Universidade Estadual de Campinas
Faculdade de Engenharia Agrícola

Bruno Henrique Pereira

Ajuste de parâmetros para o crescimento de indivíduos por meio de assimilação de dados aplicado a bovinos de corte

Parameter fitting for individual growth through data assimilation applied to beef cattle

CAMPINAS
2021

Bruno Henrique Pereira

Parameter fitting for individual growth through data assimilation applied to beef cattle

Ajuste de parâmetros para o crescimento de indivíduos por meio de assimilação de dados aplicado a bovinos de corte

Dissertation presented to the Faculty of Agricultural Engineering of the University of Campinas in partial fulfillment of the requirements for the degree of Master in Agricultural Engineering, in the area of Digital Agriculture.

Dissertação apresentada à Faculdade de Engenharia Agricola da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Agricola, na área de Agricultura Digital.

Supervisor: Prof. Dr. Stanley Robson de Medeiros Oliveira
Co-supervisor: Dr. Luis Gustavo Barioni

ESTE TRABALHO CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO DEFENDIDA PELO ALUNO BRUNO HENRIQUE PEREIRA E ORIENTADA PELO PROF. DR. STANLEY ROBSON DE MEDEIROS OLIVEIRA.

Campinas
2021

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca da Área de Engenharia e Arquitetura
Rose Meire da Silva - CRB 8/5974

Pereira, Bruno Henrique, 1992-
P414p Parameter fitting for individual growth through data assimilation applied to beef cattle / Bruno Henrique Pereira. – Campinas, SP : [s.n.], 2021.

Orientador: Stanley Robson de Medeiros Oliveira.
Coorientador: Luís Gustavo Barioni.
Dissertação (mestrado) – Universidade Estadual de Campinas, Faculdade de Engenharia Agrícola.

1. Sistemas dinâmicos. 2. Filtragem de Kalman. 3. Identificação de sistemas. 4. RFID. 5. Administração da produção - Processamento de dados. I. Oliveira, Stanley Robson de Medeiros, 1967-. II. Barioni, Luís Gustavo, 1969-. III. Universidade Estadual de Campinas. Faculdade de Engenharia Agrícola. IV. Título.

Informações para Biblioteca Digital

Título em outro idioma: Ajuste de parâmetros para o crescimento de indivíduos por meio de assimilação de dados aplicado a bovinos de corte
Palavras-chave em inglês:
Dynamical systems
Kalman filtering
System identification
RFID
Production management - Data processing
Área de concentração: Agricultura Digital
Titulação: Mestre em Engenharia Agrícola
Banca examinadora:
Stanley Robson de Medeiros Oliveira [Orientador]
João Bosco Ribeiro do Val
Angel Pontin Garcia
Data de defesa: 07-05-2021
Programa de Pós-Graduação: Engenharia Agrícola

Identificação e informações acadêmicas do(a) aluno(a)
- ORCID do autor: 0000-0002-5614-6127
- Currículo Lattes do autor: http://lattes.cnpq.br/6263398972535781

Este exemplar corresponde à redação final da **Tese de Doutorado/Dissertação de Mestrado** defendida por **Bruno Henrique Pereira**, aprovada pela Comissão Julgadora em 7 de maio de 2021, na Faculdade de Engenharia Agrícola da Universidade Estadual de Campinas.

Prof. Dr. Stanley Robson de Medeiros Oliveira – Presidente e

Dr. João Bosco Ribeiro do Val – Membro Titular

Dr. Angel Pontin Garcia – Membro Titular

A Ata da defesa com as respectivas assinaturas dos membros encontra-se no SIGA/Sistema de Fluxo de Dissertação/Tese e na Secretaria do Programa da Unidade.

Acknowledgements

I would like to express my sincere gratitude to Prof. Dr. Stanley Robson de Medeiros Oliveira, my supervisor, for all his assistance in this dissertation. I am also grateful to Dr. Luis Gustavo Barioni, my co-supervisor, for his invaluable expertise and guidance during this project.

My gratitude to the Faculty of Agricultural Engineering at Unicamp, for the opportunity to participate in the master's program in the period from 2019 to 2021.

My acknowledgments to Embrapa Agricutural Informatics for the opportunity to develop part of my master's program at its facilities, and to the Coimma company, for providing the necessary data for this project.

I would also like to thank the support I received from the technical team, led by Dr. Luis Barioni, composed of Beatriz Aria, Larissa Macul, Giovanna Castello and Natália Ramos Vilas Boas for their collaboration in carrying out this work.

My thanks the members of the examining committee for my qualification exam and the defense of the master's degree as well.

Most of all, I would like to thank my mother, Kátia, whom I can always count on for support and encouragement.

Resumo

Os criadores de gado frequentemente tomam decisões em um ambiente desafiador, onde dados objetivos são escassos e ruidosos, enquanto têm de lidar com mudanças climáticas, diferenças fenotípicas entre animais, bem como outros fatores. O planejamento objetivo baseado em dados de crescimento animal e previsão de produtividade ainda carece de instrumentação com tecnologias robustas e escaláveis. Aproveitando as tecnologias cada vez mais comuns de coleta automatizada de dados, esta pesquisa propõe o uso de técnicas de assimilação de dados, em conjunto com dados de crescimento animal, para melhorar as previsões produzidas por modelos de crescimento animal, possibilitando o planejamento de sistemas pastoris em um sistema de pecuária. Os dados de crescimento animal foram obtidos usando balanças que medem automaticamente o peso do gado, identificando indivíduos por meio de brincos RFID. Espera-se que os resultados obtidos contribuam para o avanço do estado da arte na previsão de modelos de crescimento animal e auxiliem no manejo e manutenção de sistemas de produção pecuária.

Palavras-chave: Sistemas dinâmicos, filtragem de Kalman, identificação de sistemas, RFID, admnistração da produção - processamento de dados

Abstract

Livestock farmers often make decisions in a challenging environment where objective data is scarce and noisy, while having to deal with climate changes, phenotypical differences between animals, as well as other factors. Objective planning based on animal growth data and productivity forecast still lacks instrumentation with robust and scalable technologies. Taking advantage of the increasingly common technologies for automated data collection, this research proposes the use of data assimilation techniques, in conjunction with animal growth data, to improve forecasts produced by animal growth models, thus enabling the planning of pastoral systems in a livestock system. The animal growth data was obtained using scales that automatically measure livestock weight several times a day, identifying individuals through RFID earrings. The results obtained are expected to contribute to the progress of the state of the art in the forecasting of animal growth models and to assist the management and maintenance of livestock production systems.

Keywords: Dynamical sytems, Kalman filtering, system identification, RFID, production management - data processing

List of Acronyms

DA	Data assimilation
KF	Kalman filter
EFK	Extended Kalman filter
UKF	Unscented Kalman Filter
EBKF	Empirical Bayesian derivation of the Kalman filter
PF	Particle filter
UPF	Unscented particle filter
UT	Unscented transform
SIS	sequential importance sampling
PSO	Particle swarm optimization
RE	Retained energy concentration
EBW	Empty body weigh
LWmat	Mature liveweight
EIR	Energy input rate
MEI	Metabolizable energy intake
SBW	Shrunk body weight
Maint	Energy required for maintenance and activity
LW	Liveweight
NSAF-MAE	Mean Absolute Error of N-Step Ahead Forecasts
RSD	Rolling standard deviation

MRSD	Mean rolling standard deviation
BCS	Base case scenario
MSE	Mean squared error
i.i.d.	Independent and identically distributed
RFID	Radio frequency identification

Contents

Chapter 1

Introduction

Rational decisions stem from our ability to estimate the future outcomes [12]. In cattle farming, there are several decisions that depend on estimates of animal growth rates [5, 7, 1, 38]. These include: (a) financial decisions based on forecast of expenses and revenues; (b) optimal economic slaughter time; (c) times of purchase and sale of animals; (d) optimal diet formulation; (e) amount supplementary feed to be purchased and offered; (f) pasture stocking rates; (g) selection of animals for maintenance and disposal, and; (h) formation of animal batches. Managers of beef cattle production systems are challenged to make such decisions to maximize the economic return and minimize negative externalities, while still having to deal with seasonal variations in pasture productivity [7] and significant genetic heterogeneity between animals [54].

The scarcity of objective information about the production makes it difficult to efficiently manage it, both in the planning and control phases, leading most farmers to make decisions based on trial and error, qualitative assessments and successful experiences of third parties in similar situations [51].

Spatiotemporal data sampling (observations or measurements) of a system can be used to acquire direct information about the system's variables. Measurements usually have low bias, meaning that with enough measurements we should be able to get an accurate estimate of the system's state. However, measurements may be expensive, have too much noise, and often the variable of interest simply cannot be directly measured. Mathematical models can be used to enhance the data acquired through measurements, and to obtain further insight about the system's characteristics and behaviours, such as estimating variables that cannot be observed and forecasting future states of the system. Estimates from models, however, may be hindered by poor modeling, poor parametrization or noisy input data, all of which can, over time, induce estimation bias.

There is a variety of mathematical models that are applied in livestock production forecasting based on species, grazing management and meteorological data [5, 45]. Although models of dynamics of cattle production systems have a history of at least

four decades, it is still frequent to observe bias in the produced forecasts. This bias is caused by two main reasons: (a) mathematical models are simplifications of processes in the system of interest. For this reason, mathematical models may not consider several factors that can impact productivity in the system; (b) parametrization of the models requires large amounts of measurement data, which are costly to obtain, and are usually not available in large commercial productions. In addition, different regions and individuals have different characteristics that affect the forecasts. Calibrating the model for each situation is much more difficult, due to the small amount measurements that can be performed, and usual calibration methods are not adequate.

Detailed data collection for individual animals is still rare in livestock production systems. Historically, there were high costs and effort associated with obtaining it, further inhibiting the adoption of mathematical models for livestock growth forecasting. The recent emergence of automated measurement systems, such as automatic electronic scales for weighting cattle [15], enables real time monitoring of animal growth in commercial productions. Weight data on its own is still not enough to parametrize animal growth models such as Oltjen *et al.* [43] and NASEM [40], especially for calibrating the model for small populations or short time periods of monitoring. However, the time series data of an individual animal can be used by data assimilation (DA) techniques to perform an online (e.g. in real time) correction of the models as the animal grows.

DA is comprised of a set of statistically robust techniques that enable combining mathematical models, that describe the dynamics within a system, with measurements taken by monitoring the system's behavior over time. DA has the main objectives of: (a) correcting the model's state and parameters so that their estimates follow the measurements made, reducing model bias; (b) aggregating measurement information obtained from different sources into a mathematical model in order to obtain more accurate estimates of the system's actual state; (c) estimate variables that cannot be directly measured by sensors. Therefore, DA techniques are suitable for identifying the set of parameters associated with phenotypical dynamics of an animal, such as growth rate and mature weight, which in turn can improve individual animal's weight forecast. Studies indicate that individualizing the growth trajectories of animals can generate significant economic and environmental gains [1].

This study evaluates the performance of data assimilation algorithms for online correction of productivity forecast models of beef cattle, using data from automated animal weight sensors. The objective is to select model and algorithm pairs with high accuracy and robustness, in order to provide the information required for making management decisions that promote higher bioeconomic and environmental efficiency of beef cattle production systems.

1.1 Objective

The overall objective of the research is to determine whether it is possible to reduce the error of forecasts obtained by animal growth models through the use of data assimilation techniques in conjunction with animal weight collected through automated animal scales.

The specific objectives are:

- Evaluate the the error of forecasts obtained by the studied models without performing DA.
- Compare the errors of forecasts obtained from different associations between the studied DA techniques and animal growth models.
- Determine the association between the number of days of animal weight data that the DA techniques utilized and the error of the obtained forecasts.

1.2 Hypothesis

It is possible to reduce the error of forecasts obtained by animal growth models using DA techniques in conjunction with animal weight data collected through automatic animal weight sensors.

Chapter 2

Literature review

2.1 Beef production management

Management of livestock operations involves planning, implementation, monitoring and control in a continuous cycle [23]. According to Parker, Shadbolt, and Gray [44], the planning phase requires the establishment of target levels of performance, which should be consistent with the technology and course of actions projected to implement.

Computer simulation may aid managers to compare prospects besides providing an objective way to define expected levels of performance for each scenario in livestock production. Several predictive computer models have been successfully delivered as planning tools for livestock production either through scenario analysis [20, 25] or optimization [3, 52, 16]. They may provide the best expected trajectories of indicators of a system performance prior to the implementation of the plan but they will rarely be a satisfactory match to the actual systems behavior, so monitoring and controlling are needed [44].

Deviations of the system in relation to model forecasts may stem from several factors, including: differences between the planned and implemented controllable input variables (drivers); uncertainties related to uncontrollable or unobservable model parameters, drivers or other factors, and; structural simplifications of the system in the modeling process. As a result, systems must be continuously monitored to allow the manager to take corrective management (control) actions, making sure that the expectations of a strategy outcome get updated as new evidence arrives. Control actions should be performed whenever deviations from target expectations are identified and expected benefits exceed control costs [22].

At the operational level, the main controllable drivers for beef animals in feedlots are the quantity of feed offered and feed energy concentration. For maximum performance, feed is usually offered for ad libitum levels of intake, ensuring minimum orts (up to 5% of feed offered in total mixed rations; [26]). Individual feed intake is difficult to estimate on pasture and is rarely measured in commercial operations, although technology

is already available and is often used in research experiments [13]. Large feedlot operations may have pen (batch) estimates of dry matter intake retrieved from automated feeding wagons, but, in the general case, there are no accurate feed intake (dry matter intake) estimates. Dealing with uncertainties and corrective actions are usually costly. Therefore, early access to accurate information will contribute towards the objectives [8].

2.2 Animal Growth Modeling

Conversion of feed to body weight may vary according to the concentration of energy in tissue gain (RE, retained energy concentration) and, to a lesser extent, the energy required for maintenance [40]. RE varies with the proportion of fat and protein in gain. Fat is more energetically dense (ca. 39.6 kJ.g-1) than protein (ca. 23.8 kJ.g-1) and fat-free body mass (ca. 5.24 kJ.kg-1). Therefore, higher proportions of fat in new tissue deposition will result in lower rates of liveweight gain at a given energy input rate.

Fat proportion in liveweight gain is strongly associated with the maturity of the animal. At a given weight, animals with genetics for higher mature liveweight (LWmat) will have lower proportions of fat in gain. Maturity can be represented in several ways, some explicit and others implicit. For instance, Oltjen *et al.* [43] uses an implicit maximum protein mass that can be derived from its parameters [4], while NASEM [40] uses the equivalent empty body weight (EBW) to represent the degree of maturity.

Although liveweight gain decreases as the animal matures, monitoring liveweight alone does not allow to precisely distinguish animals with higher LWmat and standard intake from animals with higher intake and lower LWmat (associated to higher fat deposition) in the short term. Fortunately, LWmat has known biological bounds (e.g. a well fed mature male Nellore bull would weight around 850 and 1300 kg) [9] is usually associated to a model parameter. Therefore, it would be possible to apply population covariances between feed intake, fat proportion and liveweight gain so that those variables are kept within the expected range.

Mechanistic models of animal growth (e.g. Oltjen *et al.* [43], NASEM [40]) assume energy conservation. In general, energy input (intake) rate (EIR) is calculated from feed energy concentration and feed intake. Energy available for growth is estimated by subtracting energy requirements for maintenance from EIR. The usual way to apply those models is to estimate performance from energy intake. However, the system is defined as a bijective function, so it is possible to backcalculate energy and feed intake [6, 17, 11]

The partition of the energy above the requirement for maintenance metabolic efficiency for growth, work and other productive activities vary [40]. The limitations of this method are related, nevertheless, to variations in efficiency of feed conversion among

animals, which may relate to the partition of energy to protein or fat.

Feed energy concentration is usually a major driver of animal performance as highly fibrous diets reduce feed intake [40], besides providing lower energy per unit of mass. Typical energy concentrations of feedstuffs are available from several tables of nutritional values (e.g. NASEM [40], Valadares Filho *et al.* [56]). After established the price of purchased animal, growth rate, feed (dry matter) intake and cost of feed (dry matter) are the most important variables associated with profitability for the stocking and finishing phases of livestock operations [37].

2.3 Data Assimilation

The idea behind DA is to combine models that describe the dynamics within a system with observations of this system, where both carry information and have finite errors. As such, intuitively, aggregating them should increase the information about the system, improving it's state estimates [34]. DA algorithms are stochastic frameworks, where the estimates from the model and observations are represented as random variables, which are usually described by its mean and variance.

Most commonly, DA algorithms are executed iteratively for each new measurement collected. Each iteration is divided into two main stages, a time update and a measurement update stages. In the time update step, the model is used to estimate the state of the system in the next time step. Due to the uncertainties associated to the model and its drivers, the time update step increases the variance of the state estimate. The mean and variance of the state estimate at the end of the time update stage are called prior estimate. In the measurement update step, the algorithm incorporates the new measurements into the prior, which in turn reduces the variance of the estimate. The mean and variance at the end of the measurement update step are called posterior estimate. [33]

The Kalman Filter (KF) [30] is one of the most widely known DA algorithms. An assumption of the KF is that the estimates of the state of the system and measurements are random variables with normal distribution that are represented by a mean vector and a covariance matrix. The mathematical model is represented by two matrices, F and H. F is known as the state transition model, and is used to compute the mean of the prior $x_t = Fx_{t-1}$ during the time update stage. The H matrix, known as the observation model, is used to project the state vector into the observation's vector space during the measurement update stage. This is essential to compute the difference (residual) between the mean of the prior and the current measurement, whose dimensions and measurement units do not always match. Although KF is an optimal estimator for linear models, nonlinear models break its assumption that the estimate of the model's state

will be normally distributed. In order to solve the problem, KF variations for nonlinear estimation were developed, such as the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF) [60].

The EKF works through the linearization of the model, expressed through the Jacobian matrix, which is then used in the KF linear transformations. However, EKF still suffers from some limitations, such as: (a) because the EKF only uses first order terms of the Taylor series expansion, it often introduces large errors, and the resulting estimates may be inaccurate [58]; (b) not all models have a Jacobian, so they cannot be linearized; (c) the Jacobian calculation can be difficult and is susceptible to errors [29].

The UKF originates from the idea that it is easier to approximate the distribution of a random variable that undergoes a nonlinear transformation rather than approximate the nonlinear function. To solve the problem of nonlinearity, UKF uses the sigma points approach, which are a set of points deterministically computed in order to accurately represent the distribution of a normal random variable. The model is composed by two functions, f and h, which are analogous to the F and H transformation matrices, respectively. A set of sigma points $\mathcal{X}_x$ are computed for representing the current system's state estimate x. When $\mathcal{X}_x$ undergoes a nonlinear transformation, either by f or h, the resulting sigma points accurately captures true mean and covariance up to the 3rd order (Taylor series expansion) for any nonlinearity [60]. The UKF is superior to the EKF due to: (a) the model does not need to be written in matrix form; (b) it is not necessary to compute the Jacobian; (c) the UKF produces better estimates than the EKF for functions that have strong nonlinearity [60].

Particle Filters (PF), also known as sequential Monte Carlo methods, are another widely used approach for nonlinear estimation. While previous nonlinear approaches rely on Gaussian approximations, PFs do not require Gaussian assumptions. The underlying concept in PFs is that it is possible to approximate the distribution of any given function by drawing repeated random samples from it, a process known as Perfect Monte Carlo Simulation [60]. In PFs a set of points, called particles, are randomly sampled from a probability distribution. Similarly to the UKF, each particle undergoes nonlinear transformations by f and h. The distribution of particles after undergoing the transformation is called proposal distribution. These particles are filtered by assigning a weight associated with its probabilistic importance (likelihood) given the observed measurements. Particles with lower likelihood are then eliminated and particles of great importance undergo a resampling process. Due to the fact that Monte Carlo approximations usually require large amounts of samples to produce good approximations [18], PFs usually have a much higher computational cost than KF variants.

The PF algorithm does, however, suffer from a well known degeneracy problem, which happens if a sample with high importance weight gets multiplied many times. When this happens, the particles have a tendency to collapse into a single point, causing the

PF estimates to deviate from measurements. Thus, the choice of proposal distribution is critical for avoiding the degeneracy of estimates. Van Der Merwe *et al.* [58] tackle this problem using the UKF to filter each particle in order to generate a more accurate proposal distribution. This technique is called the Unscented Particle Filter (UPF), and it performs much better than the PF or KF variants in the examples shown at Van Der Merwe *et al.* [57].

2.4 Data assimilation in animal growth

The KF had one of its first applications in the space navigation system at the Apollo Project in 1960 [24]. Ever since then, DA techniques have spread across several fields of knowledge. DA is prominently used in object tracking systems, such as radars, GPS (global positioning system), missile navigation, among others [24]. Many other areas have also found uses for data assimilation, such as atmospheric modeling [31, 63], oceanography [21], medicine [49], ecology [36], agriculture [62], among many others.

Oltjen and Owens [42] conducted a study on data assimilation in bovine growth models using an empirical Bayesian derivation of KF (EBKF). The study evaluates the performance of EBKF by comparing three different methods of bovine growth forecasting: (a) growth forecasting given by a model, without DA; (b) growth forecasting using the EBKF to incorporate measurements and adjust the parameters of food consumption (y1) and weight gain (y2); (c) growth forecasting using the least squares procedure to incorporate measurements and adjust the same y1 and y2 parameters. The results show that the projections made using EBKF present a smaller error when compared to the other two methods and require a shorter period of observations to present good estimates.

In view of the scarcity of literature on DA applications in cattle growth, and no literature on its application in conjunction with automatic scales, the aim of this work is to contribute to the advancement of the state of the art in animal growth estimation and forecasting.

Chapter 3

Material and methods

This chapter is organized as follows: 3.1 explains some background about space-state models, needed in order to understand the rest of the chapter; in 3.2 explains the Richards', Oltjen's and NASEM's models, which were used in this study; 3.3 explains how the data used in this study was acquired, and; 3.4 explains the DA algorithms used in this study, as well as the methodology used in order to apply them to each model.

3.1 Discrete Space-State models

A space-state model is a first order Markov process that can be divided into a state transition model and a measurement model, as shown below:

$$\begin{aligned} p(x_t|x_{t-1}) \\ p(y_t|x_t) \end{aligned} \tag{3.1}$$

where x_t is the state of the model at time t and y_t the output observations.

For the models used in this study, the space-state model can be defined as:

$$\begin{aligned} x_t &= f(x_{t-1}) + v_{t-1} \\ y_t &= h(u_t, x_t) + n_t \end{aligned} \tag{3.2}$$

where f is the state transition model, and h is the measurement model. f updates the model's state from time $t-1$ to t, u_t are the input observations, v_t the process noise and n_t the measurement noise. h computes the output observations of the model, a set of values that correspond directly to the measurements taken from the system.

3.2 Animal Growth Models

In this study, we compare three different models in order to measure their performance with and without applying DA. These models were implemented using an

inhouse Python framework for modeling and simulating dynamic systems.

3.2.1 Richards

The Richards function is an empirical model widely used to represent the growth of animals (e.g., Brown, Fitzhugh Jr, and Cartwright [10] and Fitzhugh [19]). The literature mentions alternative forms of this equation, thus we specify the functional form used in this study. In Fitzhugh [19], the Richards model is presented as a generic function that can be reduced to several other widely used growth functions, including the monomolecular, logistic, Gompertz and Bertalanffy models. According to the author, the equation in its differential form is given by:

$$\frac{dY}{dt} = MkY\left[\left(\frac{Y}{A}\right)^{\frac{-1}{M}} - 1\right] \tag{3.3}$$

in which,

- $\frac{dY}{dt}$ is the instantaneous animal growth rate,
- Y is the animal's liveweight,
- k is the animal's maturity index,
- M is the inflection parameter,
- A is the asymptotic value for the animals' weight.

The Richards function was used in this study because it is one of the most flexible functional forms of representing the continuous growth of animals [55], and because the biological interpretation of its parameters is relatively straightforward. One of the difficulties in using the Richards function, however, is parameter identification [53, 48].

3.2.2 Oltjen

The bovine growth model of Oltjen *et al.* [41] and Sainz *et al.* [47] is a dynamic model based on a system of three differential equations which compute the mass of DNA (eq. 3.4), protein (eq. 3.5) and body fat (eq. 3.6).

$$\frac{dDNA}{dt} = ND \times kD \times (DNA_{max} - DNA) \tag{3.4}$$

$$\frac{dProt}{dt} = NP \times kPs \times DNA^{0.73} - kPd \times P^{0.73} \tag{3.5}$$

$$\frac{dFat}{dt} = \frac{(FI - \frac{M}{NEm}) \times NE\_g - \frac{dProt}{dt} \times EP}{EF} \tag{3.6}$$

The parameter kD from Eq. (3.4) defines the reference the DNA rate of accretion, and the auxiliary variable ND defines the effect of the feed energy intake on its variation (Eq. (3.8)). In Eq. (3.5), FI is the feed intake; M is the net energy for maintenance; NE_g is the feed net energy concentration for liveweight gain; EF is the energy concentration of fat tissue, and; EP is the energy concentration of protein tissue.

The effect of energy intake is modeled through the ratio of the rate of metabolizable energy intake (MEI) of the animal and its reference energy intake rate (MEI_{ref}), estimated from its physiological condition.

$$EII = \frac{MEI}{MEI_{ref}} \tag{3.7}$$

$$ND = 0.7 + 1.7 \times EII \tag{3.8}$$

$$MEI_{ref} = EBW^{0.73} \times \left(MEI_0 - MEI_1 \times \frac{EBW}{EBWM} \right) \tag{3.9}$$

In Eq. (3.9) we compute the reference ingestion of metabolizable energy. MEI_0 and MEI_1 are linear regression intercepts, with original values of 0.480 and 0.2615, respectively. $EBWM$ is the mature empty body weight.

The rate of protein synthesis, Eq. (3.10), is defined as an affine function of DNA mass raised to the power of 0.73. The parameter kPs is the protein synthesis reference rate per unit of $\text{DNA}^{0.73}$, and NP is the effect of energy intake on protein synthesis. Protein degradation, Eq. (3.11), is a function of the body protein mass raised to the power of 0.73 and kPd is the protein synthesis reference rate per unit of $P^{0.73}$. The model assumes that protein degradation is insensitive to nutrition.

$$Psyn = NP \times kPs \times DNA^{0.73} \tag{3.10}$$

$$Pdeg = kPd \times Prot^{0.73} \tag{3.11}$$

$$NP = 0.83 + \frac{0.2 \times EII}{0.15 + EII} \tag{3.12}$$

The animal's empty body weight (EBW) is estimated as the sum of the animal's fat mass and lean mass. The lean mass is estimated from the protein mass, considering that the protein constitutes 22.01% of the lean mass, as described in the equation

(3.13)

$$EBW = F + \frac{P}{0.2201} \tag{3.13}$$

The animal's shrunk body weight (SBW) is estimated as a proportion of EBW (3.14). The liveweight, in turn, is computed as a proportion of SBW (3.15).

$$SBW = \frac{EBW}{0.91} \tag{3.14}$$

$$LW = \frac{SBW}{0.96} \tag{3.15}$$

3.2.3 NASEM

Since 1944, the American *National Research Council* has published a guide on the nutritional requirements of beef cattle. Now incorporated into the American *National Academy of Sciences, Engineering and Medicine*, NASEM's model [40] is probably the most widely used for formulating beef cattle diets and performance estimates in Brazil and in the world. It has been fully or partially incorporated in several decision support systems used in Brazil, such as RLM, BR-Corte, Invernada, and others.

In NASEM's model the rate of body weight gain is based on the amount of energy retained (*RE, Mcal/day*), according to Eq. (3.16).

$$EBG = 12.341 \times EQEBW^{-0.6837} \times RE^{0.9116} \tag{3.16}$$

The retained energy is obtained by the Eq. (3.17) as the difference between the amount of energy ingested by the animal and the energy costs for maintenance and activity (*Maint*, (3.18)). Dry matter intake (DMI) is the daily consumption of feed dry matter (*kg/day*), obtained by Eq. (3.19). NE_m and NE_g are respectively the concentration of net energy for maintenance and gain of body tissues that can be obtained from feed composition databases or estimated from feed digestibility according to equations available at [40].

$$RE = \frac{DMI - Maint}{NE_m} \times NE_g \tag{3.17}$$

$$Maint = \alpha \times SBW^{0.75} \tag{3.18}$$

$$DMI = (0.1493 \times NE_m - 0.046 \times NE_m^2 - 0.0196) \times SBW^{0.75} \tag{3.19}$$

NASEM is originally a static model, meaning it does not allow temporal evolution of variables. The diets are formulated and evaluated considering the average weight of the animals in a given evaluation period [40]. In this study, NASEM's model was adapted into a dynamic model by adding a state variable EBW that maintains the empty body weight of the animal. The differential equation associated with the state variable EBW is the EBG, given by Eq. (3.16).

The DMI calculation, Eq. (3.19), also required modification, since in it's original form it does not depend on the animal's maturity. Therefore, we changed NASEM's model to compute DMI using Oltjen's [43] Eq. (3.9), which computes the reference ingestion of metabolizable energy, and then compute the DMI using Eq. (3.20).

$$DMI = \frac{MEI\_ref}{MEC} \tag{3.20}$$

where MEC is the diet's metabolizable energy concentration.

3.3 Data Acquisition and Treatment

Estimates of animal weights were provided by the company Coimma. These estimates are made from measurements carried out by BalPass ® automatic scales. These scales are installed in locations where animals pass through routinely. Each time an animal walks across the scale, the animal is identified by a radio frequency identification (RFID) earring, and its weight is recorded.

The data collected by the scales is processed to estimate a daily reference weight for each animal, from which it is possible to estimate its growth over time. Since the weighing process does not require human intervention, it is common for invalid weighings to occur when an animal does not go through the scale as expected (e.g. jumping, or more than one animal at the same time). In addition, the animals' weight varies throughout the day according to the filling of its digestive tract [15]. To solve this, invalid weights are detected and removed, after which the reference weight is calculated as the average of the weight of each animal over each day.

The final dataset consists of the daily reference weights of 151 male animals, predominantly of the race Angus, in confinement. The time span during which each animal was observed varies, ranging from 31 to 146 days.

3.4 Data Assimilation algorithms

The DA techniques used in this study were: the Unscented Kalman Filter (UKF), because it provides better overall results [60] than those of EKF, does not require individual model linearization and has low computational cost; the Particle Filter (PF),

because it is able to approximate non-Gaussian distributions; and the Unscented Particle Filter (UPF) because it generally performs better than the standard PF and avoids the degeneracy problem, although the UPF has a much higher computational cost than the PF if the same number of particles is used.

Each filter was implemented as a Python class, which allows applying them for different models through a standardized code interface.

3.4.1 Unscented Kalman Filter

The UKF [27] is based on the Unscented Transform (UT) technique, which is used to compute the statistics of a random variable x when it undergoes a non-linear transformation $y = g(x)$. The UT is capable of approximating the average and covariance up to third order with good accuracy for any non-linearity [60]. For this, $2n_x + 1$ sigma points are chosen deterministically, where n_x is the dimension of a Gaussian random variable x with variance P_x, with corresponding weights. Each sigma point is represented by a vector $\mathcal{X}_i$ and weighted by a scalar W_i, and is calculated as follows:

$$\begin{aligned}
&\mathcal{X}_0 = \bar{x} \\
&\mathcal{X}_i = \bar{x} + (\sqrt{n_x + \lambda}P_x)_i, \quad i = 1, ..., n_x \\
&\mathcal{X}_i = \bar{x} - (\sqrt{n_x + \lambda}P_x)_i, \quad i = n_x + 1, ..., 2n_x \\
&W_0^{(m)} = \lambda/(n_x + \lambda) \\
&W_0^{(c)} = \lambda/(n_x + \lambda) + (1 - \alpha^2 + \beta) \\
&W_0^{(m)} = W_0^{(c)} = 1/\{2(n_x + \lambda)\}, \quad i = n_x + 1, ..., 2n_x
\end{aligned} \tag{3.21}$$

where $\lambda = \alpha^2(n_x + \kappa) - n_x$ is a scaling parameter. α determines the "spread" of the sigma points around $\bar{x}$. κ is a secondary scaling parameter and β is used to incorporate prior knowledge about the distribution of x. Each sigma point is then propagated through a non-linear function,

$$\mathcal{Y} = g(\mathcal{X}), \quad i = 0, ..., 2n_x \tag{3.22}$$

The mean and covariance of y are calculated as follows:

$$\bar{y} \approx \sum_{i=0}^{2n_x} W_i^{(m)\mathcal{Y}_i} \tag{3.23}$$

$$P_y \approx \sum_{i=0}^{2n_x} \{\mathcal{Y} - \bar{y}\}\{\mathcal{Y}_i - \bar{y}\}^T \tag{3.24}$$

Unscented Kalman Filter Algorithm

The UKF is an iterative algorithm that uses the UT to generate estimates that minimize the mean squared error [28]. The process and measurement noises related to the system studied in this work are purely additive. Therefore, we use a simplification of the UKF described by Julier and Uhlmann [28] and Van Der Merwe *et al.* [58]. The simplification of the UKF algorithm is described below:

1. Initialization

$$\bar{x}_0 = E[x_0]$$
$$P_0 = E[(x_0 - \bar{x_0})(x_0 - \bar{x_0})^T]$$

2. For $t \in \{1, ..., \infty\}$

   (a) Compute *sigma points*:

$$\mathcal{X}_{t-1} = [\bar{x}_{t-1} \;\; \bar{x}_{t-1} \pm \sqrt{(n_x + \lambda)P_{t-1}}]$$

   (b) Time update

$$\mathcal{X}_{t|t-1} = f(\mathcal{X}_{t-1})$$
$$\bar{x}_{t|t-1} = \sum_{i=0}^{2n_x} W_i^{(m)} \mathcal{X}_{i,t|t-1}$$
$$P_{t|t-1} = \sum_{i=0}^{2n_x} W_i^{(c)} [\mathcal{X}_{i,t|t-1} - \bar{x}_{t|t-1}][\mathcal{X}_{i,t|t-1} - \bar{x}_{t|t-1}]^T + Q$$
$$\mathcal{Y}_{t|t-1} = h(\mathcal{X}_{t|t-1})$$
$$\bar{y}_{t|t-1} = \sum_{i=0}^{2n_x} W_i^{(m)} \mathcal{Y}_{i,t|t-1}$$

   (c) Measurement update

$$P_{\tilde{y}_t \tilde{y}_t} = \sum_{i=0}^{2n_x} W_i^{(c)} [\mathcal{Y}_{i,t|t-1} - \bar{y}_{t|t-1}][\mathcal{Y}_{i,t|t-1} - \bar{y}_{t|t-1}]^T + R$$
$$P_{x_t y_t} = \sum_{i=0}^{2n_x} W_i^{(c)} [\mathcal{X}_{i,t|t-1} - \bar{x}_{t|t-1}][\mathcal{Y}_{i,t|t-1} - \bar{y}_{t|t-1}]^T$$
$$K_t = P_{x_t y_t} P_{\tilde{y}_t \tilde{y}_t}^{-1}$$
$$\bar{x}_t = \bar{x}_{t|t-1} + K_t(y_t - \bar{y}_{t|t-1})$$
$$P_t = P_{t|t-1} - K_t P_{\tilde{y}_t \tilde{y}_t} K_t^T$$

where Q is the model's noise covariance matrix, R is the measurement covariance matrix and K is called Kalman Gain. Note that the simplification of the UKF assumes that P, Q and R covariance matrices are independent.

3.4.2 Particle Filter

The Particle Filter is a Monte Carlo simulation method which, unlike the UKF, do not assume a fixed shape of densities [46]. Monte Carlo simulations use sets of randomly drawn samples to map integral problems into discrete sums. Thus, it allows approximating the expectation of a set of samples $x^{(i)}$ from a distribution $p(x_t|y_t)$ that undergo a nonlinear transformation $g(x^{(i)})$, as follows:

$$\mathbb{E}[g(x_t)] = \int g(x_t)p(x_t|y_t)dx_t \tag{3.25}$$

$$\overline{\mathbb{E}[g(x)]} = \frac{1}{N}\sum_{i=1}^{N} g(x^{(i)}) \tag{3.26}$$

where the samples $x^{(i)}$ are assumed to be independent and identically distributed (i.i.d.) and $\overline{\mathbb{E}[g(x)]} \xrightarrow{N\to\infty} \mathbb{E}[g(x)]$ if the variance of $g(x)$ is bounded [58].

Sequential Importance Sampling

It is often impossible to sample from $p(x_t|y_t)$ directly. To circumvent this, we can use the sequential importance sampling (SIS) method, which is performed by drawing the samples from a proposal distribution $q(x_t|y_t)$, generally defined as:

$$q(x_t|y_t) = q(x_{t-1}|y_{t-1})q(x_t|x_{t-1}, y_t) \tag{3.27}$$

The most commonly used distribution for the PF algorithm is the transition prior, $f(x_{t-1})$ [60]. Since the proposal distribution is usually not i.i.d., we must also assign an importance weight term w_t to Eq. (3.26),

$$\overline{\mathbb{E}[g(x_t)]} = \frac{1}{N}\sum_{i=1}^{N} g(x_t^{(i)})w_t(x_t^{(i)}) \tag{3.28}$$

where

$$w_t = w_{t-1}\frac{p(y_t|x_t)p(x_t|x_t-1)}{q(x_t|x_{t-1}, y_t)} \tag{3.29}$$

Resampling Stage

To avoid degeneracy of particles in the SIS method, in which the variance of the importance weights increases stochastically over time [58], we can apply a resampling stage. In this stage, we eliminate particles with lower importance weight and multiply those with high importance weights. In this study, we use the systematic sampling procedure, introduced by Kitagawa [32] and Crisan and Doucet [14], implemented in python by

Labbe [33]. In this method, we first create a sampling space by performing the cumulative sum of the normalized importance weights. We then divide the resampling space into N divisions and choose a random offset from which to sample from all divisions, ensuring each sample is 1/N appart.

Particle Filter Algorithm

Here is presented the pseudo-code algorithm for the Particle Filter, adapted from Wan and Van Der Merwe [60].

1. Initialization For $i \in \{1, ..., N\}$, draw the state $x_0^{(i)}$ from the prior $p(x_0)$
2. For $t \in \{1, ..., \infty\}$
   (a) Importance Sampling (Time update)
      - Propagate particles through model function $x_{t|t-1} = f(x_{t-1})$
      - Evaluate w_t
      - Normalize importance weights: $\tilde{w_t} = w_t \left[\sum_{j=1}^{N} w_t^{(j)}\right]^{-1}$

   (b) Resample Step (a.k.a. Selection Step or Measurement update)
      - Apply systematic resampling to the particles $x_{t|t-1}$
      - For $i \in \{1, ..., N\}$, set $w_t^{(i)} = \frac{1}{N}$

3.4.3 Unscented Particle Filter

The choice of proposal distribution is a critical design choice of the PF. If the proposal distribution does not accurately represent the system's state distribution, a few particles might end up getting a disproportionately higher importance weight, leading to degeneracy of the filter.

As mentioned in subsubsection 3.4.2, in Sequential Importance Sampling, the transition prior $f(x_{t-1})$ is the most popular choice for a proposal distribution. However, the proposal distribution can be improved if we incorporate to it the newest available observations. In order to accomplish this, instead of sampling directly from f_{t-1}, the UPF samples each particle using the UKF algorithm, which moves the proposal distribution to a region of higher likelihood [57].

3.4.4 Dual Estimation

Dual estimation is the problem of estimating both the state of a dynamic system and the parameters that compose the model that describes the system's dynamics [61].

The approach used for solving this problem in this study is concatenating the model's state and the parameters to be estimated into a single joint state vector [60]. The reasoning behind this is that the UKF, PF and UPF essentially view the models as black box functions. In other words, these filters have no knowledge of what happens inside the model's functions, besides their inputs and outputs.

Therefore, it is of no importance to the filter if an element of the joint state vector is a state or parameter, and both can be estimated simultaneously as long as the covariance matrices P, Q and R are initialized properly (see subsection 3.4.6).

3.4.5 Growth Model Calibration

Initial Calibration

The initial model calibration heavily impacts the performance of the models, both with and without DA. Thus, it is first necessary to perform an initial calibration to guarantee that each model performs its best, given the available measurements. To do this, we use the Nelder-Mead optimization algorithm [35], implemented in Python's SciPy [59] package, to minimize the mean squared error between the models' forecasts and measurements (Eq. (3.30))

$$mse = \frac{1}{I}\sum_{i=0}^{I}\frac{1}{T}\sum_{t=0}^{T}(h(f(x_{t-1}^{(i)})) - z_t^{(i)})^2 \tag{3.30}$$

Calibrated Parameters

Given the relatively small size of the dataset used in this study, as well as the lack of any other variables besides the daily reference weight of each animal, due to the existence of collinearity in the parameters, not all of them can be calibrated simultaneously. Thus, for each model were selected one parameter related to animal growth rate and another related to maximum expected weight at maturity. It is noted that, although these parameters have similar roles across the three models, they do not operate identically. After the initial calibration, both of the selected parameters of each model are further adjusted by the DA filters.
For the Richards model, the following parameters are calibrated:

- k: animal maturity index
- A: the asymptotic value for the animals' final weight when t grows indefinitely

For Oltjen's model [43], the calibrated parameters are:

- kD: reference speed for increasing the DNA mass

- DNA_{max}: maximum amount of DNA reached by the animal in maturity

Finally, for NASEM's [40] model, the calibrated parameters are:

- MEI: intercept of the metabolizable energy intake
- EBW_{max}: maximum empty body weight reached by the animal in maturity

For Oltjen's and NASEM's models, all remaining parameters that were not selected for calibration were kept with the original calibration for cattle growth available from the literature.

The Richard's model, as generic growth model, does originally have a calibration for cattle growth. Therefore, the variable M, the inflection parameter, had to be adjusted during the initial model calibration, together with the k and A parameters. However, in order to keep the number of adjustable parameters of the three models the same during filtering, M was not adjusted by the DA filters.

Joint State Vector

To perform DA, it is necessary to define the state transition function f and measurements function h for each model, required by the filtering algorithms explained in section 2.3. The f function receives as parameters the joint state vector x_{t-1} and estimates x_t. The h function receives x_t and estimates the liveweight of the animal, which will be used by the filters to compute the difference between the estimates and measurements.

The joint state vector is composed by each model's respective state, as well as the parameters that are going to be adjusted by the filter. In order for the DA algorithms to work, all the models' states must be present in the joint state vector. That includes liveweight (LW) for Richard's model; DNA, protein (Prot) and fat masses for Oltjen's [43] model, and; empty body weight, for NASEM [40].

Besides the state variables, we also wish further adjust the calibrated parameters of growth rate and maximum weight at maturity to better match the growth characteristics of each individual animal. To do this, we include the same parameters used in the initial calibration to each model's respective joint state vector.

3.4.6 Hyperparameter Optimization

As explained in subsection 3.4.1, subsection 3.4.2 and subsection 3.4.3, all the filters used in this study have three essential hyperparameters, which determine how the filters behave: the P_0, Q and R covariance matrices. The P_0 covariance matrix represents the initial uncertainty of the joint state vector of the system. The Q covariance matrix represents the uncertainty added to the joint state vector by in the time update step,

and it corresponds to the v_{t-1} error term in Eq. 3.2 The R covariance matrix represents the uncertainty observations, and it corresponds to the n_t error term in Eq. 3.2. The theoretical meaning of P_0, Q and R are common between the UKF, PF and UPF, although, in practice, their values for obtaining optimal results, while applying each filter to any particular system, may vary.

Error Function

In order to optimize the hyperparameters of the filters for each model, it is first necessary to define a function which objectively measures the filter's performance. In this study, we subjectively define an optimal result as the forecast the best matches future measurements. To do so, we use a n-step ahead error function, which measures the mean absolute error between the estimated joint state model forecast n-steps into the future at each time step and each measurement, as shown in Eq. (3.31).

$$error = \frac{1}{I}\sum_{i=0}^{I}\frac{1}{T}\sum_{t=n}^{T}|h(f^n(x_{t-n}^{(i)})) - z_t^{(i)}| \tag{3.31}$$

where f^n is the state transition function applied n times, h is the measurement function, $x_t^{(i)}$ is the DA filter estimate of the state of the system for an animal i at time t, $z_t^{(i)}$ are the observations for an animal i at time t, and I is the number of animals. We opted to use the absolute error, instead of the more usual squared error, because at the beginning of the assimilation process the n-step error tends to be much larger, due to the parameters not yet being calibrated. Thus, the optimization algorithm prioritizes adjusting the model in the beginning of the assimilation, leading the filters to perform more abrupt changes in the model's parameters, at the expense of making less accurate forecasts after a more substantial amount of measurements get evaluated and the model's parameters converge. Squaring the error further increases this problem.

Function Minimization

The hyperparameters for the DA filters have high dimensionality and the function has many local minima. To perform this task, we use an implementation of the Particle Swarm Optimization (PSO) algorithm [50], available in Python's Pyswarms package [39], which is good at escaping local minima.

In order to obtain a good result, however, the PSO needs to compute the error function thousands of times. Therefore, since the PF and UPF are already very computationally heavy algorithms, it is very impractical to use them to minimize the error function. Since the UKF is a much faster algorithm, and the covariance matrices equivalent statistical meaning in all three filters, we use the UKF to minimize the error function. It is important to note that this affects the comparison of results between the

filters, since the matrices have a calibration bias towards the UKF.

Initializing the R Matrix

The measurements used for this study refer to a single variable, the animal weight. Therefore, R has only one element which corresponds to the variance of the measurements. Since this variance depends on biological characteristics of the animals, as well as their individual behaviour, and does not depend on the accuracy of the measurement instrument alone, it has to be numerically approximated. To do so, we do a first pass, using the UKF filter, for each animal with rough guesses of the hyperparameters, in order to acquire an initial estimate of the system's state at each time. Then, we compute the variance of the error between the filter's estimates and the animal weight measurements. Since this method does not require forecasting the model many steps ahead, it is not as sensitive to hyperparameter calibration, and, as long as the filters don't overfit the measurement data, the approximation of R is not greatly affected by the initial hyperparameter guesses.

Initializing the Q Matrix

For the models used in this study, we assume that the noise added by forecasting the system's state is independent for each variable. Thus, Q is defined as a diagonal matrix that contains the noise added by forecasting the joint state vector one step into the future.

With exception of the state variables, the parameters of the models used in this study are assumed to be inherent to the system. In other words, although each animal will have a set of parameters that best describe its growth, these parameters should not change as time progresses. Thus, it is logical to conclude that the state transition function should not add any uncertainty to these parameters. As such, instead of allowing the optimizer to optimize the whole Q matrix, we fixate the values of Q which correspond to the inherent parameters to zero.

The value of Q corresponding to the state variables, however, cannot be easily approximated. Thus, we opt to use PSO to find the values which minimize the error of Eq. (3.31)

Initializing the P_0 Matrix

The P_t matrix represents the noise in the joint state vector at time t. As the system evolves, the filter takes in measurements and updates the joint state vector and the P_t matrix. However, the initial value of P_0, before running a filter, must be informed.

Since the initial state of the model corresponds to the initial measurement, the value of P_0 for the state variables can be approximated numerically using the Monte

Carlo method. To do this, we sample from a function that determines the growth model's initial state from the animal's initial liveweight. We have the distribution R for the liveweight; therefore we can approximate the model's state distribution by sampling from the initialization function.

The values of P_0 corresponding to the parameters in the join state vector are more difficult to compute, and we chose to calibrate them by minimizing the error function in Eq. (3.31). However, if the optimizer is allowed to choose any values for the parameters' variances, it frequently prioritizes giving a high variance to a single parameter, instead of the filters to proportionally adjust all parameters. As a consequence, the high variance parameter can reach unrealistic values, such as an impossibly high mature weight or animal growth rate.

In order to solve this problem, we calibrate the model, a single parameter at a time, for each animal. This gives us how much a parameter changes in order to solely explain the differences in growth between each animal, from which we compute their standard deviations. If this value is used to form P_0, the filters tends to adjust all parameters proportionally. We then multiply the standard deviations by a scalar, which is, in turn, what is adjusted by the optimizer.

Finally, all covariances between each element in the join state vector were calibrated by the optimizer.

Chapter 4

Results and discussion

The results and discussion of this study are organized as follows: in section 4.1 are presented the parameters obtained by the initial model calibration; in section 4.2 are presented the covariance matrices produced in the hyperparameter optimization process, and; in section 4.3 are presented the results obtained for dual estimation and model forecasting from the DA process for each model.

4.1 Initial Calibration

The values of the parameters after the initial calibration for each model are:

Richard's Model:

$$k = 0.006, \quad A = 907, \quad M = 15$$

Note that the value of M is not adjusted by the filters, as explained in subsubsection 3.4.5.

Oltjen's Model:

$$kD = 0.0073, \quad DNA_{max} = 500$$

NASEM Model:

$$MEI = 0.496, \quad EBW_{max} = 903$$

4.2 Hyperparameter Calibration

As explained in subsection 3.4.6, we must inform the covariance matrices P_0, Q and R of each model to the filters. R 4.1 is a 1 by 1 matrix, which represents the noise

in the animal weight data, and is the same for all models.

$$R = 3.84e + 1 \quad kg^2 \tag{4.1}$$

In order optimize the covariance matrices, we use the UKF to produce the join state estimates of each model and use Eq. (3.31) to obtain the Mean Absolute Error of N-Step Ahead Forecasts (NSAF-MAE), with $n = 50$. The reason we used $n = 50$ is that, if the forecast is too short, observation biases pollute the NSAF-MAE too much, which in turn causes the optimizer to prioritize adjusting the model's state, rather than the parameters. However, if n is too large, we are left with a time window too small to compute the NSAF-MAE.

Since not all 151 animals had the same number of observations, with some having as few as 31 observations, we opted to remove those that had fewer than 130 observations, in order to standardize the analysis. This way, we can compute the NSAF-MAE from an observation window ranging from day 50 to day 130, for a sample of 81 animals.

In order to speed up the optimization process, we parallelized the PSO execution to run each particle in a different thread, with a total of 8 threads. Furthermore, we used a vectorized version of the UKF, which computes all animals simultaneously by concatenating the sigma points into a single multidimensional array. This reduced the optimization time of the covariance matrices from several hours to about 15 minutes per model, in a 4-core (8-thread) CPU.

4.2.1 Richards Model Covariance matrix

The estimated optimal covariance matrices $\overline{P_0}$ and $\overline{Q}$ for Richards' model are:

$$\overline{P_0} = \begin{bmatrix} 3.84e+1 & 6.47e+2 & -4.76e-3 \\ 6.47e+2 & 1.79e+4 & -1.17e-1 \\ -4.76e-3 & -1.17e-1 & 1.23e-6 \end{bmatrix} \tag{4.2}$$

$$\overline{Q} = \begin{bmatrix} 2.30e-1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{4.3}$$

4.2.2 Oltjen's Model Covariance matrix

The estimated optimal covariance matrices $\overline{P_0}$ and $\overline{Q}$ for Oltjen's model are:

$$\overline{P_0} = \begin{bmatrix} 6.5e-1 & 3.63e-3 & 6.29e-1 & -1.45e-3 & 6.82e+1 \\ 3.63e-3 & 1.57e+1 & -2.5e-2 & 6.98e-3 & 3.92e+2 \\ 6.29e-1 & -2.5e-2 & 2.70e+0 & -2.32e-3 & 3.03e+0 \\ -1.45e-3 & 6.98e-3 & -2.32e-3 & 6.22e-5 & 1.21e+0 \\ 6.82e+1 & 3.92e+2 & 3.03e+0 & 1.21e+0 & 4.31e+4 \end{bmatrix} \tag{4.4}$$

$$\overline{Q} = \begin{bmatrix} 7.61e-5 & 0 & 0 & 0 & 0 \\ 0 & 1.58e-3 & 0 & 0 & 0 \\ 0 & 0 & 2.13e+1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.5}$$

4.2.3 NASEM Model Covariance matrix

The estimated optimal covariance matrices $\overline{P}$ and $\overline{Q}$ for NASEM's model are:

$$\overline{P_0} = \begin{bmatrix} 2.81e+1 & -7.8e+2 & 3.42e-1 \\ -7.8e+2 & 4.12e+4 & -1.03e+1 \\ 3.42e-1 & -1.03e+1 & 4.21e-3 \end{bmatrix} \tag{4.6}$$

$$\overline{Q} = \begin{bmatrix} 7.33e-1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{4.7}$$

4.3 Data Assimilation

Below, we present the results obtained from each of the models using different filters. Although the filters estimate both the state and parameters of the model, we focus the results on the adjustment of parameters, since it impacts long term forecasts the most. The parameters of the models are assumed to be intrinsic to each animal, in other words, the parameter's value shouldn't change throughout the animals life. Therefore, in ideal scenarios, the value of the parameters adjusted by the filters should converge as more observations are taken in. In order to present the parameter convergence, we first compute the rolling standard deviation (RSD) of the parameter adjustment of each animal throughout the execution of each filter, with a rolling time window of 10 days.

Then, we compute the mean rolling standard deviation (MRSD) by averaging the RSD of all animals.

We also show the adjustment of model forecasts as the filter receives more observations, as well as parameter adjustments, using the same animal as an example for all the models and filters. These images alone cannot be used to evaluate the performance of the filters, since the results vary from animal to animal, but can visually represent some characteristics of each filter.

4.3.1 Richards Model

As we can see in Figure 4.1, the parameters A and k start with high MRSD, which then decreases in all filters. However, at around day 60, we note that MRSD of the UKF and UPF start increasing again, indicating that those parameters may not have stabilized, or that there were higher changes in the observations at the end of the growth period.

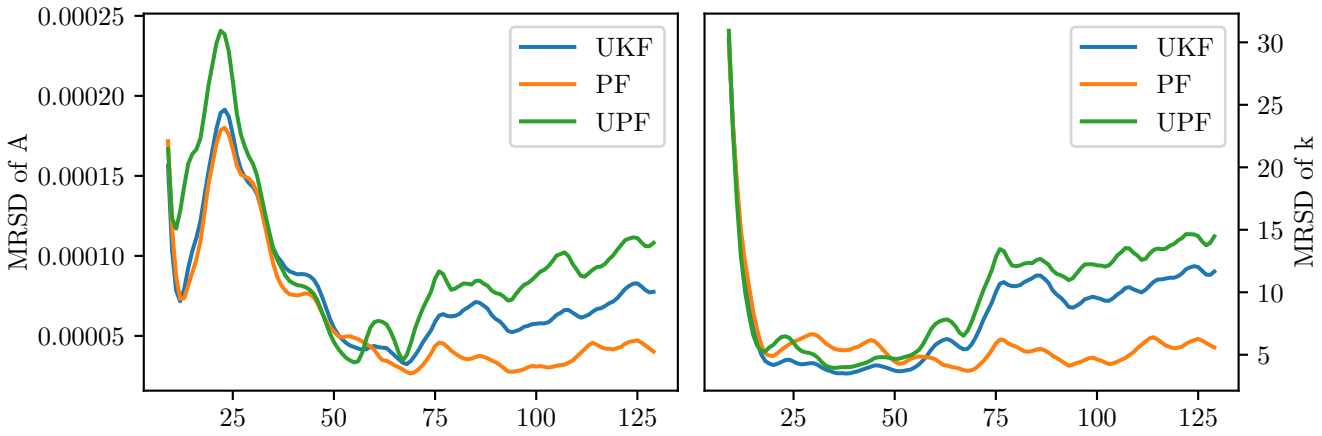
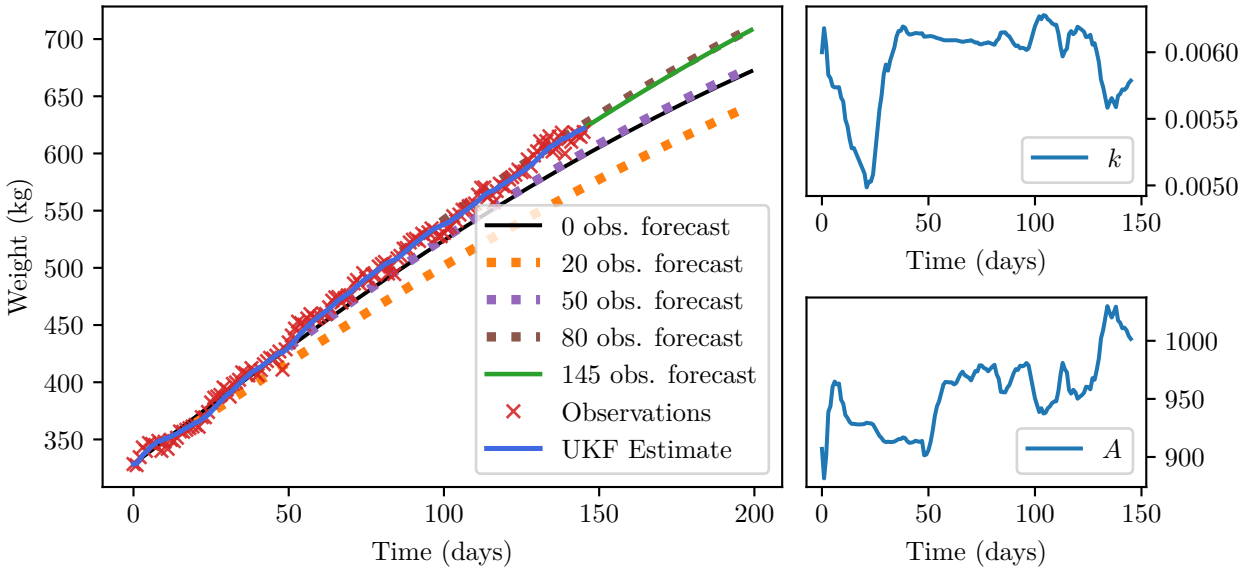


Figure 4.1: MRSD of Richards' model parameters with rolling window of 10 days

UKF

Figure 4.2 shows the DA of one animal using the UKF with Richards' model. The estimates of the filter successfully match the observations during the whole process, but the parameters, and consequently the forecasts, take some time to adjust. We also note that the parameters go through abrupt changes at times. This behavior matches with the MRSD of the UKF shown in Figure 4.1. The estimates at time 80, however, closely match the final observations.

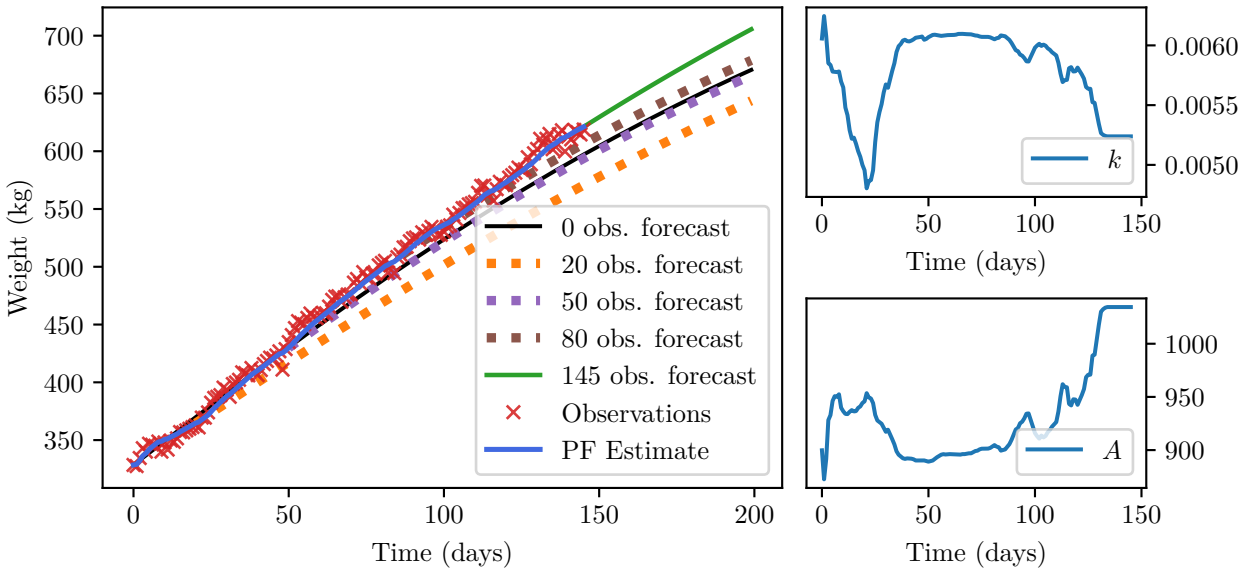


(a) Model forecast with UKF estimates. (b) Parameter adjustment during DA.

Figure 4.2: Data assimilation of Richards' model using the UKF

PF

In Figure 4.3 we see that the PF estimates correctly follow the observations, but the forecasts of the estimates have trouble matching future observations. We can also note that the parameters are adjusted less abruptly then in the UKF, but their overall trajectory is similar.

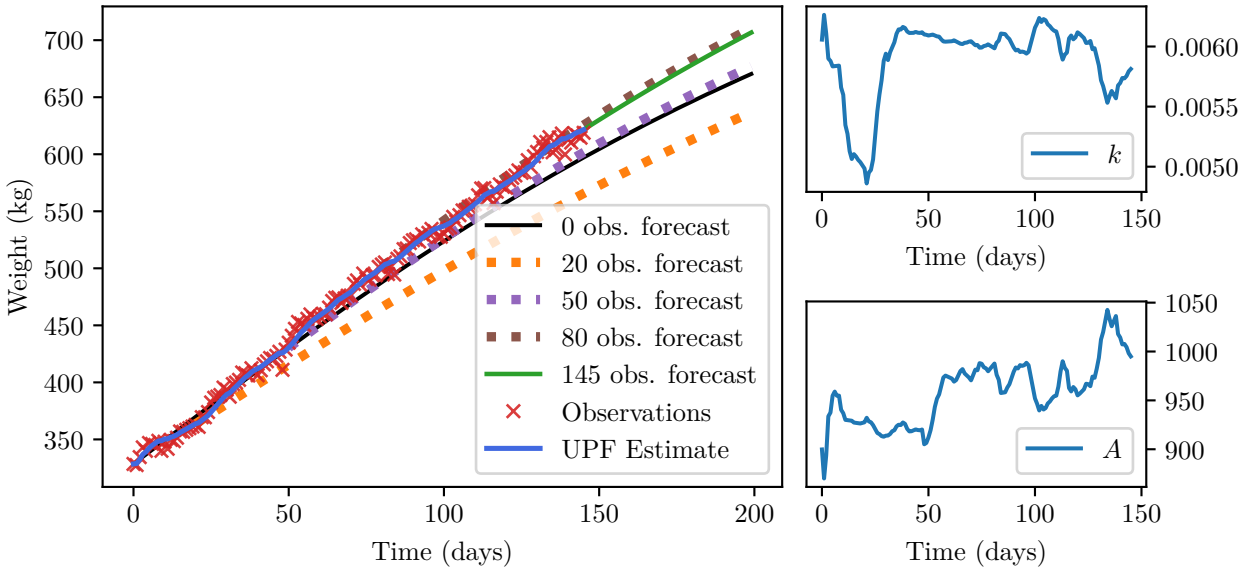


(a) Model forecast with PF estimates. (b) Parameter adjustment during DA.

Figure 4.3: Data assimilation of Richards' model using the PF

UPF

The estimates using the UPF for Richards' model, shown in Figure 4.4 are very similar to the UKF estimates, in Figure 4.2. Although the parameters go through abrupt changes, the estimates at time 80 closely match final observations.



(a) Model forecast with UPF estimates. (b) Parameter adjustment during DA.

Figure 4.4: Data assimilation of Richards' model using the UPF

4.3.2 Oltjen's Model

For Oltjen's model we only compare the results obtained by the UKF and PF, since the UPF could not be executed for Oltjen's model for most animals using the calibrated covariance matrices. This is caused because the UKF is numerically fragile, and some of the particles, when propagated through the UKF during the importance sampling stage, receive invalid (not a number, nan) or extreme values. Particles with invalid values, which cause errors in the model or the UKF, were filtered out during resampling. However, particles with extreme values, which cause the model to misbehave, are more difficult to identify, and could not be removed. While the covariance matrices could be changed in order to run the UPF, the initial conditions of the filter would not match those of the UKF and PF.

In figure 4.5, we see that both the UKF and PF have trouble converging the parameters. The PF shows an oscilatory behaviour on both parameters, and the MRSD of the parameter kD converges slowly, while the MRSD of DNA_{max} slowly increases, indicating poor convergence. The UKF also has oscilatory behaviour and slow convergence for the kD parameter, but appears to converge well for the DNA_{max}.

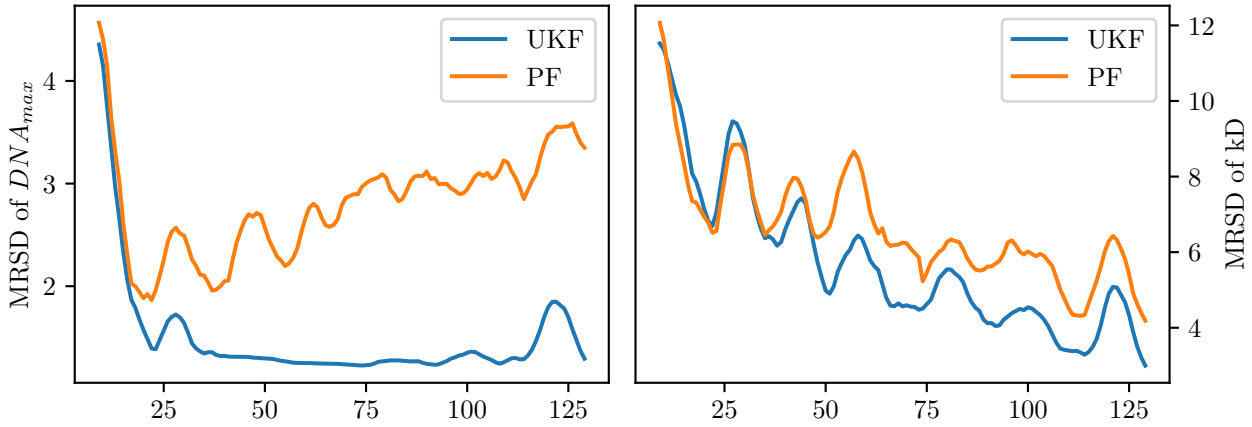
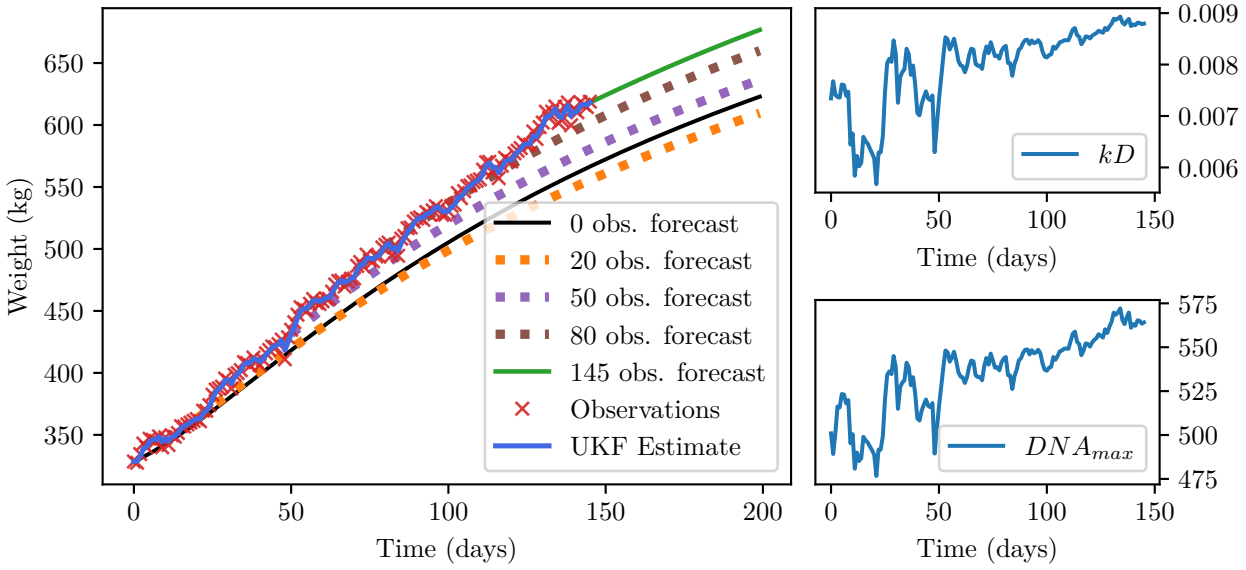


Figure 4.5: MRSD of Oltjen's model parameters with rolling window of 10 days

UKF

Figure 4.6 shows that the UKF estimates match the observations, and the parameters show some sign of converging, but the forecasts of the estimates have trouble matching future observations. We can see that this is caused by the fact that both parameters slowly increase as the DA progresses, but never converge.

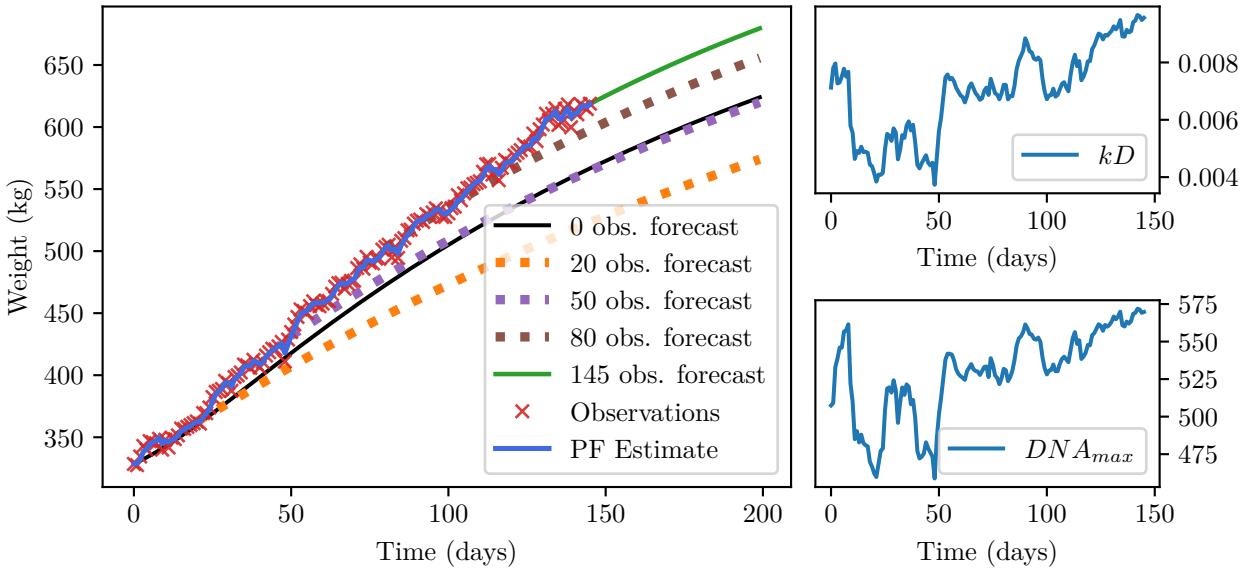


(a) Model forecast with UKF estimates. (b) Parameter adjustment during DA.

Figure 4.6: Data assimilation of Oltjen's model using the UKF

PF

Figure 4.7 shows that the PF estimates match observations, but the forecast of the estimates also have trouble adjusting to future observations. The parameter estimates made by the PF are similar to the estimates made by the UKF, in Figure 4.6, but present more difficulty to converge.



(a) Model forecast with PF estimates. (b) Parameter adjustment during DA.

Figure 4.7: Data assimilation of Oltjen's model using the PF

4.3.3 NASEM Model

Figure 4.8 shows the MRSD of adjusted parameters for NASEM's model, and we see that both parameters have almost identical adjustment by the three filters, and seem to converge as time progresses.

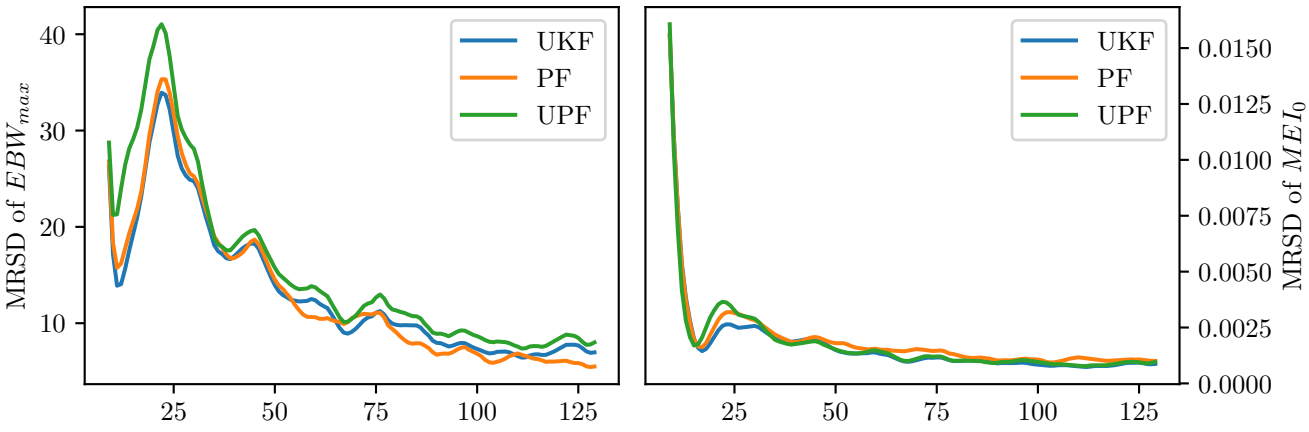
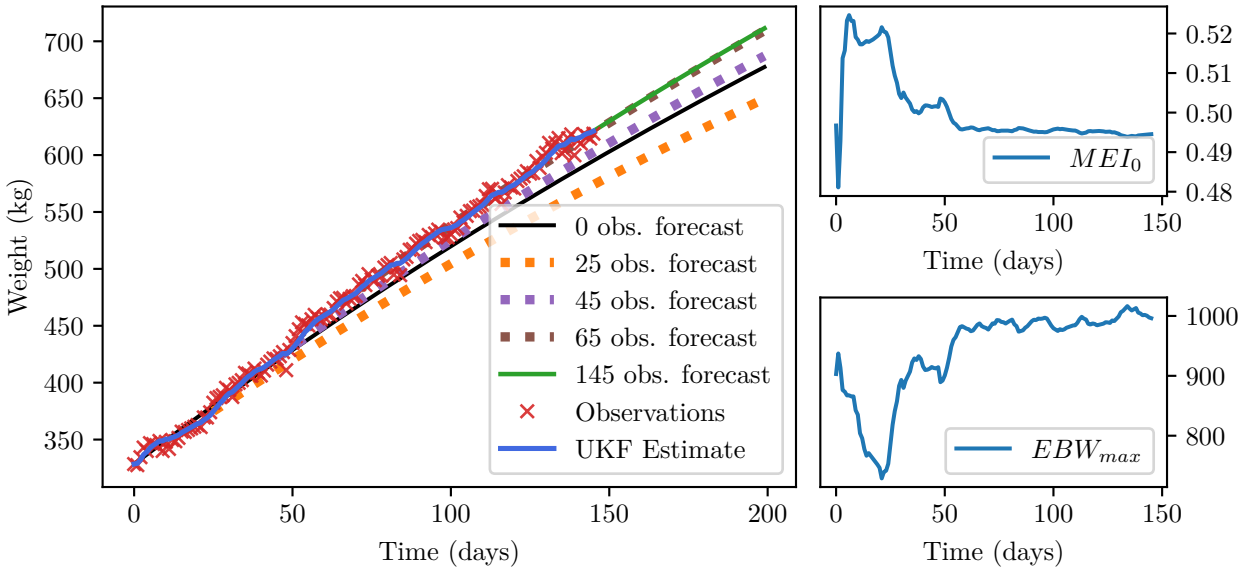


Figure 4.8: MRSD of NASEM's model parameters with rolling window of 10 days

UKF

Figure 4.9 shows that the UKF estimates follow the observations, and the forecasts of the estimates match future observations after time 80. The parameters also show reasonably stable convergence after about 60 observations.

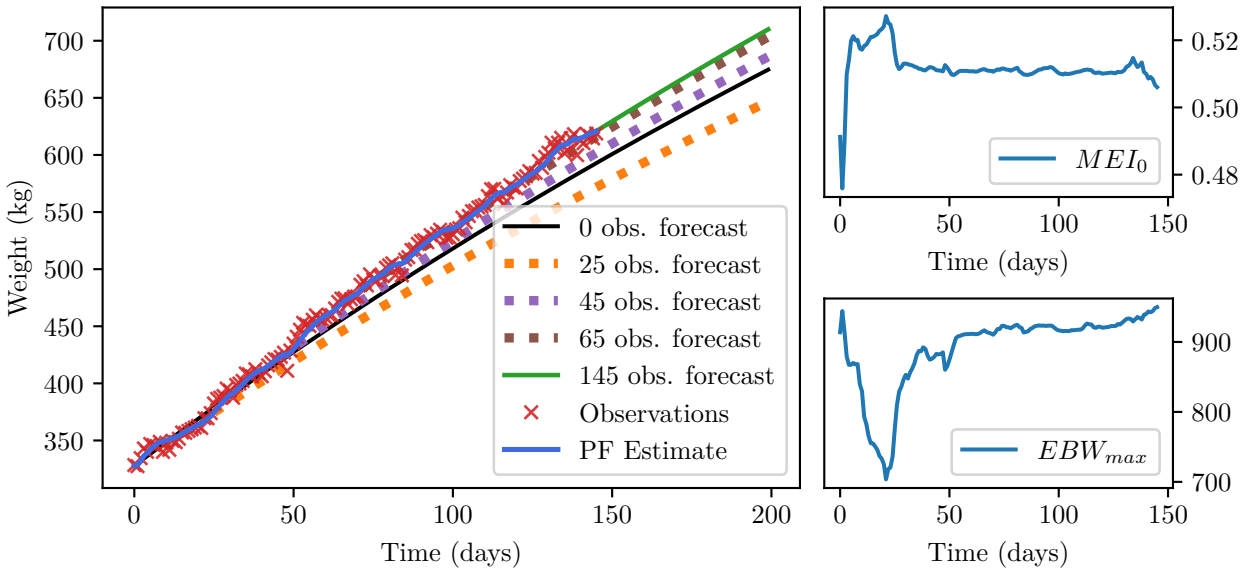


(a) Model forecast with UKF estimates. (b) Parameter adjustment during DA.

Figure 4.9: Data assimilation of NASEM's model using the UKF

PF

Figure 4.10 shows that the PF estimates follow the observations, and the forecasts of the estimates and parameter adjustment closely resemble those of the UKF, shown in Figure 4.9



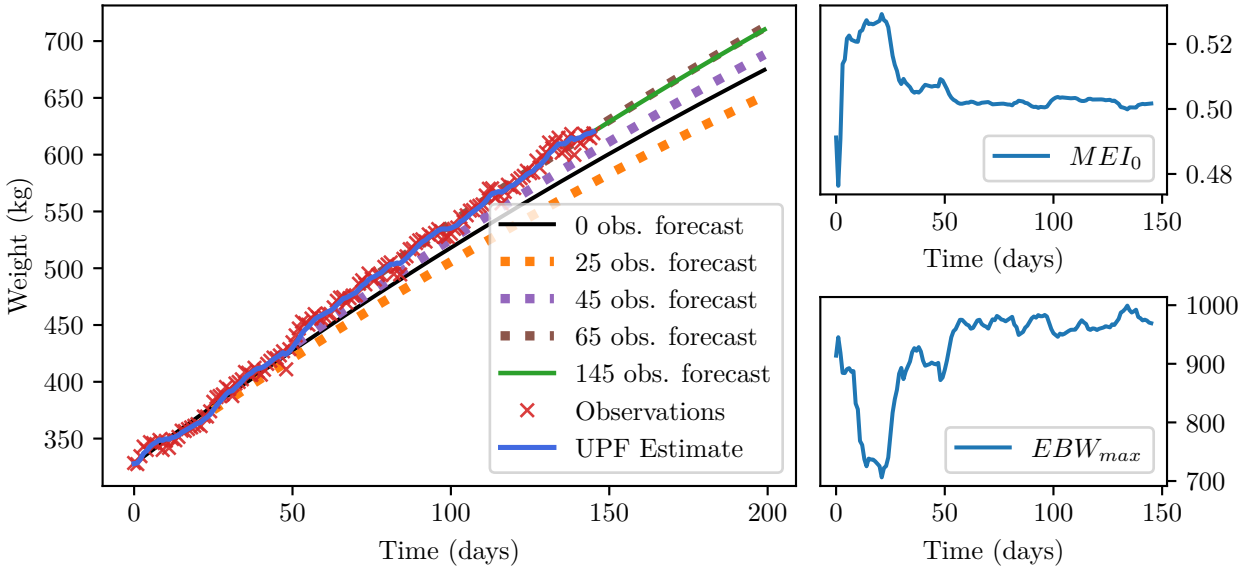
(a) Model forecast with PF estimates. (b) Parameter adjustment during DA.

Figure 4.10: Data assimilation of NASEM's model using the PF

UPF

Figure 4.11 shows that the PF estimates follow the observations, and the forecasts of the estimates and parameter adjustment closely resemble those of the UKF and

PF, shown in Figures 4.9 and 4.10.



(a) Model forecast with UPF estimates (b) Parameter adjustment during DA

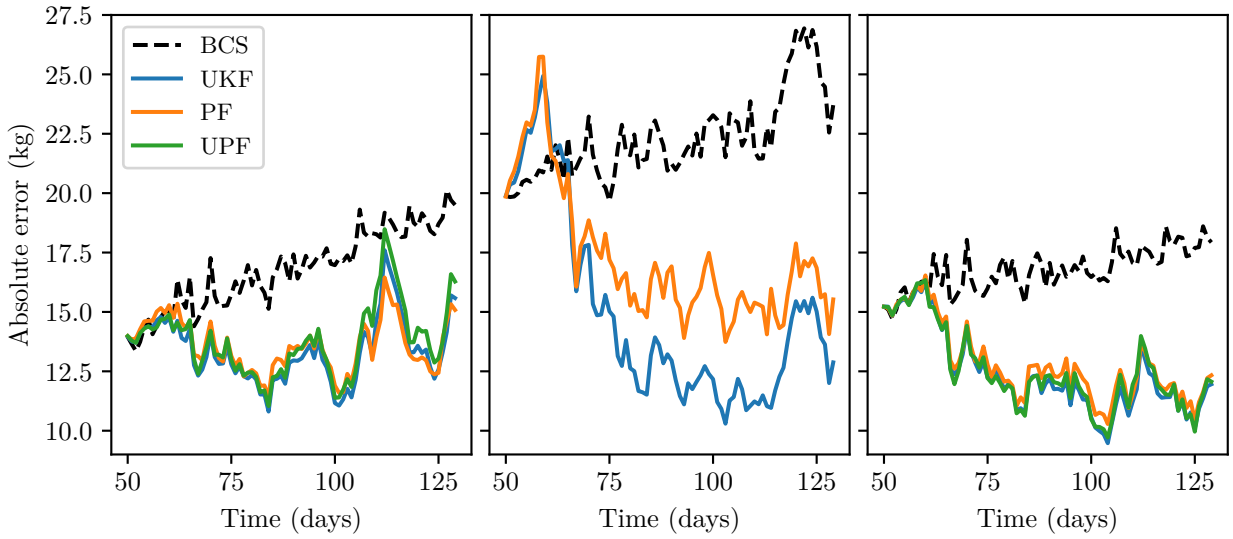
Figure 4.11: Data assimilation of NASEM's model using the UPF

4.3.4 Error Comparison

For each model, we represent the error by computing the NSAF-MAE, without taking the time average, and with 50 steps ahead, according to Eq. (4.8). As a Base Case Scenario (BCS), we compute the same metric for the forecasts of the models without performing DA, where each forecast is produced by reinitializing the state of the model for each observation. This way, we can compare the accuracy of forecasts produced by a model depending on whether it uses a filter or not, which filter is used and how many observations have been received by the filters.

$$error = \frac{1}{I}\sum_{i=0}^{I}|h(f^{50}(x_{t-n}^{(i)})) - z_t^{(i)}| \tag{4.8}$$

The figures in 4.12 show the evolution over time of the NSAF-MAE of each model for the BCS and for each filter. At a glance, we can see that all the filters have results with significantly lower NSAF-MAE, in comparison to the BCS, across all models, especially after around 20 days of measurements. We can also see that the three filters had similar performance for a same model, although the PF results in a slightly higher NSAF-MAE than the UKF when applied for Oltjen's model, in figure 4.12b.



(a) Richard's model (b) Oltjen's model (c) NASEM's model

Figure 4.12: NSAF-MAE of each model using different filters. All three graphics have the same scale. The NSAF-MAE of the UPF filter is not presented for Oltjen's model, since it did not perform properly.

Although we expect the forecast errors to get lower as the filters receive more observations, we can see that, around day 110, there is a rise in the NSAF-MAE for all three models, with a slightly more pronounced effect in the Richard's model, in figure 4.12a. This could indicate that some unmodeled conditions have changed in the cattle feedlot, such as feed quality, management or weather conditions, and the filters did not have enough time to adjust.

4.3.5 Overall Results

Table 4.1 lists the NSAF-MAE obtained for all models and filters. We can see that all filters performed better than the model alone (BCS).

	BCS	UKF	PF	UPF
Richards	16.79	13.38	13.59	13.76
Oltjen	22.29	14.93	17.25	-
NASEM	16.74	12.41	12.80	12.52

Table 4.1: NSAF-MAE of all models and filters

The best performing model in this study is NASEM's model. It produced the lowest NSAF-MAE, with matching results in all filters. With 30 observations, all filters reduced NASEM's BCS forecasting error by about 25%. With 60 observations, the filters reduce NASEM's BCS forecasting error by about 40%, and the parameters converge to relatively stable values.

It is important to note that, for NASEM's model, instead of directly adjusting the growth rate, we adjust the MEI_0 parameter, which in turn impacts in the diet of the animal. In Figure 4.8, we can see that the MEI_0 parameter converges much faster than the EBW_{max} parameter. Further study is required in order to determine if the choice of parameters is what led to better results in NASEM's model.

This study could not reproduce the results obtained by Van Der Merwe *et al.* [57], in which the UPF presents mean squared error (MSE) about 4 to 5 times lower than both UKF and PF. The difference in performance in comparison to the results in this study could be related to the characteristics particular to the studied system or model. The UPF was also the most difficult technique to perform, since it runs multiple instances of the UKF for different particles, making it vulnerable to numerical approximations and possible model instabilities.

The UKF presents the lowest NSAF-MAE for Richards' and Oltjen's models, while for NASEM's model, both UKF and UPF presented the lowest NSAF-MAE, with almost identical NSAF-MAE. The UKF also achieved the fastest execution time, averaging 26ms single-threaded wall-clock execution time per animal, while the PF averages 225ms and the UPF averages 800ms. Each filter has similar execution times for the three models. The execution can be further reduced by vectorizing or parallelizing the filters. Given these results, the UKF is overall the best performing algorithm in this study.

The best model-filter combination is NASEM's model with UKF, which has approximately the same error as the NASEM-UPF combination, but has a much lower computational cost, which enables scaling the technology for large animal productions or implementing it in low computational power field equipment.

Chapter 5

Conclusion

We conclude that it is possible to use DA techniques in order to reduce the error of animal growth model forecasts, by using the UKF, PF and UPF to adjust Richards', Oltjen's and NASEM models. The UKF and PF successfully improved forecasts of all three models, with at least a 10% reduction in model forecasting error after incorporating 20 observations, while the UPF only worked for Richard's and NASEM.

Overall, the best performing DA algorithm was the UKF one, since it has the lowest computational cost and its estimates presented model forecasting errors which were among the lowest across all studied models.

Considering the tested models, NASEM's was the one that achieved the lowest forecasting errors across the BCS and all tested filters. The NASEM model was also the only one whose parameters converged to stable values after incorporating about 60 measurements by using the UKF, PF or UPF.

The forecasts with the lowest errors were made by the NASEM model with the UKF and UPF filters, with almost identical results, and reduced the NSAF-MAE of the NASEM model by 25% after 30 observations, and 40% after 60 observations.

Furthermore, because the UKF can be executed in just under 4% of the execution time of the UPF, we conclude that the best performing model-filter combination is the NASEM model in conjunction with the UKF.

Thus, it was possible to achieve the objective the overall objective of this study, which was to determine whether it is possible to reduce the error of forecasts obtained by animal growth models through the use of data assimilation techniques in conjunction with animal weight collected through automated animal scales.

Further research can be carried out in order to improve the results obtained in this study, some of which are:

- Study other techniques that can be used to initialize the covariance matrices, such as Monte Carlo Markov Chain algorithms.

- Evaluate how the adjustment of other parameters of the animal growth models might impact the forecast errors.
- Evaluate the performance of other animal growth models in conjunction with DA algorithms
- Implement other DA algorithms, such as the Cubature Kalman Filter [2] and Moving Horizon Estimation [46]
- Evaluate the techniques listed in this study with other animal observation data, possibly including other variables, such as animal diet.

Bibliography

[1] Albertini, T., Biase, A., Barbosa, M, Cerqueira, A., Gonçalves, H., Barioni, L., Caixeta-Filho, J. V., Péra, T., Dias, C., Medeiros, S., *et al.* "627 BeefTrader: Optimal economical endpoint decision support system for feedlots and meat packers". In: *Journal of Animal Science* 95.suppl_4 (2017), pp. 307–307.

[2] Arasaratnam, I. and Haykin, S. "Cubature kalman filters". In: *IEEE Transactions on Automatic Control* 54.6 (2009), pp. 1254–1269. ISSN: 00189286. DOI: 10.1109/TAC.2009.2019800.

[3] Barioni, L., Dake, C., and Parker, W. "Optimizing rotational grazing in sheep management systems". In: *Environment International* 25.6 (1999). International Congress on Modelling and Simulation, pp. 819–825. ISSN: 0160-4120. DOI: https://doi.org/10.1016/S0160-4120(99)00051-3. URL: https://www.sciencedirect.com/science/article/pii/S0160412099000513.

[4] Barioni, L. G. "Modelagem dinamica e otimizacao metaheuristica para apoio a tomada de decisoes na recria e engorda de bovinos de corte." In: *Embrapa Cerrados-Teses/dissertações (ALICE)* (2002). DOI: https://doi.org/10.11606/T.11.2019.tde-20191220-124512.

[5] Barioni, L. G., Tonato, F., and Albertini, T. Z. "Orçamentação forrageira: revisitando os conceitos e atualizando as ferramentas". In: *Simpósio sobre o Manejo de Pastagem* 26.2011 (2011), pp. 71–96.

[6] Barioni, L. G. *Tabelas para estimativa de ingestão de matéria seca de bovinos de corte em crescimento em pastejo*. Tech. rep. 2007. URL: https://ainfo.cnptia.embrapa.br/digital/bitstream/CPAC-2010/28925/1/comtec-142.pdf.

[7] Behrendt, K., Cacho, O., Scott, J. M., and Jones, R. "Using seasonal stochastic dynamic programming to identify optimal management decisions that achieve maximum economic sustainable yields from grasslands under climate risk". In: *Agricultural Systems* 145 (2016), pp. 13–23. ISSN: 0308521X. DOI: 10.1016/j.agsy.2016.03.001. URL: http://dx.doi.org/10.1016/j.agsy.2016.03.001.

[8] Bennett, J. R., Maxwell, S. L., Martin, A. E., Chadès, I., Fahrig, L., and Gilbert, B. "When to monitor and when to act: Value of information theory for multiple management units and limited budgets". In: *Journal of Applied Ecology* 55.5 (2018), pp. 2102–2113. DOI: https://doi.org/10.1111/1365-2664.13132. eprint: https://besjournals.onlinelibrary.wiley.com/doi/pdf/10.1111/1365-2664.13132. URL: https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/1365-2664.13132.

[9] Boligon, A. and Albuquerque, L. "Genetic parameters and relationships of heifer pregnancy and age at first calving with weight gain, yearling and mature weight in Nelore cattle". In: *Livestock Science* 141.1 (2011), pp. 12–16. ISSN: 1871-1413. DOI: https://doi.org/10.1016/j.livsci.2011.04.009. URL: https://www.sciencedirect.com/science/article/pii/S1871141311001673.

[10] Brown, J., Fitzhugh Jr, H., and Cartwright, T. "A comparison of nonlinear models for describing weight-age relationships in cattle". In: *Journal of Animal Science* 42.4 (1976), pp. 810–818.

[11] Buendia, E, Tanabe, K, Kranjc, A, Baasansuren, J, Fukuda, M, Ngarize, S, Osako, A, Pyrozhenko, Y, Shermanau, P, and Federici, S. "refinement to the 2006 IPCC guidelines for national greenhouse gas inventories". In: *IPCC: Geneva, Switzerland* (2019).

[12] Byrski, A., Oplatková, Z., Carvalho, M., and Kisiel-Dorohinicki, M. "Advances in Intelligent Modelling and Simulation". In: *Simulation Tools and Applications (: Springer)* (2012).

[13] Chizzotti, M. L., Machado, F. S., Valente, E. E., Pereira, L. G., Campos, M. M., Tomich, T. R., Coelho, S. G., and Ribas, M. N. "Technical note: Validation of a system for monitoring individual feeding behavior and individual feed intake in dairy cattle". In: *Journal of Dairy Science* 98.5 (2015), pp. 3438–3442. ISSN: 15253198. DOI: 10.3168/jds.2014-8925.

[14] Crisan, D. and Doucet, A. "Convergence of sequential Monte Carlo methods". In: *Signal Processing Group, Department of Engineering, University of Cambridge, Technical Report CUEDIF-INFENGrrR38* 1 (2000).

[15] Dickinson, R. A., Morton, J. M., Beggs, D. S., Anderson, G. A., Pyman, M. F., Mansell, P. D., and Blackwood, C. B. "An automated walk-over weighing system as a tool for measuring liveweight change in lactating dairy cows". In: *Journal of Dairy Science* 96.7 (2013), pp. 4477–4486. ISSN: 00220302. DOI: 10.3168/jds.2012-6522. URL: http://dx.doi.org/10.3168/jds.2012-6522.

[16] Dowson, O., Philpott, A., Mason, A., and Downward, A. "A multi-stage stochastic optimization model of a pastoral dairy farm". In: *European Journal of Operational Research* 274.3 (2019), pp. 1077–1089. ISSN: 0377-2217. DOI: https://doi.org/10.1016/j.ejor.2018.10.033. URL: https://www.sciencedirect.com/science/article/pii/S0377221718308865.

[17] Eggleston, S., Buendia, L., Miwa, K., Ngara, T., Tanabe, K., *et al. IPCC guidelines for national greenhouse gas inventories*. 2006. URL: https://www.osti.gov/etdeweb/biblio/20880391.

[18] Fang, H., Tian, N., Wang, Y., Zhou, M., and Haile, M. A. "Nonlinear Bayesian estimation: From Kalman filtering to a broader horizon". In: *IEEE/CAA Journal of Automatica Sinica* 5.2 (2018), pp. 401–417. ISSN: 23299274. DOI: 10.1109/JAS.2017.7510808. arXiv: 1712.01406.

[19] Fitzhugh, H. "Analysis of growth curves and strategies for altering their shape". In: *Journal of animal Science* 42.4 (1976), pp. 1036–1051.

[20] Garcia, F., Sainz, R., Agabriel, J., Barioni, L., and Oltjen, J. "Comparative analysis of two dynamic mechanistic models of beef cattle growth". In: *Animal Feed Science and Technology* 143.1 (2008). Mathematical Models that Predict the Effects of Feed Characteristics on Animal Performance, pp. 220–241. ISSN: 0377-8401. DOI: https://doi.org/10.1016/j.anifeedsci.2007.05.012. URL: https://www.sciencedirect.com/science/article/pii/S0377840107001708.

[21] Ghil, M. and Malanotte-Rizzoli, P. "Data assimilation in meteorology and oceanography". In: *Advances in geophysics*. Vol. 33. Elsevier, 1991, pp. 141–266.

[22] Ghosh, S. and Bhattacharya, D. "Optimization in microbial pest control: An integrated approach". In: *Applied Mathematical Modelling* 34.5 (2010), pp. 1382–1395. ISSN: 0307-904X. DOI: https://doi.org/10.1016/j.apm.2009.08.026. URL: https://www.sciencedirect.com/science/article/pii/S0307904X09002649.

[23] Gray, D., Parker, W., Kemp, E., Kemp, P., Brookes, I., Horne, D., Kenyon, P., Matthew, C., Reid, S. J., and Valentine, I. "Feed planning - alternative approaches used by farmers". In: *Proceedings of the New Zealand Grassland Association* (2003), pp. 211–217. ISSN: 0369-3902. DOI: 10.33584/jnzg.2003.65.2495. URL: https://www.researchgate.net/publication/339618064.

[24] Grewal, M. S. and Andrews, A. P. "Applications of Kalman Filtering in Aerospace 1960 to the Present". In: *IEEE Control Systems* 30.3 (2010), pp. 69–78. ISSN: 1066033X. DOI: 10.1109/MCS.2010.936465.

[25] Gustavo Barioni Adauto Luiz Mancini, L. "Desenvolvimento de simuladores na agropecuária". In: *Tecnologias da informação e comunicação e suas relações com a agricultura*. 2014. URL: http://www.alice.cnptia.embrapa.br/alice/handle/doc/1010756.

[26] Holter, J., West, J., and McGilliard, M. "Predicting Ad Libitum Dry Matter Intake and Yield of Holstein Cows1". In: *Journal of Dairy Science* 80.9 (1997), pp. 2188–2199. ISSN: 0022-0302. DOI: https://doi.org/10.3168/jds.S0022-0302(97)76167-8. URL: https://www.sciencedirect.com/science/article/pii/S0022030297761678.

[27] Julier, S. J. and Uhlmann, J. K. *A general method for approximating nonlinear transformations of probability distributions*. Tech. rep. 1996, pp. 1–27. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.46.6718.

[28] Julier, S. J. and Uhlmann, J. K. "New extension of the Kalman filter to nonlinear systems". In: *Signal Processing, Sensor Fusion, and Target Recognition VI* 3068 (1997), p. 182. ISSN: 0277786X. DOI: 10.1117/12.280797.

[29] Julier, S. J. and Uhlmann, J. K. "Unscented filtering and nonlinear estimation". In: *Proceedings of the IEEE* 92.3 (2004), pp. 401–422. ISSN: 00189219. DOI: 10.1109/JPROC.2003.823141.

[30] Kalman, R. E. "A New Approach to Linear Filtering and Prediction Problems". In: *Transactions of the ASME–Journal of Basic Engineering* 82.Series D (1960), pp. 35–45.

[31] Kalnay, E. *Atmospheric modeling, data assimilation and predictability*. Cambridge university press, 2003.

[32] Kitagawa, G. "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models". In: *Journal of computational and graphical statistics* 5.1 (1996), pp. 1–25.

[33] Labbe, R. *Kalman and Bayesian Filters in Python*. https://github.com/rlabbe/Kalman-and-Bayesian-Filters-in-Python. 2020.

[34] Lahoz, W., Khattatov, B., and Menard, R. *Assimilation, Data*. 2010. ISBN: 9783540747024.

[35] Luersen, M. A. and Le Riche, R. "Globalized Nelder–Mead method for engineering optimization". In: *Computers & structures* 82.23-26 (2004), pp. 2251–2260.

[36] Luo, Y., Ogle, K., Tucker, C., Fei, S., Gao, C., LaDeau, S., Clark, J. S., and Schimel, D. S. "Ecological forecasting and data assimilation in a data-rich era". In: *Ecological Applications* 21.5 (2011), pp. 1429–1442.

[37] Marques, J., de O. Silva, R., Barioni, L., Hall, J., Tedeschi, L., and Moran, D. "An improved algorithm for solving profit-maximizing cattle diet problems". In: *Animal* 14 (2020), s257–s266. ISSN: 1751-7311. DOI: https://doi.org/10.1017/S1751731120001433. URL: https://www.sciencedirect.com/science/article/pii/S1751731120001433.

[38] Meier, E., Prestwidge, D., Liedloff, A., Verrall, S., Traill, S., and Stower, M. "Crop Livestock Enterprise Model (CLEM) – a tool to support decision-making at the whole-farm scale". In: (2019).

[39] Miranda, L. J. V. "PySwarms, a research-toolkit for Particle Swarm Optimization in Python". In: *Journal of Open Source Software* 3 (21 2018). DOI: 10.21105/joss.00433. URL: https://doi.org/10.21105/joss.00433.

[40] NASEM. *Nutrient requirements of beef cattle*. 2016.

[41] Oltjen, J. W., Bywater, A. C., Baldwin, R. L., and Garrett, W. N. "Development of a Dynamic Model of Beef Cattle Growth and Composition". In: *Journal of Animal Science* 62.1 (1986), pp. 86–97. ISSN: 0021-8812. DOI: 10.2527/jas1986.62186x.

[42] Oltjen, J. W. and Owens, F. N. "Beef Cattle Feed Intake and Growth: Empirical Bayes Derivation of the Kalman Filter Applied to a Nonlinear Dynamic Model". In: *Journal of Animal Science* 65.5 (Nov. 1987), pp. 1362–1370. ISSN: 0021-8812. DOI: 10.2527/jas1987.6551362x. eprint: http://oup.prod.sis.lan/jas/article-pdf/65/5/1362/22991507/jan0650051362.pdf. URL: https://doi.org/10.2527/jas1987.6551362x.

[43] Oltjen, J., Bywater, A., Baldwin, R., and Garrett, W. "Development of a Dynamic Model of Beef Cattle Growth and Composition". In: *Journal of Animal Science* 62 (Jan. 1986). DOI: 10.2527/jas1986.62186x.

[44] Parker, W. J., Shadbolt, N. M., and Gray, D. I. *Strategic planning in grassland farming: Principles and applications*. Tech. rep. 1997, pp. 191–197.

[45] Pezzopane, J. R. M., Santos, P. M., Cruz, P. G. da, Bosi, C., and Sentelhas, P. C. "An integrated agrometeorological model to simulate Marandu palisade grass productivity". In: *Field crops research* 224 (2018), pp. 13–21.

[46] Rawlings, J. B. and Bakshi, B. R. "Particle filtering and moving horizon estimation". In: *Computers and Chemical Engineering* 30.10-12 (2006), pp. 1529–1541. ISSN: 00981354. DOI: 10.1016/j.compchemeng.2006.05.031.

[47] Sainz, R., Barioni, L., Paulino, P., Valadares Filho, S., and Oltjen, J. "Growth patterns of Nellore vs. British beef cattle breeds assessed using a dynamic, mechanistic model of cattle growth and composition". In: *Nutrient Digestion and Utilization in Farm Animals: Modelling Approaches*. 2006, pp. 160–170.

[48] Sarmento, J. L. R., Regazzi, A. J., Sousa, W. H. d., Torres, R. d. A., Breda, F. C., and Menezes, G. R. d. O. "Estudo da curva de crescimento de ovinos Santa Inês". In: *Revista Brasileira de Zootecnia* 35.2 (2006), pp. 435–442.

[49] Sermesant, M., Moireau, P., Camara, O., Sainte-Marie, J., Andriantsimiavona, R., Cimrman, R., Hill, D. L., Chapelle, D., and Razavi, R. "Cardiac function estimation from MRI using a heart model and data assimilation: advances and difficulties". In: *Medical Image Analysis* 10.4 (2006), pp. 642–656.

[50] Shi, Y. and Eberhart, R. "A modified particle swarm optimizer". In: *1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence (Cat. No. 98TH8360)*. IEEE. 1998, pp. 69–73.

[51] Sorge, U. S., Cherry, C., and Bender, J. B. "Perception of the importance of human-animal interactions on cattle flow and worker safety on minnesota dairy farms". In: *Journal of Dairy Science* 97.7 (2014), pp. 4632–4638. ISSN: 15253198. DOI: 10.3168/jds.2014-7971. URL: http://dx.doi.org/10.3168/jds.2014-7971.

[52] Tedeschi, L. A.-s. O., Fox, D. G., Sainz, R. D., Barioni, L. A.-s. G., Medeiros, S. A. R. d., and Boin, C. "Mathematical models in ruminant nutrition". en. In: *Scientia Agricola* 62 (Jan. 2005), pp. 76 –91. ISSN: 0103-9016. URL: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-90162005000100015&nrm=iso.

[53] Tedeschi, L. O., Boin, C., Nardon, R. F., and Leme, P. R. "Estudo da curva de crescimento de animais da raça Guzerá e seus cruzamentos alimentados a pasto, com e sem suplementação: 1. Análise e seleção das funções não-lineares". In: *Revista Brasileira de Zootecnia* 29.2 (2000), pp. 630–637.

[54] Tedeschi, L. O., Fox, D. G., and Guiroy, P. J. "A decision support system to improve individual cattle management. 1. A mechanistic, dynamic model for animal growth". In: *Agricultural Systems* 79.2 (2004), pp. 171–204. ISSN: 0308521X. DOI: 10.1016/S0308-521X(03)00070-2.

[55] TUTKUN, M. "Growth curve prediction of Holstein-Fresian bulls using different non-linear model functions". In: *Applied Ecology and Environmental Research* 17 (Jan. 2019), pp. 4409–4416. DOI: 10.15666/aeer/1702_44094416.

[56] Valadares Filho, S. C., Machado, P. A. S., Chizzotti, M. L., Amaral, H. F., Magalhães, K. A., Rocha Junior, V. R., and Capelle, E. R. *CQBAL 4.0. Tabelas Brasileiras de Composição de Alimentos para Ruminantes.* Tech. rep. 2018. URL: https://www.cqbal.com.br/#!/.

[57] Van Der Merwe, R., Doucet, A., De Freitas, N., and Wan, E. "The unscented particle filter". In: *Advances in Neural Information Processing Systems* (2001). ISSN: 10495258.

[58] Van Der Merwe, R., Doucet, A., De Freitas, N., and Wan, E. A. *The Unscented Particle Filter(Tech Report)*. Tech. rep. 9. 2000, pp. 1689–1699. arXiv: arXiv:1011.1669v3. URL: https://www.eecs.yorku.ca/course{_}archive/2010-11/W/4421/lectures/upf.pdf.

[59] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat,
bibinitperiodI., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

[60] Wan, E. A. and Van Der Merwe, R. "The unscented Kalman filter for nonlinear estimation". In: *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*. Ieee. 2000, pp. 153–158.

[61] Wan, E. A., Van Der Merwe, R., and Nelson, A. T. "Dual estimation and the unscented transformation". In: *Advances in Neural Information Processing Systems* (2000), pp. 667–672. ISSN: 10495258.

[62] Weiss, M, Troufleau, D, Baret, F, Chauki, H, Prévot, L, Olioso, A, Bruguier, N, and Brisson, N. "Coupling canopy functioning and radiative transfer models for remote sensing data assimilation". In: *Agricultural and Forest Meteorology* 108.2 (2001), pp. 113–128.

[63] Xue, M., Wang, D., Gao, J., Brewster, K., and Droegemeier, K. K. "The Advanced Regional Prediction System (ARPS), storm-scale numerical weather prediction and data assimilation." In: *Meteorology & Atmospheric Physics* 82 (2003).