac{l}{2}\cos\varphi\right). \tag{18}$$

The modelled system has two degrees of freedom. The general equations of motion in general form for two variables, θ and φ, are

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\dot\theta}\right) - \frac{\partial L}{\partial\theta} + \frac{\partial D}{\partial\dot\theta} = Q_\theta, \tag{19}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\dot\varphi}\right) - \frac{\partial L}{\partial\varphi} + \frac{\partial D}{\partial\dot\varphi} = Q_\varphi, \tag{20}$$

where the Lagrange function is

$$L = E_K - E_P, \tag{21}$$

the generalized forces are

$$Q_\theta = \frac{\partial W}{\partial\theta},\ Q_\varphi = \frac{\partial W}{\partial\varphi}, \tag{22}$$

and the dissipation function

$$D = 0. \tag{23}$$

The virtual work of the non-conservative time periodical torques is

$$\delta W = T_1\delta\theta + T_2\delta\varphi \tag{24}$$

where T_1 and T_2 are the torques which act on the bars. Substituting (13), (18) and (24) in (19) and (20), the differential equations become

$$\begin{aligned}&\left(\frac{4}{3}m + M\right)l^2\ddot\theta + \frac{1}{2}ml^2\cos(\theta+\varphi)\ddot\varphi - \frac{1}{2}ml^2\dot\varphi^2\sin(\theta+\varphi) - \\ &-gl\left(\frac{3}{2}m + M\right)\sin\theta = T_1\end{aligned}, \tag{25}$$

$$\frac{1}{2}ml^2\cos(\theta+\varphi)\ddot\theta + \frac{1}{3}ml^2\ddot\varphi - \frac{1}{2}ml^2\dot\theta^2\sin(\theta+\varphi) + mg\frac{l}{2}\sin\theta = T_2. \tag{26}$$

Equations (25) and (26) are the non-linear differential equations of motion for the model representing cow walk sketched in Figure 6.7.

The forces acting on the system during walking simulation of the model are presented in Figure 6.8.

Decoupling the model, the following system of equations in point C_1 in the direction of x and y axes marked in Figure 6.8 are formed

$$m\ddot{x}_{C1} = F_T + X_M \,, \tag{27}$$

$$m\ddot{y}_{C1} = F_N - mg - Mg + Y_M \,, \tag{28}$$

and in point C_2,

$$m\ddot{x}_{C2} = -X_M \,, \tag{29}$$

$$m\ddot{y}_{C2} = -mg - Y_M \,. \tag{30}$$

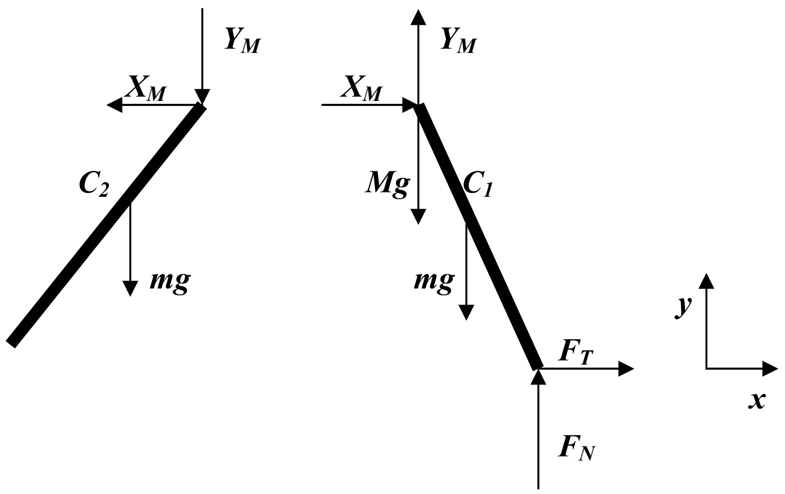


Figure 6.8: Forces and joins reaction of the model.

Reaction forces in point M are X_M and Y_M, while the reaction forces at the ground connection A are F_N and F_T. Substituting (29) in (27) and (30) in (28) reaction forces between bar and ground are

$$F_T = m(\ddot{x}_{C1} + \ddot{x}_{C2}), \tag{31}$$

$$F_N = m(\ddot{y}_{C1} + \ddot{y}_{C2}) + g(M + 2m), \tag{32}$$

where x_{C2} and y_{C2} are formulated in equations (5) and (6), respectively, and

$$x_{C1} = (l/2)\sin\theta \,, \tag{33}$$

$$y_{C1} = (l/2)\cos\theta \,. \tag{34}$$

The second derivative of (5), (6), (33) and (34) is substituted in (31) and (32), and the final equations of ground reaction forces are created

$$F_T = m\left(-\frac{3}{2}l\sin\theta\dot{\theta}^2 + \frac{3}{2}l\cos\theta\ddot{\theta} - \frac{1}{2}l\sin\varphi\dot{\varphi}^2 + \frac{1}{2}l\cos\varphi\ddot{\varphi}\right), \tag{35}$$

$$F_N = m\left(-\frac{3}{2}l\cos\theta\dot{\theta}^2 - \frac{3}{2}l\sin\theta\ddot{\theta} + \frac{1}{2}l\cos\varphi\dot{\varphi}^2 + \frac{1}{2}l\sin\varphi\ddot{\varphi}\right) + g(M + 2m). \tag{36}$$

The aim is to connect ground reaction vertical force F_N with equations of motion of the model (25) and (26). From coupled system of differential equations of motion the values of θ, $\dot{\theta}$, $\ddot{\theta}$, φ, $\dot{\varphi}$ and $\ddot{\varphi}$ can be calculated using numerical calculus method. The equation (26) is modified

$$\ddot{\varphi} = -\frac{3}{2}\cos(\theta + \varphi)\ddot{\theta} + \frac{3}{2}\sin(\theta + \varphi)\dot{\theta}^2 - \frac{3}{2}\frac{g}{l}\sin\theta + \frac{3}{ml^2}T_2, \tag{37}$$

and substituted in (25). The equation (25) is then expressed as

$$\ddot{\theta} = \frac{1}{\frac{4}{3}m + M - \frac{3}{4}m\cos^2(\theta + \varphi)}\left[-\frac{3}{4}\dot{\theta}^2 m\sin(\theta + \varphi)\cos(\theta + \varphi) + \frac{3}{4}\frac{g}{l}m\cos(\theta + \varphi)\sin\varphi + \frac{1}{2}m\sin(\theta + \varphi)\dot{\varphi}^2 + \frac{g}{l}\left(\frac{3}{2}m + M\right)\sin\theta + \frac{1}{l^2}T_1\right], \tag{38}$$

It is assumed for initial conditions that the angle position of the bars correspond to the ending position of the previous step,

$$\theta(t_{START}) = \varphi(t_{END}), \tag{39}$$

$$\varphi(t_{START}) = \theta(t_{END}). \tag{40}$$

As the kinetic energy is lost in inelastic impact of foot with the ground it is taken that the angular velocities of bars for the start position of each step are zero,

$$\dot{\theta}(0) = 0, \tag{41}$$

$$\dot{\varphi}(0) = 0. \tag{42}$$

The values of bar position angles, angular velocities and angular accelerations can be calculated from (37) and (38) for assumptions stated in (39) – (42). However, the values of point mass and bar masses, as well as, bar length have to be chosen. First, value of $\ddot{\theta}$ is calculated from (38) and then the value of $\ddot{\varphi}$ is calculated from (37). These values are integrated by numerical integration method and then the computed value of θ, $\dot{\theta}$, φ and $\dot{\varphi}$ are substituted in (38) and (37) in the new iteration. The new computed values of angular accelerations are obtained. The process continues dependent on the number of iteration in numerical calculus. At every iteration the value of ground reaction vertical force is calculated according to equation (36).

As exposed in mathematical model development, there are three parameters to constrain in the model: mass of point mass (*M*), masses of bars (*m*) and length of the bars (*l*). The total mass is the sum of point mass and two bars masses. Initially, it was chosen the mass of the bars to be 16 % of total mass. This was done according to known data on mass ratio of the cattle body parts. The length of the bars was fixed to one meter, since the average length of the legs of an adult cow is approximately one meter. The desired position angles of the bars are calculated as

$$\theta_d(t) = -A\cos(\omega t), \tag{43}$$

$$\varphi_d(t) = A\cos(\omega t). \tag{44}$$

where A is the maximal (minimal) angle between bars and vertical and ω denotes the frequency of oscillation. This constant was chosen to be 10 degrees according to average leg swing of cattle. According to (43) and (44) simulation starts at maximal extension of bars. The steps are set to be ideal, producing an ideal walk simulation. The controller chosen was a simple proportional-differential (PD). The parameters of the PD controller had to be adjusted such that the output force was smooth and close to realty. An example of cow walk model simulation is presented for better understanding of model and simulation.

The simulation was run with the use of two equation of motion (37) and (38), and the ground reaction vertical force equation (36). The desired bar position angles were of 10 degrees amplitude and 1 rad/s oscillation frequency. The total body mass was 700 kilograms. The parameters of the PD controller were set to 10000 and –5000 for proportional and differential part, respectively. The calculated vertical force during simulation is plotted in Figure 6.9. In Figure 6.9 the solid line marks the calculated ground reaction vertical force, while the dashed line marks the angle θ (angle values were increased by 700 to be visible with force curve). As shown in Figure 6.9, the force reaches its minimums when the bars are in the vertical position. The maximums are reached at the right most and left most position of bars. The frequency of force oscillation is double the frequency of position angles. This, however, coincide with reality ground reaction vertical force during walking (exp. human walk). The right most and left most positions are the switching of legs points (change of direction of model oscillation). At these points the most force is applied to the ground. The plot of force in Figure 6.9 is not a perfect harmonic. In the first step calculated force has slight variations compared to the following steps. This is caused by the fact that the system moves from the still position and later the constant inertia is created producing same amount of ground reaction for each step. The amplitude of 40 kilograms in force oscillation is obvious from the plot. The force oscillation amplitude changes as the total mass and/or bars position angle frequency change.

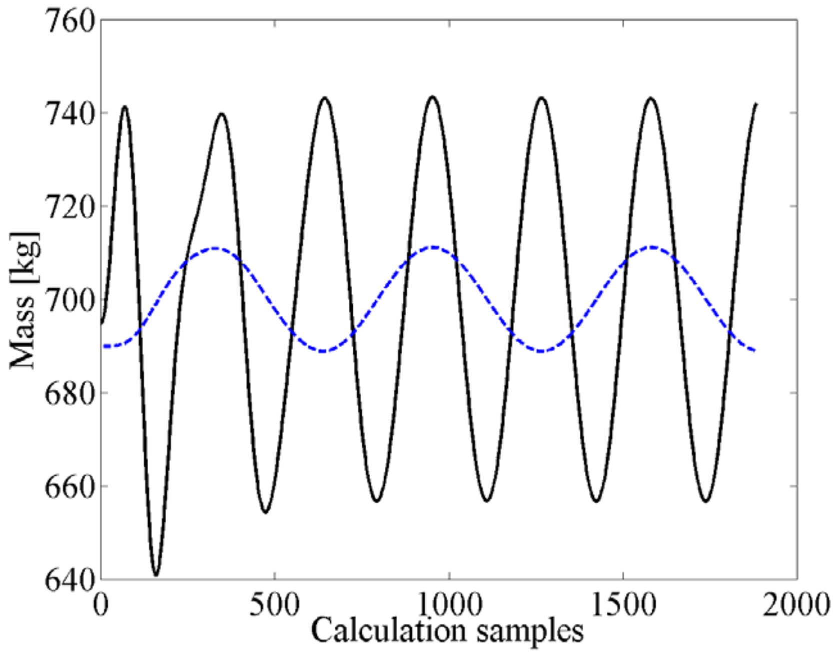


Figure 6.9: Plot of calculated force (700 kg, 1 rad/s).

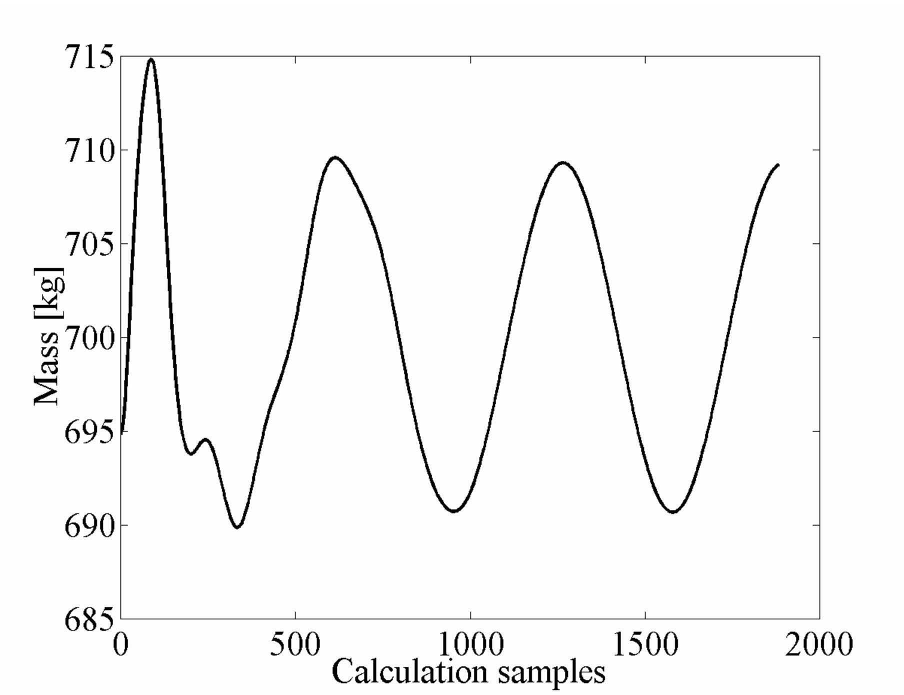


Figure 6.10: Plot of calculated force (700 kg, 0.5 rad/s).

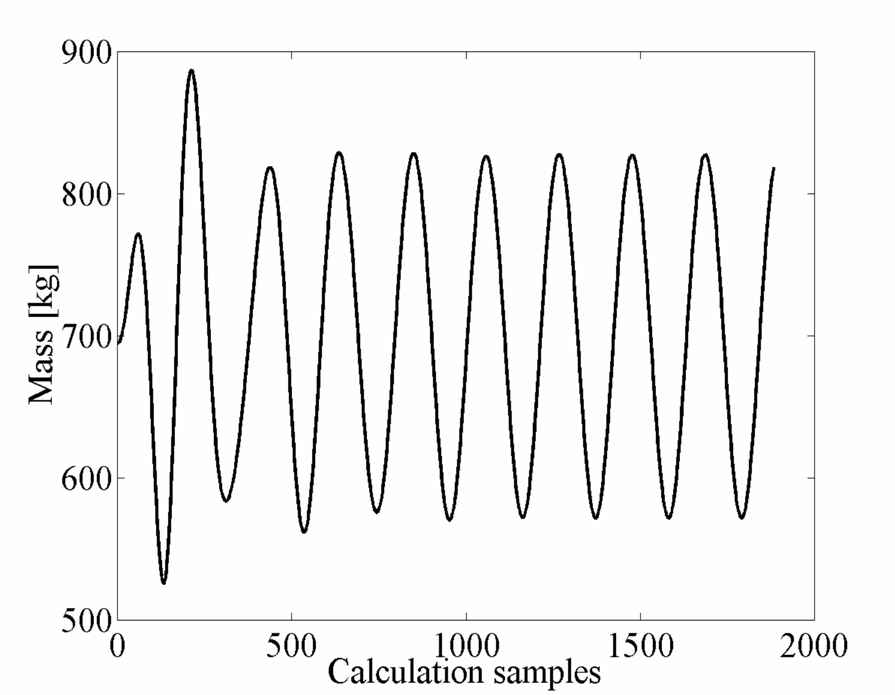


Figure 6.11: Plot of calculated force (700 kg, 1.5 rad/s).

In Figures 6.10 and 6.11 calculated force is plotted for two frequencies. The force calculated and plotted in Figure 6.10 was computed for the total mass of 700 kilograms and angle oscillation frequency of 0.5 rad/s. This is the case of slower walking than in the first case. The amplitude of vibration is approximately 10 kilograms. In Figure 6.11 faster walking simulation results are presented. The angle oscillation frequency was 1.5 rad/s and therefore produced the vibration amplitude of about 130 kilograms. From the previously exposed it is obvious that the acceptable bar oscillation frequency range between 0 and 1.5 rad/s. The value of bars oscillation frequency is directly proportional to walking velocity.

According to above presented the model used in simulation had two inputs and one output. The inputs in the system were total mass and bars oscillation frequency analogue to body weight of the animal and the walking velocity. The output was the ground reaction vertical force produced analogue to force between feet and weighing platform.

6.1.2 Model application

The process of body weight recognition for a cow passing over the walk-through weighing scale is done similar to earlier explained model verification method. The ground reaction force is measured and filtered. The valuable part of the signal is analysed. The developed mathematical model is used inversely for finding the correct body mass from known output force. The optimisation of two parameters is performed in order to match the measured force signal. The two parameters are the total mass and the bars oscillation frequency. The selected value of total mass from the optimisation process represents the body weight of the animal measured on the walk-through weighing scale.

An algorithm was created to extract the average values of maximums and minimums of force signal and the distance between peaks. Based on the information on average maximum and minimum of the force signal the parameters in the mathematical model are adjusted.

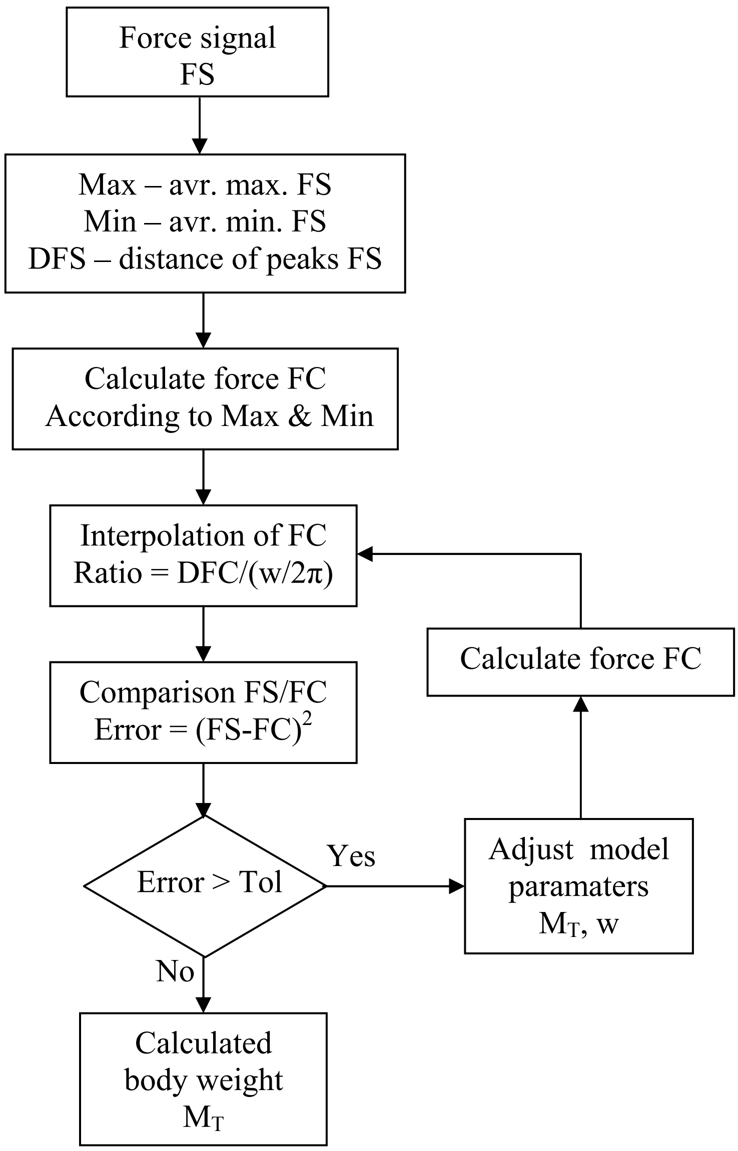


Figure 6.12: Block diagram of the optimisation algorithm.

After first round of model parameters selection the so calculated force was compared to measured force. In order to match the peaks of the two curves the information on distance between peaks is used. The divergence of the two curves is computed as square error. The part of the force signal where the error is smallest is cut out for further comparison. The procedure repeats for new parameter values until the best match is achieved. The block diagram of the body weight recognition algorithm is shown in Figure 6.12.

6.2 Fuzzy logic method

As presented in previous chapter, mathematical model of cow movement might be applied in body weight recognition of cattle weighted on the walk-through weighing scale. Other ways of solving the problem of dynamic body weight measurement were analysed. Fuzzy logic based method of body weight recognition for cattle weighted on the walk-through weighing scale is exposed.

6.2.1 Model creation

The approximation capability of fuzzy logic applied for force signal measured during cow walk is analysed. Takagi-Sugeno fuzzy system was created. Takagi-Sugeno fuzzy model was chosen since its representative power. It is capable of describing a non-linear system using sufficient rules and training data. The model consists of: Gaussian fuzzifier, product inference engine and centre average defuzzifier. Gaussian membership function (MF) is used to denote the grade of the input for particular membership class

$$\mu_{Ai} = \exp\left(-\frac{(x - c_{Ai})^2}{a_{Ai}^2}\right), \tag{45}$$

$$\mu_{Bi} = \exp\left(-\frac{(y - c_{Bi})^2}{a_{Bi}^2}\right), \tag{46}$$

where x and y are the inputs and c_{Ai} and c_{Bi} are the parameters of the Gaussian function characterising centre and a_{Ai} and a_{Bi} are the parameters of the Gaussian function characterising width of the function. Product inference engine performs multiplication of the grades combined according to Takagi-Sugeno fuzzy rules

$$w_i = \mu_{Ai} \times \mu_{Bi} . \tag{47}$$

The consequent of the Takagi-Sugeno fuzzy rules are described by linear function

$$F_i = p_i x + q_i y + r_i , \quad (48)$$

where p_i, q_i and r_i are the parameters of the Takagi-Sugeno fuzzy rules consequent.
Centre average defuzzification calculates the output of the fuzzy system

$$y = \frac{\sum w_i F_i}{\sum w_i} . \quad (49)$$

The adjustable parameters of the system are c_{Ai}, c_{Bi}, a_{Ai}, a_{Bi}, p_i, q_i and r_i.
The measured force signal is the output of a non-linear function. This non-linear function has a number of inputs. Among them are the body length, length of legs, body weight, body velocity, velocity of legs and position angle of legs. The only input that is possible to be measured during weighing scale crossing is the walking velocity. It is calculated from the measured time of crossing the scale and the length of the scale. However, the walking velocity is assumed to be constant during the whole stride period. The function is therefore simplified to a single input single output (SISO) function. The input is the time of crossing the scale, while the output is the vertical force created by animal feet during walking. The fuzzy model to be created to approximate such a function has one input and one output. Takagi-Sugeno fuzzy model for one input uses Gaussian membership function (45). Product inference engine is not used since only one input is applied and the fuzzy rules are simple. The number of membership functions that cover the input region define the number of consequent parts in fuzzy rules. Each membership function corresponds to a consequent. The Takagi-Sugeno fuzzy rule consequent for single input system is

$$F_i = p_i x + r_i . \quad (50)$$

Since the single input system is used equation (47) is simplified in

$$w_i = \mu_{Ai} . \quad (51)$$

The number of adjustable parameters is

$$N_{PAR} = 4 N_{MF} , \quad (52)$$

where N_{MF} is the number of membership functions.

The parameters of the fuzzy model to approximate force signal measured with the walk-through weighing device on dairy farm can be adjusted with a number of training methods proposed in literature. One of the most powerful training tools is the so-called hybrid training in adaptive network based fuzzy inference system (ANFIS). It combines gradient descent method and least square estimation method to fit the parameters of the fuzzy system so the square error of force signal and approximation curve are minimised. ANFIS model for single input single output system is drawn in Figure 6.13. In Figure 6.13 neuro-fuzzy model with two membership functions is shown. X stands for the crossing time, A_i is the membership function, P stands for the product inference engine, while N is the normalised part and F_i is the Takagi-Sugeno fuzzy rule consequent. The model has five layers. The first layer performs fuzzification with the Gaussian bell shaped function (45).

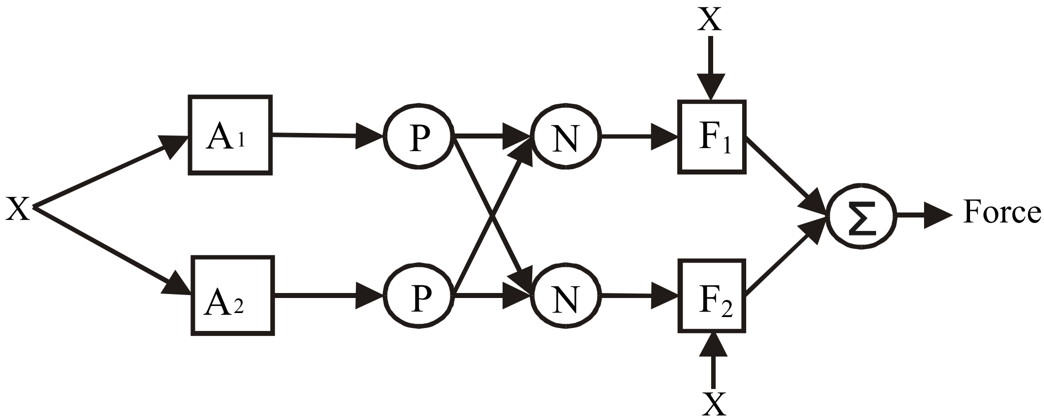


Figure 6.13: ANFIS SISO model with two MFs.

Second and third layers are the product inference engine and the normalised part, respectively. As mentioned, the product inference engine is not used for single input system. The normalised part computes normalised fuzzy value for each input following (51)

$$\overline{w}_i = \frac{w_i}{\sum w_i}. \qquad (53)$$

The normalised layer performs a part of calculation for centre average defuzzification.The fourth layer is the Takagi-Sugeno rules consequent described with equation (50). The last layer is the summation. For the known input-output signal pair, i.e., time of crossing the scale and vertical force acting on the scale during walking, the system parameters are calculated.

To illustrate the possibility of the created fuzzy model in function approximation an example is considered. Approximation of a force signal curve produced by 618 kilograms cow and time of crossing 1.68 seconds is analysed. In Figure 6.14 the filtered valuable part of the force signal is marked with crosses. The solid line is the fuzzy logic based approximation of the force curve. The fuzzy model included ten membership functions.

Consequently, forty parameters (52) were adjusted with hybrid training for input-output pairs of measured sampling time and force measured at that time. The approximation force function with ten membership functions (MFs) is highly precise. On the other hand, the input into the fuzzy system is not adequate to create a useful fuzzy system that can be used as a model for any situation of pace. The adequate input into the system would be the motion of each of four legs. Unfortunately, angles and velocities of the legs are impossible to measure on dairy cows. Yet, it is possible to measure these values but the system would be too expensive for common use on dairy farms. Therefore, the only left option was the crossing time – force, input – output model.

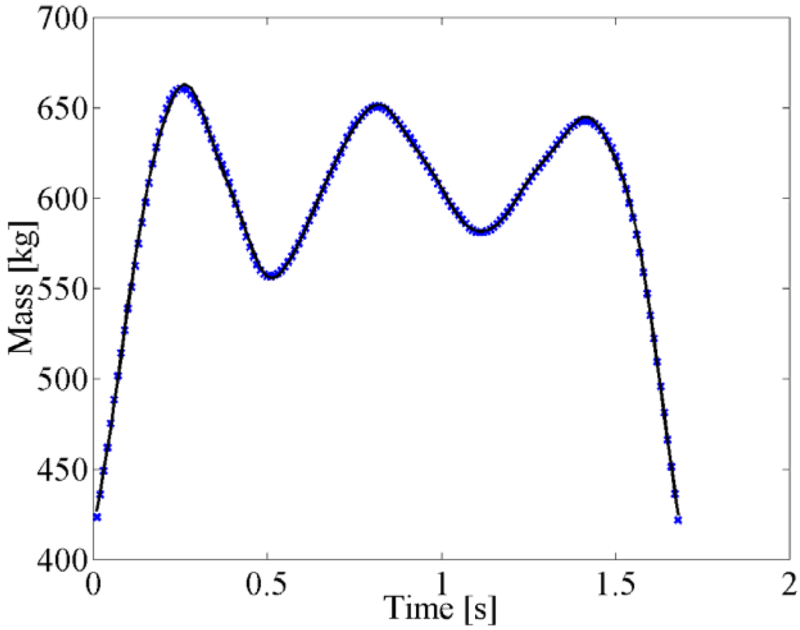


Figure 6.14: Force signal approximation with fuzzy model (10 MFs).

The plot of fuzzy model with two membership functions approximating the force signal is shown in Figure 6.15. The fuzzy model approximation marked with solid line shows an imprecise approximation of force curve. The fuzzy model with two membership functions performs so-called non-linear averaging of the force signal. Comparing the obtained solutions of force signal approximation it is concluded that the lower order fuzzy model is more desirable from our standpoint since the goal was to obtain the average value of measured force excluding extra influences. The extra influences are the length of the step, body swinging, hobble, etc. Namely, the model should only model body weight and time dependencies. For a cow with 618 kilograms a number of walk-through body weight measurements for various walking velocities were performed. The measured times of scale crossing were: 1.43, 1.52, 1.67, 1.82, 1.98 and 2.03 seconds. The fuzzy model with two membership functions was applied in function approximation for the measured force signal. Before approximation, the force signal was processed and filtered as explained in Chapter 5. The created curves are plotted in Figure 6.16.

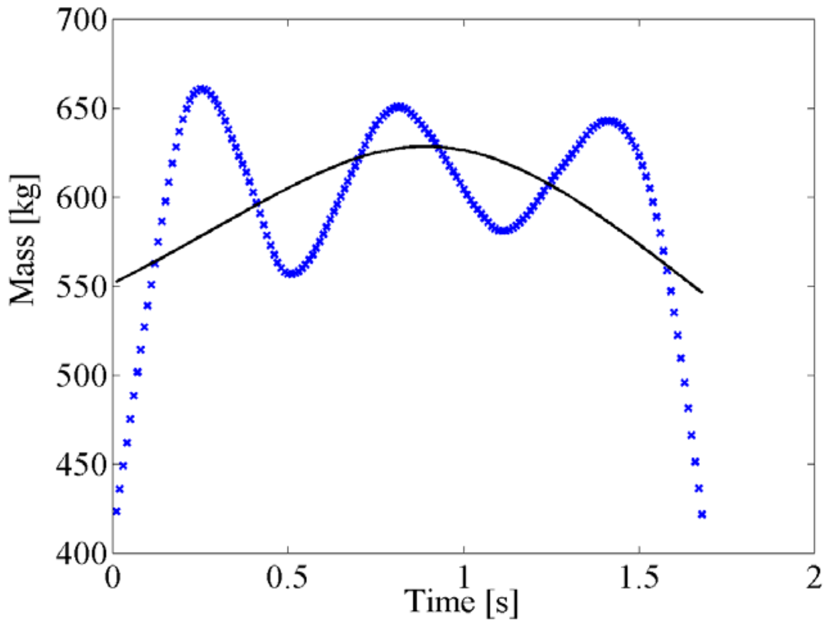


Figure 6.15: Force signal approximation with fuzzy model (2 MFs).

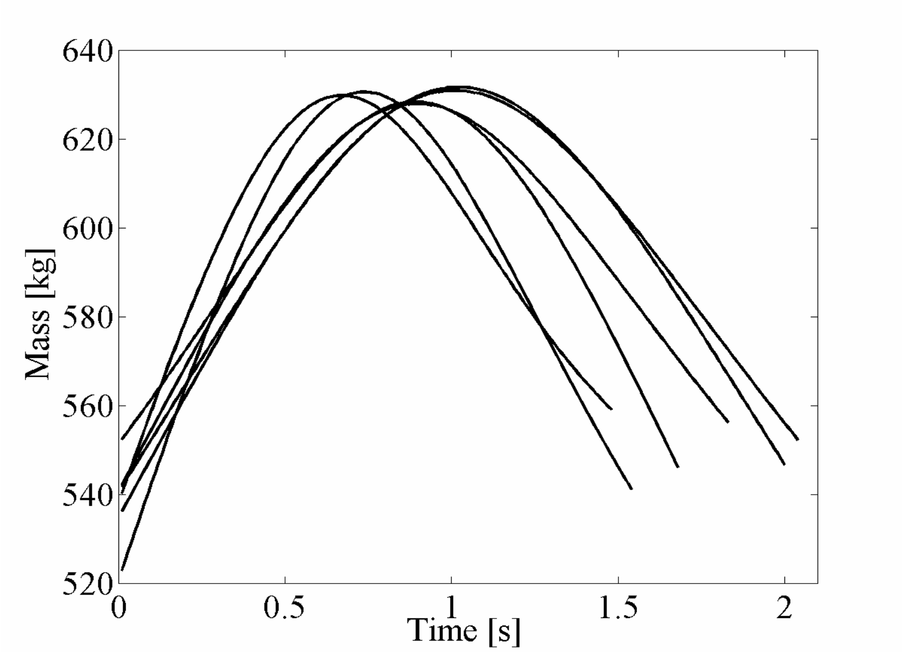


Figure 6.16: Approximation curves of force signals for 618 kg cow (2 MFs).

The curves shown in Figure 6.16 are similar to each other. They differ in the width, which is produced by different crossing times. If centred and adjusted to the same number on the horizontal axis they would create a close match. Consequently, the curves created with fuzzy approximation of force signal are dependent on body weight and time of crossing the scale.

It was of great importance to investigate the approximation capabilities of the fuzzy model with two membership functions not only for same body weight and various walking velocities but also for different body weights and similar crossing times. Experiments were performed on three dairy cows with different body weights. The animals tested were with 558, 618 and 700 kilograms body mass. After a number of measurements the close time of crossing the scale were 1.81, 1.83 and 1.85 seconds, from lightest to heaviest animal. The filtered force signals for the three crossing cases are shown in Figure 6.17. The fuzzy model based approximation of force signals plotted in Figure 6.17 is plotted in Figure 6.18. As shown in Figures 6.17 and 6.18, the force signals differ among each other, while fuzzy approximated curves are alike. The similarity among approximated force signals enabled the possibility of representing various situations of stride for the same animal for similar crossing times with one fuzzy model (Figure 6.19). In Figure 6.19 approximated force signals for a cow with 700 kilograms body mass and 1.85 and 1.88 seconds crossing times are plotted.

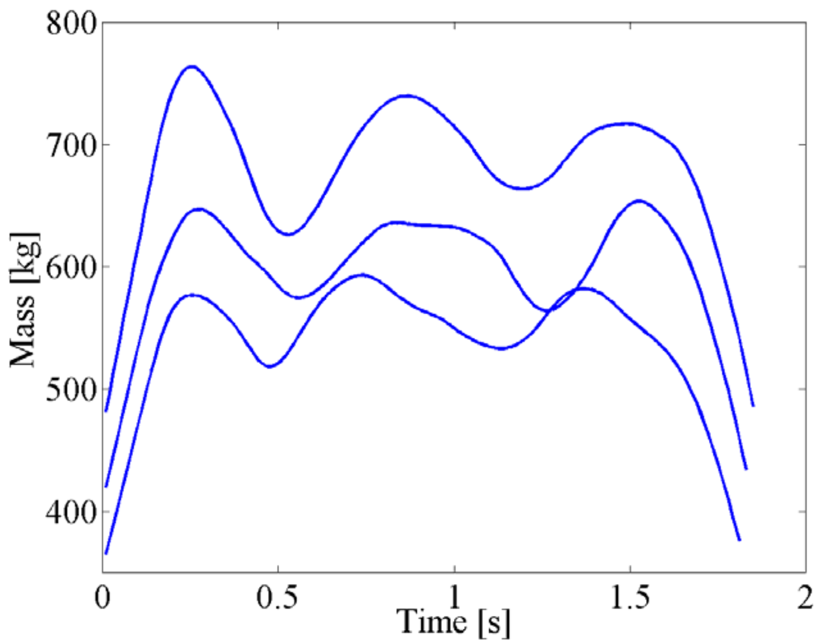


Figure 6.17: Filtered force signals of 558, 618 and 700 kg cows.

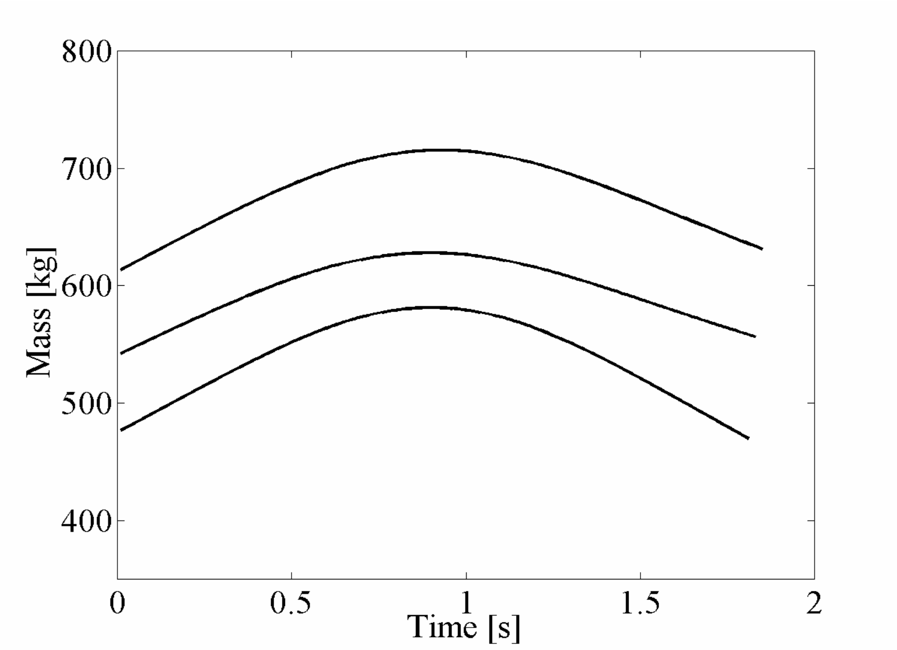


Figure 6.18: Approximation curves of force signals for 558, 618 and 700 kg cows (2 MFs).

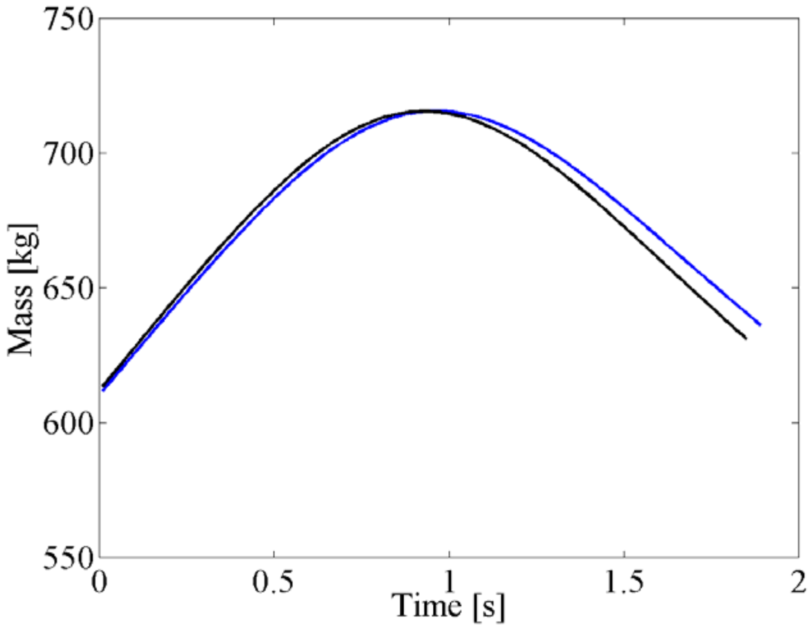


Figure 6.19: Approximation curves of force signals for 700 kg cow (1.85; 1.88 s).

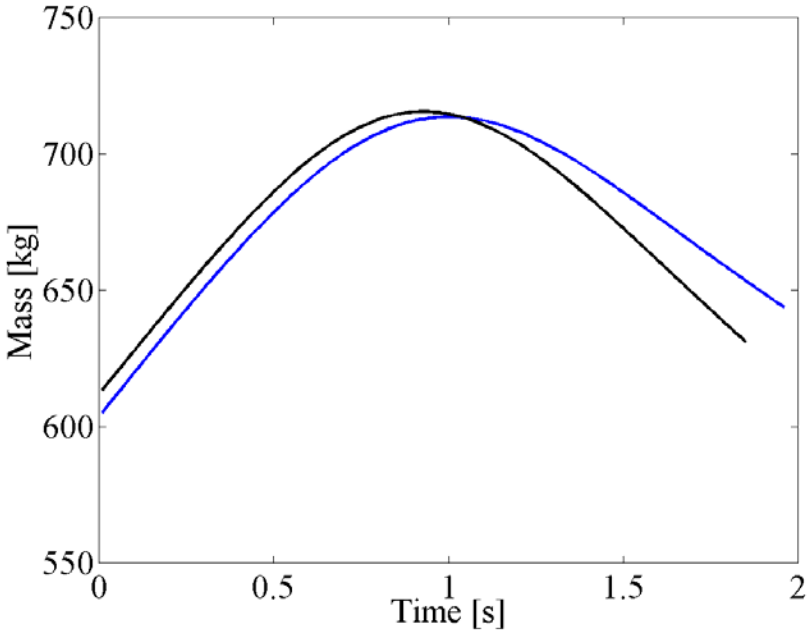


Figure 6.20: Approximation curves of force signals for 700 kg cow (1.85; 1.97 s).

The curves in Figure 6.19 match while curves in Figure 6.20 are shifted. The differences between times of crossing in the two figures are 0.03 and 0.12 seconds, respectively. Although the time differences are quite small one fuzzy model could not represent stride differing more than 0.1 seconds from that the model was created. The calculated parameters of the fuzzy model approximating force signal did not have any physical meaning. Consequently, fuzzy models had to be created for groups of crossing times, i.e. 1.09, 1.15, 1.31, 1.54, etc. To cover most of crossing times a number of experiments were done. The cattle varied in their body weights and age, since the younger and lighter animals were quicker than older and heavier. The models created were gathered in a database of models. In the database of models, parameters of the fuzzy model were stored. Two membership functions Takagi-Sugeno fuzzy model contains eight adjustable parameters ($a_1, a_2, c_1, c_2, p_1, p_2, r_1, r_2$). From a variety of processed force signals for fuzzy model detection, some parameters of the models were possible to constrain. The parameters in the first layer (a_1, a_2, c_1, c_2) were shown to be linearly dependent on crossing time T_C

$c_1 = 0$, $c_2 = T_C$,

$a_1 = 0.6T_C$, $a_2 = 0.6T_C$.

Therefore, the parameters of the model that needed to be calculated were only in the last layer of the model. Four parameters (p_1, p_2, r_1, r_2) of the fuzzy model were representing every crossing situation of the weighing scale. Since only parameters in the Takagi-Sugeno fuzzy rules consequent were computed, the hybrid training was simplified by least squares estimation. This simplification of approximation procedure decreased the model parameters computation time. The database of model parameters contained the values of four parameters (p_1, p_2, r_1, r_2). Database of model parameters was created with body masses 558, 618 and 723 kilograms as shown in Table 6.1.

Table 6.1: Database of model parameters.

Crossing time (s)	1.09	1.15	1.31	1.54	1.63	1.74	1.81	2.00	2.10
Body weight (kg)	558	723	723	618	618	558	558	618	618
p_1	0.377	8.182	3.854	1.601	2.433	0.746	2.740	1.740	2.592
r_1	520.7	629.0	657.1	626.7	576.3	525.5	482.3	565.9	549.9
p_2	-0.11	7.208	1.792	2.248	1.721	-0.47	1.123	0.999	1.714
r_2	578.6	49.48	461.3	326.0	352.0	588.8	327.3	411.9	268.1

As presented in Table 6.1, crossing times covered range between 1.00 and 2.20 seconds. The so created database of models was used in body weight recognition of cattle measured on walk-through weighing scale.

6.2.2 Model application

The usage of created database of models and fuzzy function approximation in body weight recognition problem is here explained. The body weight recognition of cattle measured on the walk-through weighing scale was based on fuzzy model approximation of force signal measured on the scale and filtered with the signal processing algorithm. The Takagi-Sugeno fuzzy model with two membership functions was used in signal approximation. The output of the fuzzy model is

$$y = \overline{w}_1 F_1 + \overline{w}_2 F_2 , \tag{54}$$

where

$$\overline{w}_1 = \frac{w_1}{w_1 + w_2}, \tag{55}$$

$$\overline{w}_2 = \frac{w_2}{w_1 + w_2}, \tag{56}$$

$$F_1 = p_1 x + r_1, \tag{57}$$

$$F_2 = p_2 x + r_2. \tag{58}$$

The elements in equation (55) and (56) are calculated as

$$w_1 = \exp\left(-\frac{(x - c_1)^2}{a_1^2}\right), \tag{59}$$

$$w_2 = \exp\left(-\frac{(x - c_2)^2}{a_2^2}\right). \tag{60}$$

Switching (57) and (58) in (54) the output equation is formed

$$y = \overline{w}_1 p_1 x + \overline{w}_1 r_1 + \overline{w}_2 p_2 x + \overline{w}_2 r_2. \tag{61}$$

Equation (61) can be written in matrix form

$$y = X^T P. \tag{62}$$

The matrices are of form

$$X = \begin{bmatrix} \overline{w}_1 x \\ \overline{w}_1 \\ \overline{w}_2 x \\ \overline{w}_2 \end{bmatrix}, \tag{63}$$

$$P = \begin{bmatrix} p_1 \\ r_1 \\ p_2 \\ r_2 \end{bmatrix}. \tag{64}$$

Parameters a_1, a_2, c_1 and c_2 are fixed as explained earlier. Parameters grouped in the matrix P were adjusted with least square estimation method. Least square training method used following equations

$$P = P + SX\left(y - X^T P\right), \tag{65}$$

where S was calculated as

$$S = S - \frac{SXX^T S}{1 + X^T SX}. \tag{66}$$

Initial value of S was chosen to be

$$S_{INIT} = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}. \qquad (67)$$

The approximator produced a fuzzy model with output dependent on animal body weight and walking velocity. Plotted fuzzy model output created a curve which was similar to those plotted for similar walking velocities. Based on similarity on output curves body weight recognition method was created.

Body weight recognition method consisted of time and surface comparison. For every new object crossing over the scale, the force vs. time was approximated using the described fuzzy method creating a new curve. The parameters of the fuzzy model were selected from the initially created database of model parameters according to measured time of crossing. The new curve and the database curve were compared. The difference between the integrals of the two curves denoted the new body weight of the measured animal.

The procedure of body weight estimation is explained on an example. The force vs. time measurement was filtered and the valuable part of the signal was selected. The time animal stayed on the weighing platform with all four legs was 1.85 seconds. Applying the fuzzy approximation for the measured force signal a fuzzy output was created and solid line curve was plotted in Figure 6.21. Crosses in Figure 6.21 marks the force signal.

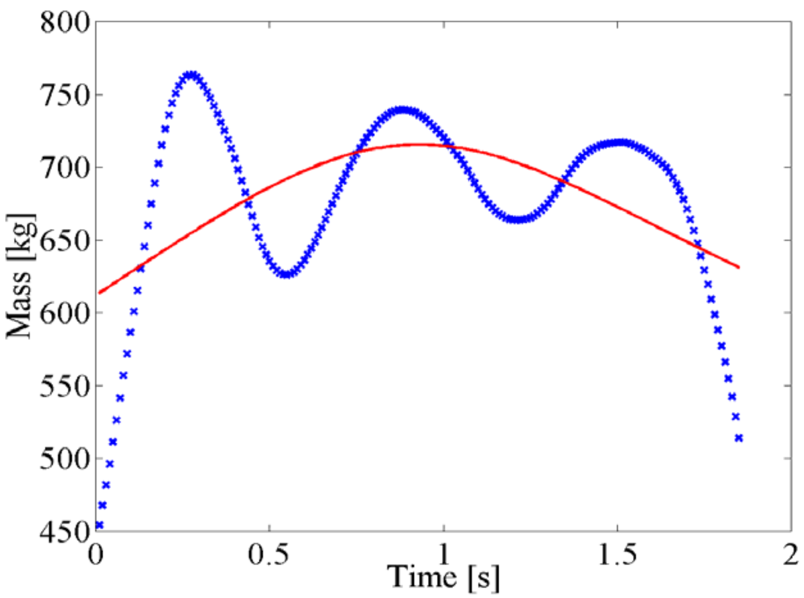


Figure 6.21: Force signal approximation with fuzzy model.

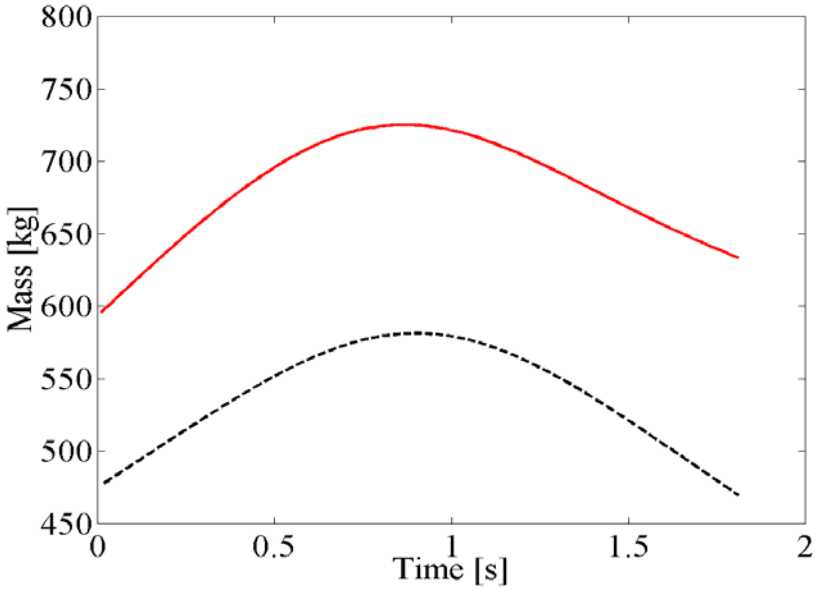


Figure 6.22: Fuzzy approximation curve of new (solid) and database (dashed) force signal.

The parameters of the model from database were chosen according to the time of crossing the scale. As presented in Table 6.1, the closest time of crossing from the database is 1.81 seconds for body mass 558 kilograms. The output of the fuzzy model for the chosen database model parameters is plotted in Figure 6.22 with dashed line. The solid line in Figure 6.22 presents fuzzy approximation of the new measured force vs. time.

To find the body weight of the new animal the curves plotted in Figure 6.22 were compared. The surface between the solid and the dashed line was computed. The surface above solid line was positive, while under the solid line was negative. With such signs total intersection surface was calculated. The database fuzzy model output values were increased or decreased in order to minimise the total intersection surface. This was obtained with an optimisation algorithm. Once the minimum intersection surface was reached the added value to the database fuzzy output was added to the database body weight for which the parameters were selected. In this example, the best fit was achieved adding 143 kilograms to the database fuzzy output. Figure 6.23 shows the best fit of the new and the database curves. Adding 143 kilograms to 558 kilograms, body weight of new animal was determined. Statically measured body mass of the new animal was 700 kilograms. The calculated mass was 1 kilogram bigger than the measured one.

The above procedure for body weight estimation is applicable not only for single crossing case, but also for the crowded case when animals follow each other in quick succession over the scale. The valuable parts of the crowded crossing case force signal are filtered and the body weight recognition method is applied. An example of two cows following each other over the weighing scale is discussed. The animals were statically weighted, such that the first was 567 and the second was 633 kilograms. Filtered force signal is presented in Figure 6.24.

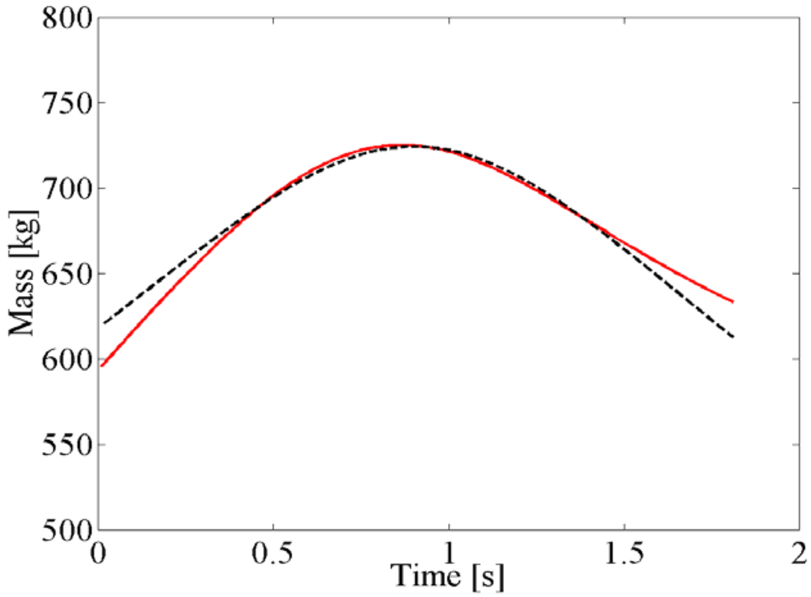


Figure 6.23: Best fit of modelled curves.

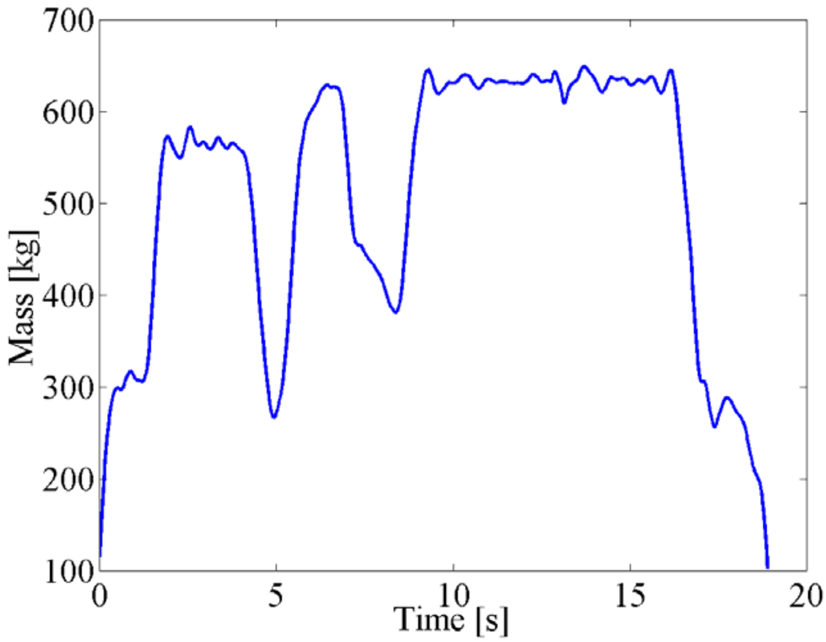


Figure 6.24: Filtered force signal (567 & 633 kg).

The valuable parts of the signal were selected as explained in Chapter 5. After 5.25 seconds the first cow stepped on the scale the second moved forward. Valuable parts of the signal were between 1.47 - 4.45 seconds and 8.62 - 16.79 seconds. The selected parts of the force signal were approximated with presented fuzzy model and the body weight recognition procedure was applied. However, the database comparison model was chosen according to the time of crossing the platform. In other words, the time between step on and off the scale of the animal was of interest for database match up, but the comparisons of the created curves were done for the part of the signal when the animal was alone on the scale. The calculated body weights were 570 and 623 kilograms.

The proposed method for denoting animal body weight from measured force signal is tested for various crossing situations of the weigher and the results are presented in Chapter 7.

6.3 Neural networks method

The method includes neural networks calculations. In the first part of the chapter the selection of peaks in force signal is explained. The training rules of the network for input-output pairs are also described.

6.3.1 Model creation

The dynamic weighing problem can be turned into a classification task. Classification in common sense is to link input vector with some class. The classification task in dynamic weighing associate input vectors with specific output vectors. In dynamic weighing the output vector is well known. It is the body weight of the animal measured on the manual (static) weighing scale. However, the parameters of the force signal in input vector to be associated to body weight need to be determined.

The measured force signal depends on walking velocity and body weight. The walking velocity and the body weight influence the amplitudes of force produced by feet. The amplitude of force increases as the walking speed increases. The amplitude of force also increases as body weight increases for constant walking speed. To illustrate that a cow with 558 kilograms body weight was measured on the walk-through weighing scale.

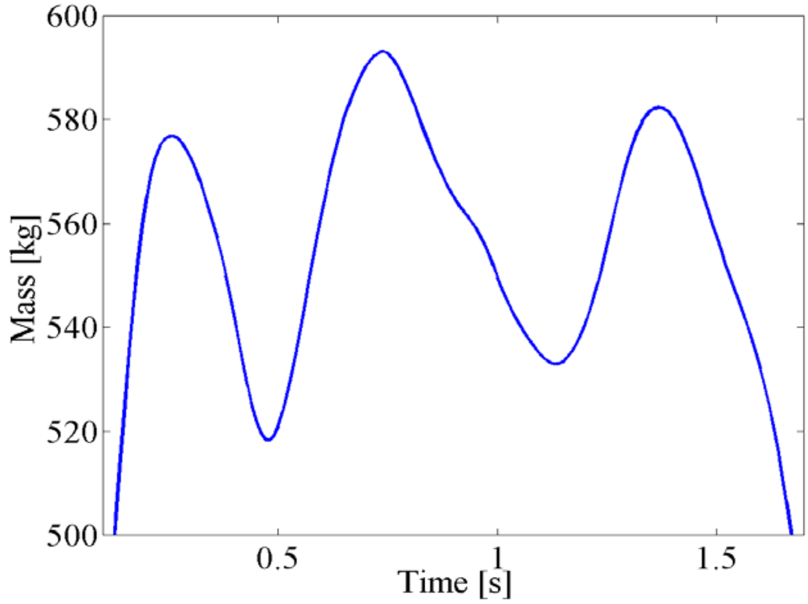


Figure 6.25: Measured and filtered force signal of 558 kg cow.

In Figure 6.25 and 6.26 the filtered force signal valuable parts are plotted. The crossing time of the weighing scale were 1.65 and 0.85 seconds, respectively. The maximum force values measured for the first and second case were 590 and 610 kilograms, while the minimums were 520 and 500 kilograms, respectively. In other words, the forces measured and plotted in Figure 6.25 and 6.26 had the amplitude of 35 and 55 kilograms. As expected, the amplitude of force oscillation was bigger for faster crossing of the scale.

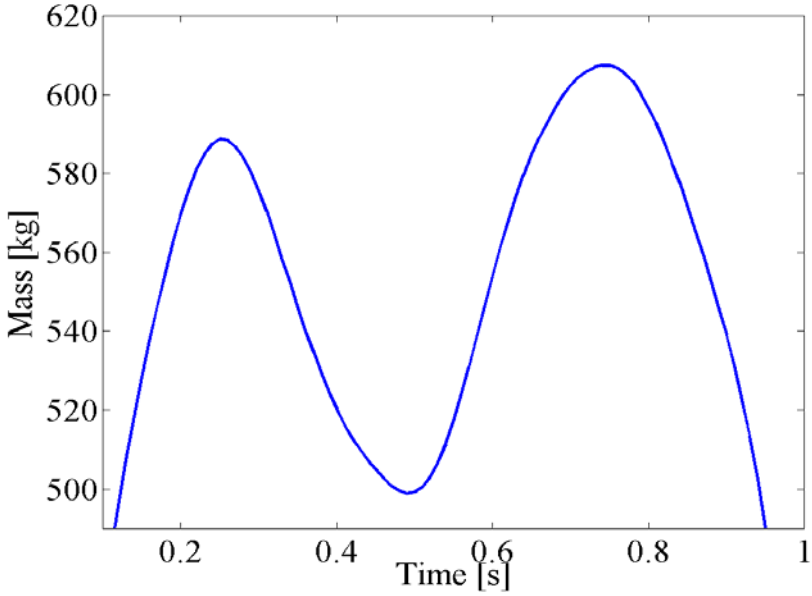


Figure 6.26: Measured and filtered force signal of 558 kg cow.

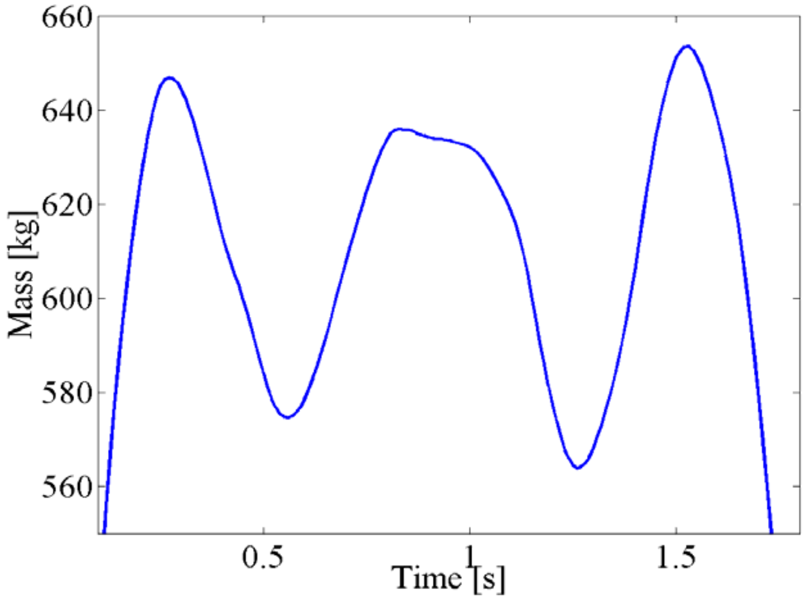


Figure 6.27: Measured and filtered force signal of 618 kg cow.

In Figure 6.27 measured force signal of 618 kilograms body weight cow is plotted. The crossing of the weighing platform lasted 1.67 seconds. This was similar to the crossing time of 558 kilograms cow plotted in Figure 6.25. The maximum force value measured for 618 kilograms cow was 650 kilograms, while minimum was 570 kilograms. This produced the force signal oscillation amplitude of 40 kilograms. Comparing the characteristics of the first and third force signal, it was noticed that the amplitudes differed for 5 kilograms, while the body weights differed for 60 kilograms and the walking velocity was approximately equal.

This leads to the conclusion that each crossing of the walk-through weighing scale produce force signal which can be characterized by maximum and minimum value of the force swinging. These values are directly dependent on animal's body weight and walking velocity. Consequently, the maximum and minimum of force oscillation can represent every measured force signal recorded on the walk-through weigher. Therefore, the two parameters of force signal can be selected as input vector for classification task.

Since the input-output vector was denoted, it was necessary to define the neural network used for classification task. There are a number of networks structure suggested in literature that are good classifiers. Two major groups are the linear and non-linear classifiers. In the case of inputs defined as maximum and minimum of force signal and output as body weight, there is no strict linear relation between input and output vector and so the non-linear classifier is preferred. The two well-known non-linear classifiers are radial basis network and multi-layer perceptron network. They contain hidden layers with non-linear activation function and an output layer with linear activation function. Radial basis network requires more neurons than standard multi-layer perceptron network. Probabilistic neural network is the variation of radial basis network commonly used in classification problems. Probabilistic network has a competitive output layer instead of a linear layer. The parameters in the probabilistic neural network increase exponentially with dimensions of input space. The probabilistic network approximates locally, while multi-layer perceptron network does it globally. However, multi-layer perceptron network performs successful classification of only smooth non-linearities. Since in the body weight recognition of dynamically weighted animals the training pairs of a classifier could not be introduced covering the whole surface of possibilities, the generalization need to be introduced in the network. Yet, the probabilistic neural network is formed for input-output pairs without typical training but with adjustment of the network parameters to the input-output vector, which results in limited classification possibilities. Since in the body weight recognition task a small number of training sets can be introduced to the network the probabilistic neural network should be avoided. Consequently, the multi-layer perceptron network would be the first choice of the neural network structure adequate for body weight classification. The structure of multi-layer perceptron network with one hidden layer is shown in Figure 6.28.

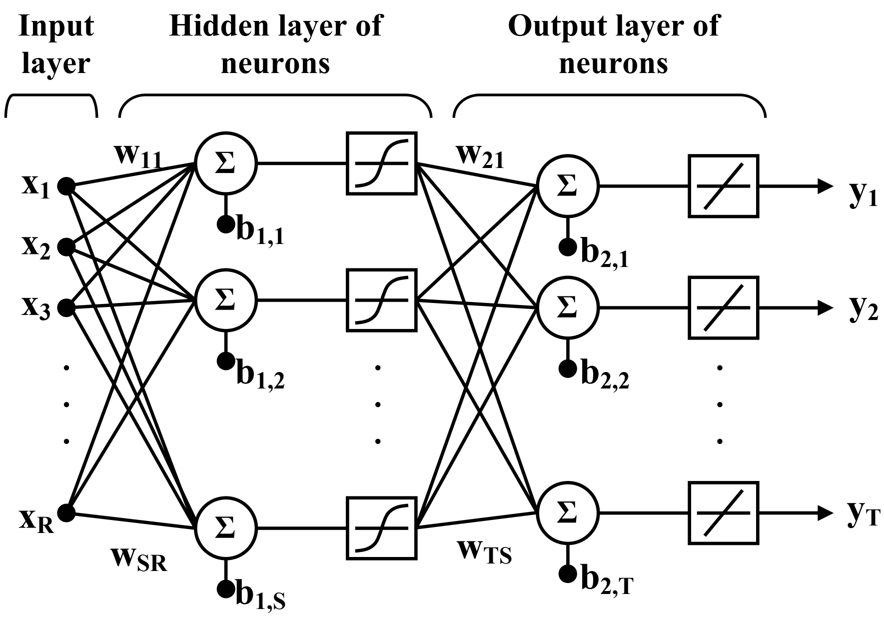


Figure 6.28: Multi-layer preceptron network structure with one hidden layer.

The final structure of the neural network used in body weight classification was determined after the network was tested. The training rule was first selected for this type of classification task.

One of the problems that occur during neural networks training is over fitting. The training data set error is minimized during training, but for the new data set error is large. The network is fully adapted to training data and leaves no space for new data. To eliminate the problem of over fitting the network needs to be learned to generalize to new situations. To improve network generalization the network has to be just large enough for an adequate fit. Larger networks result in over fitting, while smaller result in under fitting. The optimal network size is impossible to know without testing. However, there are methods developed to perform generalization on neural network. Bayesian regularization is one of the methods to accomplish neural network generalization. Bayesian regularization takes place within some training technique. Training methods for multi-layer perceptron network are mostly developed based on Widrow-Hoff algorithm, which is based on mean square error. The training mode needed in body weight recognition task was the batch mode, which updated the parameters of the network after all training sets were introduced. The batch training mode was important since it improved the generalization of the network. The recommended training method for small and medium size neural networks is Levenberg-Marquardt training, since its rapid convergence. On the other hand, Levenberg-Marquardt training can be adapted to perform Bayesian regularization. Levenberg-Marquardt training uses numerical optimisation technique and large memory storage. However, the suggested training method is not suitable for large networks. For training of large neural networks commonly used methods are conjugate descent and some heuristic technique training methods. The most powerful conjugate descent training for a variety of tasks is scaled conjugate gradient algorithm. This algorithm is designed to avoid the time-consuming line search per learning iteration, which makes the algorithm faster than other second order algorithms. Among the heuristic training methods resilient backpropagation represents simple batch mode training with fast convergence and minimal storage requirements. The resilient backpropagation training eliminates the effect of small gradient values and so decreases the training time of standard steepest descent training. Since the architecture of the multi-layer perceptron network used for body weight recognition task was not known various models and training methods were tested in order to find the optimal network structure and values of network parameters (weights and biases).

First, a multi-layer perceptron network was selected among various layer and neuron numbers with the use of Levenberg-Marquardt training. The training algorithm included Bayesian regularization. The Levenberg-Marquardt training method was developed from Newton's method. The basic Newton's method computed the Hessian matrix, which was the second derivative of performance function at the particular weights and biases. Performance function is commonly mean square error. Due to complexity of the calculation proposed by Newton, Levenberg-Marquardt offered an alternative without having to compute the Hessian matrix. The Jacobian matrix was computed instead Hessian matrix as a first derivative of performance function with respect to weight and bias variables

$$J = \left[\frac{\partial F}{\partial x_i} \right] \tag{68}$$

where F is the performance function (mean square error)

$$F = \frac{1}{N} \sum_{i=1}^{N} e_i^2 \tag{69}$$

and x is the vector of weights and biases.

The Hessian matrix can be approximated as

$$H = J^T J . \tag{70}$$

The Newton's method learning step is

$$x_{K+1} = x_K - H_K^{-1} g_K . \tag{71}$$

The gradient can be approximated as

$$g = J^T e , \tag{72}$$

where e represents vector of network errors. Therefore, the final Levenberg-Marquardt learning step is of form

$$x_{K+1} = x_K - \left[J^T J + \mu I \right]^{-1} J^T e , \tag{73}$$

with μ being a scalar determining the speed of the convergence. When μ is zero the Levenberg-Marquardt training method turns into the Newton's method. The Levenberg-Marquardt training method becomes gradient descent method when μ is large. The aim is to decrease μ since the Newton's method is faster and more accurate near error minimum than gradient descent. On the other hand, gradient descent method with small step size converge to error minimum slower but with continuous reduction of performance function. Consequently, the scalar μ is decreased after each step when the performance function decreases and increased only when the performance function increases. This leads to fast convergence with constant reduction of the performance function. However, the requirements of the storage memory and computation are big.

The Bayesian regularization is implemented in Levenberg-Marquardt training method by changing the performance function as

$$F = \beta E_D + \alpha E_W \text{ ,} \tag{74}$$

where E_D is the mean square error

$$E_D = \frac{1}{N}\sum_{i=1}^{N} e_i^2 \text{ ,} \tag{75}$$

and E_W is the mean of the sum of squares of the network weights

$$E_W = \frac{1}{n}\sum_{i=1}^{n} w_i^2 \text{ ,} \tag{76}$$

and α and β are the performance function parameters.

The effective number of parameter used in the network is

$$\gamma = N - 2\alpha TR(H)^{-1}, \tag{77}$$

where N is the total number of parameters in the network, and H is the Hessian matrix calculated according to equation (70).

The estimates of the performance function parameters are computed as

$$\alpha = \frac{\gamma}{2E_W(w)}\text{, and} \tag{78}$$

$$\beta = \frac{n-\gamma}{2E_D(w)}, \tag{79}$$

where w stands for weights, and n is the number of weights.

The procedure of Bayesian regularization applied on neural networks with Levenberg-Marquardt training method is as follows:

1. Initialise weights and biases, and set parameters $\alpha = 0$ and $\beta = 1$.
2. Perform one step of the Levenberg-Marquardt algorithm to minimize the performance function (74).
3. Calculate the effective number of parameters used in the network (77).
4. Calculate the new estimates for performance function parameters according to (78) and (79).
5. Repeat steps 2 – 4 until convergence, or train epochs, or train error goal.

Another testing of multi-layer perceptron network was done with the use of scaled conjugate gradient training method. The training method steps are as follows:

1. Choose initial weights w_1 and scalars $\sigma > 0$, $\lambda_1 > 0$ and $\overline{\lambda}_1 = 0$. Set $p_1 = r_1 = -E'(w_1)$, $k = 1$, where $E'(w) = \left[\sum_{S=1}^{N} \frac{dE_S}{dw_{11}}, \ldots, \sum_{k=1}^{N} \frac{dE_S}{dw_{nm}} \right]$ and $E(w)$ is the mean square error, while E_S is associated to the training data set, and N is the number of training data sets.
2. Calculate the second order information:
$$\sigma_k = \frac{\sigma}{|p_k|},\ s_k = \frac{E'(w_k + \sigma_k p_k) - E'(w_k)}{\sigma_k},\ \delta_k = p_k^T s_k .$$
3. Scale s_k:
$$s_k = s_k + (\lambda_k - \overline{\lambda}_k) p_k ,\ \delta_k = \delta_k + (\lambda_k - \overline{\lambda}_k) |p_k|^2 .$$
4. If $\delta_k \leq 0$ then make Hessian matrix positive definite:
$$s_k = s_k + \left(\lambda_k - 2\frac{\delta_k}{|p_k|^2} \right) p_k ,\ \overline{\lambda}_k = 2\left(\lambda_k - \frac{\delta_k}{|p_k|^2} \right),\ \delta_k = -\delta_k + \lambda_k |p_k|^2 ,\ \lambda_k = \overline{\lambda}_k .$$
5. Calculate step size:
$$\mu_k = p_k^T r_k ,\ \alpha_k = \frac{\mu_k}{\delta_k} .$$
6. Calculate the comparison parameter:
$$\Delta_k = \frac{2\delta_k [E(w_k) - E(w_k + \alpha_k p_k)]}{\mu_k^2} .$$
7. If $\Delta_k \geq 0$ then a successful reduction in mean square error can be done:
$$w_{k+1} = w_k + \alpha_k p_k ,\ r_{k+1} = -E'(w_{k+1}),\ \overline{\lambda}_k = 0 .$$
Else a reduction in error is not possible: $\overline{\lambda}_k = \lambda_k$.
8. If $k == N$ then restart the algorithm: $p_{k+1} = r_{k+1}$,
else create a new conjugate direction:
$$\beta_k = \frac{|r_{k+1}|^2 - r_{k+1} r_k}{\mu_k},\ p_{k+1} = r_{k+1} + \beta_k p_k .$$
9. If $\Delta_k \geq 0.75$ then reduce the scale parameter: $\lambda_k = \frac{1}{2}\lambda_k$.
10. If $\Delta_k < 0.25$ then increase the scale parameter: $\lambda_k = 4\lambda_k$.

11. If the steepest descent direction $r_k \neq 0$ then set $k = k + 1$ and go to step 2, else terminate the training and return the desired weights as w_{k+1}.

The results of the training methods and network structure selection are presented in the Chapter 7 where the results of the body weight recognition methods are given.

6.3.2 Model application

The described neural network classifier can be used for body weight recognition of cattle weighted on the walk-through weighing scale. Once the network structure and the parameters of the classifier are denoted using the proposed training method and training pairs, it is possible to perform body weight recognition from the force signal. The input of the classifier is selected from the force signal which was recorded when the animal was alone on the scale with its whole body weight. A computer program was created to select the maximums and minimums of the force signal based on numerical differentiation of the signal. The force signal peaks are recognised where the differentiated signal changed sign. The maximums and minimums are averaged, respectively, and included in the neural network calculation. The output of the neural network classification is the body weight of the weighted animal.

7 Validations of models and methods

In this section the results of body weight recognition methods in dynamic weighing of cattle exposed in previous chapters are presented. First, the validation of the created mathematical model is done and then the model is applied in body weight recognition. Second, the results of fuzzy method in body weight recognition for single animal crossing case and two animal crowding on the weigher are shown. Third, the neural network suggested in the previous chapter for body weight recognition was trained and tested for single crossing case, as well as for two cows crowding on the weighing platform. Finally, fifty-four dairy cows were weighted on the walk-through weighing scale. The measured force signals were processed with the suggested methods. The results are presented and analysed.

7.1 Validation method

The validation of the created methods for body weight recognition in dynamic weighing of cattle is accomplished comparing the calculated and the experimentally measured body weight. A number of dynamic weighing of dairy cows were completed and the data were processed with the proposed filtering and body weight recognition methods. The animals were also weighted statically in order to denote their true weight before dynamical weighing.

7.2 Mathematical model

The body weight of the cattle weighted dynamically and processed with mathematical model body weight recognition was denoted comparing the real measured values of vertical force produced by cattle feet recorded on dairy farm and that calculated through the model. The cow was let to cross the walk-through weighing scale described in previous section. The signal filtering process was done as explained in the signal processing section of this work. Therefore, the valuable part of the force signal when the animal was alone on the scale with its whole body weight was selected. In order to provide information on accuracy and worthiness of the developed lumped-parameter model of cow motion a number of experiments were done.

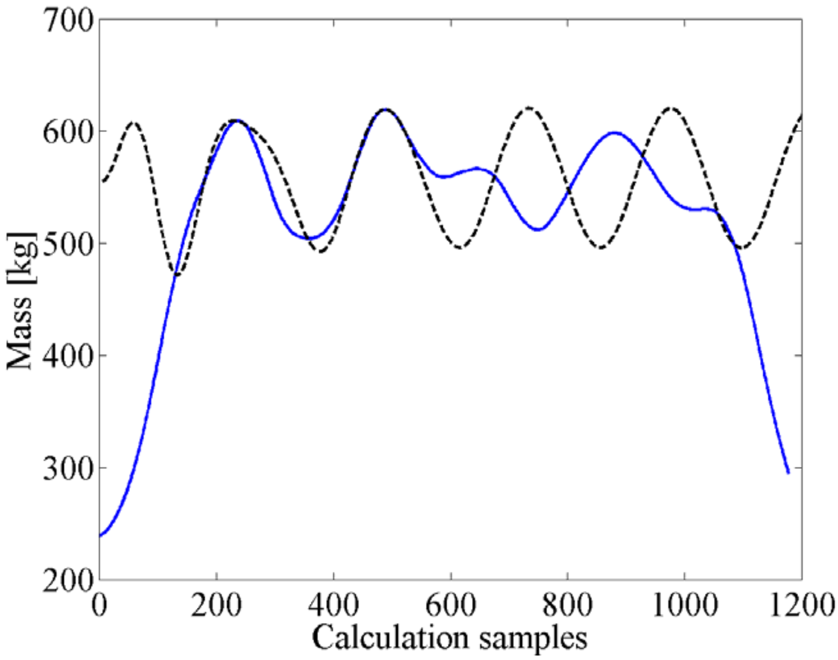


Figure 7.1: Measured and calculated ground reaction forces during cow walk (558 kg).

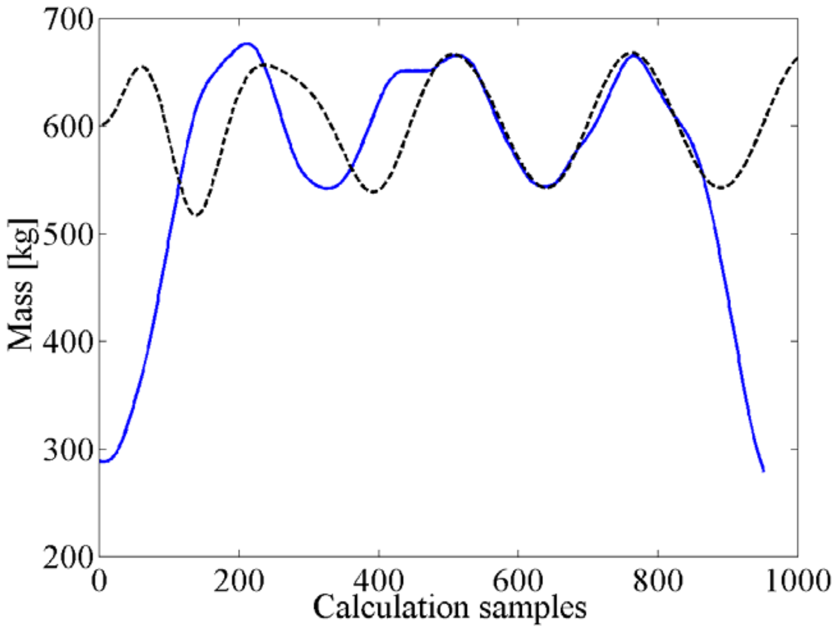


Figure 7.2: Measured and calculated ground reaction forces during cow walk (618 kg).

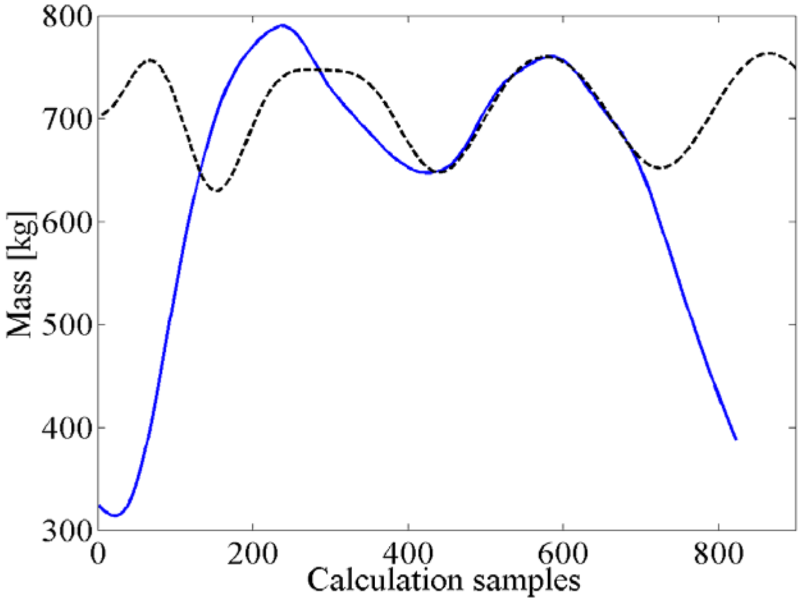


Figure 7.3: Measured and calculated ground reaction forces during cow walk (723 kg).

The experiments were performed on the created walk-through weighing scale of the dairy farm. Eleven cows with mass of: 558, 567, 585, 600, 618, 624, 633, 700, 723, 762 and 860 kilograms were included in tests. With these animals thirty data sets were recorded, processed and used in model verification. To complete the calculated and measured force curve comparison additional computation algorithm was developed as described in Chapter 6. The selected total mass and bars oscillation frequency were put into the mathematical model calculation. The results of the simulation were compared to measured forces as presented in Figures 7.1 – 7.3.

The measured force signal is presented with solid line, while the calculated force curve is marked with dashed line. Analysing the graphs of forces it can be concluded that curves are not ideally matching each other over the whole time interval. As shown in plots only a part of the measured force signal is matching the calculated force curve. This was expected since the model represents synchronised and ideal steps during walking. In reality every step is different. This stochasticity could not be modelled. Yet, the matching parts are long enough to achieve successful comparison of the curves.

The maximal error for single crossing case did not exceed 2% of the correct cattle body weight. Such a small error was not expected due to complexity of animal walk. The simplified model showed good results and the program was capable of recognising the body weight with 2 % error. However, it did not show good results for the case when more cows were following each other in quick succession over the scale. The valuable part of the force signal that was selected after filtering was too short to give sufficient data for accurate body weight recognition process. Therefore, the mathematical model cannot be used in body weight recognition where the crowded crossing case is frequent.

7.3 Fuzzy logic method

A number of experiments were completed for various animals with different masses and walking velocities. The animals included in walk-through weighing were statically weighted for comparison with calculated results. The body mass of the animals was: 558, 567, 585, 600, 618, 624, 633, 700, 723, 762 and 860 kilograms. Each cow repeated at least three walk-through measurements. The database of model parameters used in body weight recognition was as presented in Table 6.1. The results of body weight calculations are shown in Table 7.1.

Table 7.1: Percentage calculation error of measured animals for single crossing.

Body weight (kg)	567	585	600	624	633	700	762	860
Percentage error (%)	1.2	0.2	0.2	1.3	0.2	0.1	1.8	1.9
	1.0	0.6	0.3	0.2	0.5	0.3	1.2	0.1
	1.1	0.2	1.6	0.7	1.5	0.0	0.5	0.1
Average error (%)	1.1	0.3	0.7	0.7	0.7	0.1	1.2	0.7

In Table 7.1 the differences between estimated and true body weights of animals in percent are marked with crosses. As shown in Table 7.1 maximal difference did not exceed 2 %. Among tested animals the maximal error of 17 kilograms was calculated for 860 kilograms animal. The average error for single crossing case was 0.7 %.

For crowded crossing case experiments were performed with same cattle as in single crossing case. Two cows were forced to cross the weighing scale following each other in quick succession. The pairs weighted on the walk-through weighing scale were: 558 - 723, 567 – 633, 585 – 762, 624 – 600 and 618 – 700 kilograms. Impaired cattle crossed the weighing scale a least three times. The calculated body weights were compared to statically measured ones. The errors of calculated body weights were plotted in Table 7.2. As it can be seen the maximal error did not exceed 2.6 %. The average error for crowded crossing case was 1.3 %.

Table 7.2: Percentage calculation error of measured animals for crowded crossing.

Body weight (kg)	558	567	585	600	618	624	633	700	723	762
Percentage error (%)	1.4	1.0	0.3	1.3	0.3	0.8	2.0	2.0	1.2	0.7
	1.2	1.4	2.6	2.3	0.8	0.8	1.1	1.2	1.2	1.4
Average error (%)	1.3	1.2	1.5	1.8	1.0	0.8	1.6	1.6	1.2	1.1

The highest errors occurred for the both crossing cases when model parameters for certain crossing times were not defined in database. For example, the crossing time of the 600 kilograms cow was 3.55 seconds and the maximum database crossing time was 2.14 seconds. This was expected since the fuzzy model could not present stride situations whose walking velocity was much lower or higher than the modelled. For this reason the database of model might be enlarged in order to minimise the calculation error. However, even with this small database the results are satisfying. On the other hand, the database does not need to be enlarged over some reasonable crossing time value. For long crossing times the walking velocity is approximately zero and such signals can be averaged to give good estimation of the animal body weight.

7.4 Neural network method

The Levenberg-Marquardt training method was applied on fourteen training data sets with body weights 558, 618, 633, 700 and 762. The neural network was tested for both single and crowded crossing cases of the weighing scale. The network was tested with seventeen test data sets in single crossing case for body weights 567, 585, 600, 624 and 723. The network is also tested for crowded crossing case with 12 situations of two animals consecutively crossing the scale. The animals body weights included in testing were 558, 567, 585, 600, 618, 624, 633, 700, 723 and 762. The training data sets covered the body weight surface such that testing data were not outside the training surface. Yet, if it would not be the case neural network could not create a reasonable response to testing data. After testing various multi-layer perceptron network structure with the Levenberg-Marquardt training method the best results were obtained for network structures 10-1 and 1-5-1. The hidden layer neurons had hyperbolic tangent sigmoid activation function while the output layer had a linear activation function. The maximal percentage error of training data sets was 1.1 % and the average training error was 0.35 %. The maximal error occurred for single crossing case testing data sets was 2 %, with average error of 0.8 %. The maximal error computed for crowded crossing case testing data sets was 2.7 %, with average error of 0.8 %.

The network was also trained with scaled gradient conjugate training method. The training and testing for different neural network structure were with same input-output data set as for Levenberg-Marquardt training. The best results obtained with this training were for multi-layer perceptron structure 1-5-1 and 10-1, same as with previous training. The optimal results were calculated faster with Levenberg-Marquardt training than with scaled conjugate gradient method. The optimal results were those for which the mean and maximum errors were minimised for training, single testing and crowded testing data sets. Enlarging the size of the network produced reduction in error of training sets, but increase in error for testing data sets.

Adding the time of crossing the scale as another input parameter enlarged the size of the input vector that consisted of minimum and maximum values from the measured force signal. With such a three dimensional input vector and scalar output multi-layer perceptron network was trained. Training algorithms used in network parameter adjustment were the earlier presented Levenberg-Marquardt and scaled conjugate gradient training. The same network architecture was selected for three dimensional input as for two dimensional input vector. Slightly better results were obtained with larger input vector. The average training error was 0.2 %, while the maximal error was 0.8 %. For single crossing testing data average error was 0.6 % and the maximal error was 1.5 %. For crowded crossing case the maximal percentage error was 2.5 % with average error of 0.8 %. However, the testing data pairs that included crossing times bigger and smaller than those introduced with training data sets were discarded, since the network could not response to not trained situations.

Although the results produced by multi-layer perceptron network with three dimensional input were somewhat better than two dimensional input the training of such a network would be a hard task. With maximum and minimum input vector the network could be trained and ready to give reasonable results if the testing body weight was smaller than maximum training body weight and bigger than minimum training body weight. For the network including the information of crossing time significantly decreased the possibility of the network since in many cases the crossing time was not in the trained region. To train the network to cover the whole crossing time region was impossible, since the experimentally gathered data were limited to the possibilities of the animal crossing the weighing scale. Consequently, the neural network including the maximum and minimum of the force signal as input and body weight as output was the reasonable solution. The selected multi-layer perceptron network with 1-5-1 structure seemed to be adequate for body weight recognition task.

7.5 Comparison of developed methods

To inspect the possibilities of the created body weight recognition methods real situations of the dynamic weighing on the dairy farm are investigated. The dairy cows were weighted on the walk-through weighing scale. The measured force signals were coupled with the identification (ID) signals. The force signals were processed with the signal filtering methods to select the valuable parts of the signals. The chosen parts of the force signals were associated to the cattle ID numbers. From this stage the fuzzy approximation and the neural network classification methods were independently applied on the signals. The calculated body weights of the animals were compared to the statically measured body weights.

Fifty-four cows were measured on the walk-through weighing scale. Nine groups of six cows were formed. After milking six cows were let to cross the weighing scale as they were leaving the milking parlour. After crossing the scale the animals were sent to the milking parlour and let to cross the scale again. They were measured dynamically three times on the weighing scale. All animals were weighted statically on the same weighing scale by closing the entrance and exit doors mounted on the construction of the device. The measured force versus time was processed with the signal filtering and the body weight recognition methods. Results of the methods were compared to the statically measured body weight.

The errors of calculated body weights with the fuzzy approximation method are given in Table 7.3.

Table 7.3: Errors of calculated body weights using the fuzzy approximation method for groups of cattle.

Group	1	2	3	4	5	6	7	8	9	Mean
MPE [%]	1.2	2.8	2.5	1.5	2.0	2.0	2.7	3.0	2.7	2.3
APE [%]	0.7	1.0	0.9	0.8	0.9	0.8	1.0	1.3	1.2	1.0

The maximal percentage error (MPE) and the average percentage error (APE) are computed for each group of six cows weighted on the walk-through weighing scale. The MPE does not exceed 3 % error, while the average error is 1 %. In all groups crossing the weighing scale the calculated body weight error was larger for crowded crossing case than for single crossing case. The database model parameters were used in the body weight recognition process as presented in Table 6.1. The database model parameters covered bigger crossing times than defined in the database. For the crossing times bigger than 5 seconds averaging of force signal was performed. For such a long signal mean value gives reasonable accuracy since the animal is stationary on the scale.

For the same measured force signals for the fifty-four cows the neural network classification body weight recognition method was applied. The selected multi-layer perceptron network with 1-5-1 structure was used. The weights and biases of the neural network were denoted after training the network with Levenberg-Marquardt training algorithm. The training pairs were formed for the cattle with body mass of: 558, 618, 633, 700 and 762 kg. For each animal three data pairs were formed. As suggested earlier the inputs were the maximum and the minimum of the force signal valuable part, while the output was the statically measured body weight of the animal. The neural network was tested the same way as fuzzy approximation method and the results are shown in Table 7.4.

Table 7.4: Errors of calculated body weights using the neural network method for groups of cattle.

Group	1	2	3	4	5	6	7	8	9	Mean
MPE [%]	1.3	1.6	2.0	1.5	1.8	1.2	3.5	3.0	2.4	2.0
APE [%]	0.6	0.7	0.8	0.7	0.7	0.5	1.2	1.1	1.0	0.8

The maximal percentage error (MPE) and the average percentage error (APE) are computed for each group of six cows weighted on the walk-through weighing scale. The largest value of MPE occurred for the group seven when the crowding on the weighing scale of six animals was measured. The maximal error compared to the correct body weight is 3.5 %. The average error for all calculated situations is 0.8 %.

7.6 Summary

Body weight estimation was performed on dairy cows weighted dynamically after milking process. The recorded force signal was filtered and the valuable parts were extracted. The body weight recognition methods were applied on the force signal.

Signal processing performed good signal filtering for both crossing cases of the scale. Mathematical model based body weight recognition showed accuracy of 2 % for single crossing case, but for crowded crossing case it could not be successfully used due to briefness of the processed signal. Fuzzy logic based body weight recognition was effectively used for body weight estimation with accuracy of 3 % (average 1 %) for both single and crowded crossing case. Neural networks based body weight recognition denoted body weights of dynamically weighted animals with preciseness of 3.5 % (average 0.8 %) for all crossing situations.

8 Discussion

To evaluate the results in dynamic weighing achieved with the created signal processing and body weight recognition methods, some association to the earlier developed dynamic weighing methods need to be done. The comparison is to be done with the most influencing methods of dynamic weighing proposed by FILBY et al. (1979) [19], PEIPER et al. (1993) [58] and REN et al. (1992) [62]. Comparing the achievements of the developed methods for body weight estimation to the results of earlier dynamic weighing methods some major differences can be noticed. The comparison is done for weighing scale design, data sampling, signal filtering and above all the body weight estimation accuracy.

The weighing platform design is quite simple unlike that of PEIPER et al. (1993) [58], which contained a step to slow down animal movement and to separate them and so prevent crowding. By the simplicity of the weighing platform design REN et al. (1992) [62] is the closest to that here presented. The data sampling rate here used (100 samples per second) is much higher than those earlier proposed (5 – 20 samples per second). This, however, enables more preciseness, since the valuable parts of the signal are not lost, but increases the amount of information and complicates signal processing.

The earlier proposed signal processing (FILBY et al. 1979 [19], PEIPER et al. 1993 [58] and REN et al. 1992 [62]) was based on force signal averaging and comparison to the reference body weight value. FILBY et al. (1979) [19] performed averaging of the incoming signal and kept the maximum of such signal as new body weight if within 30 kilograms compared to previous body weight of the same animal. This was an initial and simple signal processing method. More complexity in signal processing was proposed in later research. REN et al. (1992) [62] suggested comparing of the incoming signal with the reference value (previous measured body weight) with allowed 5 % divergence and then averaging the filtered signal. The most sophisticated force signal processing introduced by PEIPER et al. (1993) [58] proposed averaging of incoming force signal and then a special algorithm recognised the relevant body weight from a vector of averaged data. Afterward the selected body weight was validated comparing to the reference value (previous measured body weight) with allowed error margin of 30 kilograms. The last two methods showed similar accuracy. For two daily measurements (REN et al. 1992 [62]) 78 % were acceptable weights, with weekly accuracy of 2 % compared to statically measured weight. For three daily measurements (PEIPER et al. 1993 [58]) 76.5 % were acceptable weights, with monthly accuracy of 1.5 % compared to statically measured weight. As a result, at least for thirty cows from one hundred, there were no new body weight data after dynamic weighing. In other words, 30 % of the animals weighted on walk-through weighing scale did not satisfy the error margin of 30 kilograms (roughly 5 % of the previous recorded body weight). This was caused by fast crossing over the scale and crowding of animals on the weighing platform.

The here presented signal processing method coupled with body weight recognition methods showed better outcome after dynamic weighing than the above mentioned. All the cows weighted on the walk-through weighing scale had good readings of force signal and the weights were always estimated. The previous body weight of the animal was relevant for signal filtering in crowding crossing case not as an error margin, but as a calculation helper for better signal separation. The estimated body weight compared to the statically measured weight diverted for the maximum of 3.5 %, while the average divergence was 1 %. Consequently, one hundred percent of dynamically weighted animals were within the 3.5 % body weight estimation error, which was much lower estimation error than the earlier achieved. The most important of all is that the methods successfully solved fast and crowded crossing cases. There were no unrecognised weighing situations, which is the major breakthrough in dynamic weighing. This enables that every animal is weighted each time it is measured on the walk-through weighing scale.

Comparing among the three developed body weight recognition methods, the fuzzy logic and neural network methods showed better estimation possibilities than the mathematical model. The mathematical model estimated body weight for single crossing cases with accuracy of 2 %. Crowded crossing cases could not be processed with this method due to the briefness of the measured signal. Fuzzy and neural network procedures were applicable for both single and crowded cases and showed similar accuracy. The fuzzy method showed lower average accuracy (1 %) than the neural network method (0.8 %) when used in body weight recognition. However, slightly higher inaccuracies were reached with the neural network method (3.5 %) than with the fuzzy logic method (3 %). The neural network method required less computation because the model parameters were earlier defined. The fuzzy method required more calculation due to function approximation and database model computation. The advantage of the fuzzy method was that the database could be easily enlarged adding new model parameters, while the neural network needed to be separately trained for sets of training parameters and then the new parameters of the model were entered. Generally, the neural network method would be easier to implement in a separate device used for commercial purposes on dairy farms.

The created methods estimate the body weight of animals with maximum error of approximately 20 kilograms, which is smaller than daily body weight oscillation of 30 kilograms. This fact discards the possibility of false alert for animal sickness and enables good tracking of animal health condition. Such a secure and precise data on animal body weight is crucial for completing the gathering of relevant parameters for each animal.

The created signal processing and body weight recognition methods applied in dynamic weighing of cattle successfully accomplished body weight estimation for both single and crowded crossing cases, as well as for fast and slow crossing speeds.

9 Conclusion

The dynamic weighing of dairy cows, especially the processing of the measured force signal is elaborated in this work. The weighing scale made by DeLaval was used in dynamic weighing of cattle. The dimension of the weighing platform was correctly chosen. It allowed one cow to stay with its whole body weight on the platform enough time to gather all the relevant force data. On the other hand, it enabled the maximum of three pairs of leg to touch the platform at once. It simplified the procedure of signal filtering in crowded crossing case. The force signal measured on the weighing platform was recorded. The signal processing performed force signal filtering from noise and separation of the signal part when a cow was measured with its whole body weight. Adjusting the force signal in frequency domain the signal was modified in preferred form. The body weight recognition methods were applied on the filtered signal to determine the body weight of the animal weighted on the walk-through weighing scale. Three independent methods were developed for body weight estimation. All of the methods were produced for stochastic non-linear system of cow walk. In the mathematical model the stochastic influence was ignored, while in fuzzy and neural model it was included. Mathematical model method could not be used in crowded crossing case weighing, but fuzzy logic and neural network methods responded on all weighing cases giving the estimation of animal’s body weight. The mathematical model can be successfully used in body weight recognition for dynamic weighing without crowding. This might be the case for rotary milking parlours, where the separate exit of animals is common. The fuzzy logic and neural network model were based on completely different concepts. The fuzzy body weight recognition method contained database of model parameters, which were created from experiments of dynamic weighing. The database models were used for body weight estimation of new measured animals. The neural network classification method was used for body weight recognition after the network was trained with experimentally obtained training sets.

The methods were tested on a large number of situations of crossing the scale. Various animals were tested under different conditions, so all possible cases of dynamic weighing were inspected. The created signal processing and body weight recognition methods applied in dynamic weighing of cattle were successful in body weight estimation. The fuzzy logic and neural network method with described signal processing can be used in commercial walk-through weighing devices on dairy farms as an accurate and secure weighing system.

References

[1] **Abe, S., Lan, M.S.:** Fuzzy rules extraction directly from numerical data for function approximation. In: IEEE transactions on systems, man, and cybernetics (1995), 25: 119-129.

[2] **Aguirre, L.A., Billings, S.A.:** Improved structure selection for nonlinear models based on term clustering. In: International journal of control (1995), 62: 569-587.

[3] **Alexander, R.McN.:** Mechanics of Bipedal Locomotion. In: P. Spencer Davies (ed): Perspectives in Experimental Biology. Pergamon, Oxford, (1976), 1: 493-504.

[4] **Alexander, R.McN.:** Mechanics of Animal Locomotion. Springer, Berlin, 1992.

[5] **Alexander, R.McN.:** Simple Model of Human Movement. In: Applied Mechanical Review (1995), 48: 461-469.

[6] **Anderson, J.A.:** A simple neural network generating an interactive memory. In: Mathematical biosciences (1972), 14: 179-220.

[7] **Anderson, J.A., Silverstein, J.W., Ritz, S.A., Jones, R.S.:** Distinctive features, categorical perception, and probability learning: some application of a neural model. In: Psychological Review (1977), 84: 413-451.

[8] **Battiti, R.:** First and second order methods for learning: between steepest descent and Newton's method. In: Neural computation (1992), 4: 141-166.

[9] **Beale, E.M.L.:** A derivation of conjugate gradients. In: Lootsma, F.A.: Numerical methods for nonlinear optimisation, London: Academic Press, 1972.

[10] **Billings, S.A., Korenberg, M.J., Chen, S.:** Identification of non-linear output-affine systems using an orthogonal least-squares algorithm. In: International journal of control (1988), 19: 1559-1568.

[11] **Billings, S.A., Chen, S., Korenberg, M.J.:** Identification of MIMO non-linear systems using a forward-regression orthogonal estimator. In: International journal of control (1989), 49: 2157-2189.

[12] **Buttazzoni, L., Mao, I.L.:** Genetic Parameters of Estimated Net Efficiencies for Milk Production, Maintenance, and Body Weight Change in Dairy Cows. In: Journal of Dairy Science (1989), 72: 671-677.

[13] **Castro, J.L.:** Fuzzy logic controllers are universal approximators. In: IEEE transactions on systems, man, and cybernetics (1995), 25: 629-635.

[14] **Castro, J.L., Delgado, M.:** Fuzzy systems with defuzzification are universal approximators. In: IEEE transactions on systems, man, and cybernetics (1996), 26: 149-152.

[15] **Charalambous, C.:** Conjugate gradient algorithm for efficient training of artificial neural networks. In: IEEE Proceedings (1992), 139: 301-310.

[16] **Chen, S., Billings, S.A., Grant, P.M.:** Non-linear system identification using neural networks. In: International journal of control (1990), 51: 1191-1214.

[17] **Cybenko, G.:** Approximations by superpositions of sigmoidal functions. In: Mathematics of control, signals and systems (1989), 2: 303-314.

[18] **Dickerson, J.A., Kosko, B.:** Fuzzy function approximation with ellipsoidal rules. In: IEEE transactions on systems, man, and cybernetics (1996), 26: 542-560.

[19] **Filby, D.E., Turner, M.J.B., Street, M.J.:** A Walk-through Weigher for Dairy Cows. In: Journal of Agricultural Engineering Research (1979), 24: 67-78.

[20] **Fletcher, R., Reeves, C.M.:** Function minimization by conjugate gradients. In: Computer journal (1964), 7: 149-154.

[21] **Fukusima, K.:** A hierarchical neural network capable of visual pattern recognition. In: Neural networks (1988), 1:119-130.

[22] **Funahashi, K.:** On the approximation realization of continuous mappings by neural networks. In: Neural networks (1989), 2:183-192.

[23] **Garcia, M., Chatterjee, A., Ruina, A., Coleman, M.:** The Simplex Walking Model: Stability, Complexity, and Scaling. In: Journal of Biomechanical Engineering (1998), 120: 281-288.

[24] **Girosi, F., Poggio, T.:** Networks and the best approximation property. In: Biological cybernetics (1990), 63: 169-176.

[25] **Gonzales, J., Rojas, I., Pomares, H., Ortega, J.:** A new clustering technique for function approximation. In: IEEE transaction on neural networks (2002), 13: 132-142.

[26] **Haber, R., Unbehauen, H.:** Structure identification of nonlinear dynamic systems – a survey on input/output approaches. In: Automatica (1990), 26: 651-677.

[27] **Hagan, M.T., Menhaj, M.:** Training forward networks with the Marquardt algorithm. In: IEEE Transactions on neural networks (1994), 6: 989-993.

[28] **Hagan, M.T., Demuth, H.B., Beale, M.H.:** Neural network design. Boston, MA: PWS Publishing, 1996.

[29] **Hartman, E.J., Keeler, J.D., Kowalski, J.M.:** Layered neural networks with Gaussian hidden units as universal approximators. In: Neural computation (1990), 2: 210-215.

[30] **Haykin, S.:** Neural networks: a comprehensive foundation. Upper Saddle River: Prentice-Hall Inc., 1999.

[31] **Hecht-Nielsen, R.:** Neurocomputing. Reading, MA: Addison-Wesley Publishing, 1990.

[32] **Hilton, G.F., Sejnowski, T.J., Ackley, D.H.:** Boltzmann machines: constraint satisfaction networks that learn. In: Carnegie Mellon technical paper (1984), CMU-CS-84-119.

[33] **Hopfield, J.J.:** Neural networks and physical systems with element collective computational abilities. In: Proceeding of the national academy of science (1982), 79: 2554-2558.

[34] **Hubbard, M.:** Computer Simulation in Sport and Industry. In: Journal of Biomechanics (1993), 26: 53-61.

[35] **Ipema, A.H., Pluijgers, H.G.:** Automatic Weighing of Dairy Cows. In: Automation in Dairying, Proceedings: Third Symposium, September 9-11, 1987, Wageningen, The Netherlands, 110-115.

[36] **Jang, J.S.R.:** ANFIS: Adaptive-network-based fuzzy inference system. In: IEEE transactions on systems, man, and cybernetics (1993), 23: 665-685.

[37] **Juditsky, A., Hjalmarsson, H., Benveniste, A., Delyon, B., Ljung, L., Sjöberg, J., Kohonen, T.:** Correlation matrix memories. In: IEEE computers (1972), 21:353-359.

[38] **Kohonen, T.:** Self-organized and associative memory. Berlin: Springer-Verlag,1987.

[39] **Landajo, M., Rio, M.J., Perez, R.:** A note on smooth approximation capabilities of fuzzy systems. In: IEEE transactions on fuzzy systems (2001), 9: 229-237.

[40] **Leontaritis, I.J., Billings S.A.:** Input-output parametric models for non-linear systems. Part I: deterministic non-linear systems. In: International journal of control (1985), 41: 303-328.

[41] **Leontaritis, I.J., Billings S.A.:** Input-output parametric models for non-linear systems. Part II: stochastic non-linear systems. In: International journal of control (1985), 41: 303-328.

[42] **Ljung, L.:** System identification: Theory for the user. Prentice-Hall PTR, Upper Saddle River, NJ, 1987.

[43] **Ljung, L., Glad, T.:** Modeling of dynamic systems. Prentice-Hall PTR, Englewood Cliffs, NJ, 1994.

[44] **Maltz, E., Kroll, O., Spahr, S.L.:** Parameters Required for Individual Cocentrates Supplementation of the Dairy Cow: Their Recording, Analyse and Use. In: Automation in Dairying, Proceedings: Third Symposium, September 9-11, 1987, Wageningen, The Netherlands, 86-101.

[45] **Maltz, E., Kroll, O., Spahr, S.L., Devir, S., Genizi, A., Sagi, R.:** Milk yield, parity and cow potential as variables for computerized concentrate supplementation strategy. In: Journal of Dairy Science (1990), 74: 2277-2289.

[46] **Maltz, E., Devir, S., Kroll, O., Zur, B., Spahr, S.L., Shanks, R.D.:** Comparative Responses of Lactating Cows to Total Mixed Rations of Computerized Individual Concentrates Feeding. In: Journal of Dairy Science (1992), 75: 1588-1603.

[47] **Margaria, R.:** Biomechanics and Energetics of Muscular Exercise. Claredon Press, Oxford, 1976.

[48] **Mao K.Z., Billings S.A.:** Algorithms for minimal model structure detection in non-lineat dynamic system identification. In: International journal of control (1997), 68: 31-330.

[49] **McGeer, T.:** Passive dynamic walking. In: International Journal of Robotics (1990), 9: 62-82.

[50] **McGeer, T.:** Principles of walking and running. In: R.McN. Alexander (ed): Mechanics of Animal Locomotion. Springer, Berlin (1992), 11: 113-139.

[51] **Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, M., Teller, E.:** Equation of state calculations by fast computing machines. In: Journal of chemical physics (1953), 21: 1087-1092.

[52] **Moller, M.F. :** A scale conjugate gradient algorithm for fast supervised learning. In: Neural networks (1993), 6: 525-533.

[53] **Moore, B., Poggio, T.:** Representations properties of multilayer feedforward networks. In: Abstracts of the first annual INNS meeting, NY: Pergamon Press (1988), 502.

[54] **Nguyen, D., Widrow, B.:** Improving the learning speed of 2-layer neural networks by choosing initial values of adaptive weights. In: Proceedings of the international joint conference on neural networks (1990), 3: 21-26.

[55] **Parker, S.R., Perry, F.A.:** A discrete ARMA model for nonlinear system identification. In: IEEE transactions on circuits and systems (1981), 28: 224-233.

[56] **Park, J., Sandberg, I.W.:** Universal approximation using radial basis function networks. In: Neural computation (1991), 3: 246-257.

[57] **Peiper, U.M., Maltz, E., Edan, Y., Babazada, M., Schon, H., Artman, R.:** Routine Automatic Dairy Cow's Body Weight Measurements for the Use of Husbandry Decisions. In: Automation in Dairying, Proceedings: Third Symposium, September 9-11, 1987, Wageningen, The Netherlands, 116-128.

[58] **Peiper, U.M., Edan, Y., Devir, S., Barak, E., Maltz, E.:** Automatic Weighing of Dairy Cows. In: Journal of Agricultural Engineering Research (1993), 56: 13-24.

[59] **Pomares, H., Rojas, I., Gonzales, J., Prieto, A. :** Structure identification in complete rule-based fuzzy systems. In: IEEE transactions on fuzzy systems (2002), 10: 349-359.

[60] **Powell, M.J.D.:** Restart procedures for the conjugate gradient method. In: Mathematical programming (1977), 12: 241-254.

[61] **Raibert, M.H.:** Legged Robots that Balance. The MIT Press, Cambridge, MA, 1986.

[62] **Ren, J., Buck, N.L., Spahr, S.L.:** A Dynamic Weight Logging System for Dairy Cows. In: Transactions of the ASAE (1992), 35(2): 719-725.

[63] **Rojas, I., Pomares, H., Ortega, J., Prieto, A.:** Self-organized fuzzy system generation from training examples. In: IEEE transactions on fuzzy systems (2000), 8: 23-36.

[64] **Rosenblatt, F.:** Principles of neurodynamics. Washington DC: Spartan books, 1962.

[65] **Rumelhart, D.E., Hinton, G.E., Williams, R.J.:** Learning representations by back-propagation errors. In: Nature (1986), 323: 533-536.

[66] **Rumelhart, D.E., Hinton, G.E., Williams, R.J.:** Learning internal representations by error propagation. In: Rumelhart, D.E., McClelland, J.L.: Parallel data processing, Vol1., Cambridge MA: The MIT Press (1986), 318-362.

[67] **Runkler, T.A., Bazdek, J.C.:** Alternating cluster estimation: A new tool for clustering and function approximation. In: IEEE transactions on fuzzy systems (1999), 7: 377-393.

[68] **Sagi, R., Kroll, O., Maltz, E., Spahr, S.L.:** Individual concentrates supplementation of dairy cows according to milk production. In: (preparation) (1986)

[69] **Salmeri, M., Mencattini, A., Rovatti, R.:** Functional approximation using non-normalized SISO fuzzy systems. In: International journal of approximate reasoning (2001), 26: 211-231.

[70] **Sjöberg, J., Zhang, Q., Ljung, L., Benveniste, A., Delyon, B., Glorennec, P.Y., Hjalmarsson, H., Juditsky, A.:** Non-linear black-box modelling in system identification: a unified overview. In: Automatica (1995), 31: 1691-1724.

[71] **Söderström, T., Stoica, P.:** System identification. Prentice-Hall International, Hertfordshire, UK, 1989.

[72] **Spahr, S.L., Shanks, R.D., McCoy, G.C., Maltz, E., Kroll, O.:** Lactation Potential as a Criterion for Strategy of Feeding Total Mixed Rations to Dairy Cows. In: Journal of Dairy Science (1993), 76: 2723-2735.

[73] **Sugeno, M., Kang, G.T.:** Structure identification of fuzzy model. In: Fuzzy sets and systems (1988), 28: 15-33.

[74] **Takagi, T., Sugeno, M.:** Fuzzy identification of systems and its application to modelling and control. In: IEEE transactions on systems, man, and cybernetics (1985), 15: 116-132.

[75] **Thawonmas, R., Abe, S.:** Function approximation based on fuzzy rules extracted from partitioned numerical data. In: IEEE transactions on systems, man, and cybernetics (1999), 29: 525-534.

[76] **Thorstensson, A., Roberthson, H.:** Adaptation to Changing Speed in Human Locomotion: Speed of Transition between walking and running. In: Acta Physiologica Scandinavica (1987), 131: 211-214.

[77] **Todd, D.J.:** Walking Machines: An Introduction to Legged Robots. Kogan Page Ltd., 1985.

[78] **Vogl, T.P., Mangis, j.K., Rigler, A.K., Zink, W.T., Alkon, D.L.:** Accelerating the convergence of the backpropagation method. In: Biological cybernetics (1988), 59: 256-264.

[79] **Vukobratovic, M., Borovac, B., Surla, D., Stokic, D.:** Biped Locomotion, Springer Verlag, New York Inc., 1990.

[80] **Wang, X.L., Mendel, J.M.:** Fuzzy basis function, universal approximation, and orthogonal least-square learning. In: IEEE transactions on neural networks (1992), 3: 807-814.

[81] **Wang, L.X., Wie, C.:** Approximation accuracy of some neuro-fuzzy approaches. In: IEEE transactions on fuzzy systems (2000), 8: 470-478.

[82] **Wang, L.X.:** A course in fuzzy systems and control. London: Prentice Hall PTR, 1997.

[83] **Wasserman, P.D.:** Advanced methods in neural computing. New York: Van Nostrand Reinhold, 1993.

[84] **Widrow, B., Hoff, M.E.:** Adaptive switching circuits. In: IRE WESCON conventional record (1960), 96-104.

[85] **Widrow, B., Sterns, S.D.:** Adaptive signal processing. New York: Prentice-Hall, 1985.

[86] **Williams, S.R.O., Moore, G.A., Currie, E.:** Automatic Weighing of Pigs Fed ad Libitum. In: Journal of Agricultural Engineering Research (1996), 46: 1-10.

[87] **Ying, H.:** Sufficient conditions on general fuzzy systems as function approximators. In: Automatica (1994), 30: 521-525.

[88] **Zarrugh, M.Y., Todd, F.N., Ralston, H.J.:** Optimization of Energy Expenditure during Level Walking. In: European Journal of Applied Physiology (1974), 33: 293-306.

[89] **Zeng, X.J., Singh, M.G.:** Approximation theory of fuzzy systems SISO case. In: IEEE transactions on fuzzy systems (1994), 2: 162-176.

[90] **Zhang, Q.:** Non-linear black-box modelling in system identification: mathematical foundation. In: Automatica (1995), 31: 1724-1750.