

Article

# Remote Sensing Data Assimilation in Dynamic Crop Models Using Particle Swarm Optimization

Matthias P. Wagner<sup>1,\*</sup>, Thomas Slawig<sup>2</sup> , Alireza Taravat<sup>1</sup> and Natascha Oppelt<sup>1</sup> 

<sup>1</sup> Earth Observation and Modelling, Dept. of Geography, Kiel University, 24118 Kiel, Germany; art23130@gmail.com (A.T.); oppelt@geographie.uni-kiel.de (N.O.)

<sup>2</sup> Algorithmic Optimal Control—CO<sub>2</sub> Uptake of the Ocean, Dept. of Computer Science, Kiel University, 24118 Kiel, Germany; ts@informatik.uni-kiel.de

\* Correspondence: m.p.wagner@gmx.de; Tel.: +49-176-5788-6714

Received: 14 December 2019; Accepted: 7 February 2020; Published: 10 February 2020



**Abstract:** A growing world population, increasing prosperity in emerging countries, and shifts in energy and food demands necessitate a continuous increase in global agricultural production. Simultaneously, risks of extreme weather events and a slowing productivity growth in recent years has caused concerns about meeting the demands in the future. Crop monitoring and timely yield predictions are an important tool to mitigate risk and ensure food security. A common approach is to combine the temporal simulation of dynamic crop models with a geospatial component by assimilating remote sensing data. To ensure reliable assimilation, handling of uncertainties in both models and the assimilated input data is crucial. Here, we present a new approach for data assimilation using particle swarm optimization (PSO) in combination with statistical distance metrics that allow for flexible handling of model and input uncertainties. We explored the potential of the newly proposed method in a case study by assimilating canopy cover (CC) information, obtained from Sentinel-2 data, into the AquaCrop-OS model to improve winter wheat yield estimation on the pixel- and field-level and compared the performance with two other methods (simple updating and extended Kalman filter). Our results indicate that the performance of the new method is superior to simple updating and similar or better than the extended Kalman filter updating. Furthermore, it was particularly successful in reducing bias in yield estimation.

**Keywords:** particle swarm optimization (PSO); AquaCrop-OS; data assimilation; uncertainty quantification; crop yield estimation; model updating; canopy cover (CC)

## 1. Introduction

After decades of continuously rising yields, recent years have seen a slowing down in agricultural productivity growth in Europe. Furthermore, decreasing global production may be expected under certain climate scenarios [1,2]. Simultaneously, a growing world population, rising income per capita, and increasing demand for energy are expected to drive demand for agricultural products [3,4]. Combined with increasing risks of extreme weather events, these factors emphasize the need for timely and accurate crop production monitoring. A common approach is the use of dynamic biophysical crop models that simulate the soil–plant–atmosphere interface [5]. These models can simulate environmental interactions and field management, but have a limited capacity to represent geospatial information on larger scales.

To address this drawback, remote sensing imagery and crop models can be merged. Remote sensing can introduce high-resolution spatial information about plant development and health into the modeling process. The increasing availability of free satellite data helps to reduce costs, especially when replacing traditional field measurements or airborne campaigns. The abundance of data from the

Landsat archive and the Copernicus program by the European Space Agency (ESA) further fosters the integration of satellite data into crop models [6].

Following the early work by Delécolle et al., crop model data assimilation techniques may be categorized into three broad groups: forcing, re-calibration, and updating [7]. Forcing refers to the replacement of simulated values with measured data. This method is very efficient and easy to implement, but has several drawbacks. First, it requires measurements for each simulation step (e.g., daily observations), which are often unavailable or need to be interpolated. When integrating optical remote sensing data, in particular, frequent cloud cover can drastically reduce the number of available observations, even with shorter revisit times in constellations such as Sentinel-2. Second, forcing effectively breaks up the simulation loop because it replaces intermediate results with external inputs [8]. Third, it does not consider measurement uncertainties and therefore directly transfers errors to the model. Due to these drawbacks, a few recent studies have considered forcing.

A frequently applied technique is re-calibration, sometimes separated into re-initialization and re-parametrization. Here, the initial values and parameters of the crop model are iteratively changed by minimizing a cost function measuring the distance between the simulated state variables and observed ones [7,9]. Re-calibration therefore obtains a new set of parameters or initial values, thus allowing a simulation that resembles better observations. Although this method often improves model-based yield predictions, it has two flaws. First, re-calibration settings may be unrealistic or may represent an unreliable parameter setup [9]. Second, re-calibration can be computationally demanding because it requires multiple re-runs of the model, hampering larger scale applications.

Updating performs the assimilation during the simulation, only interfering when an observation is available. It therefore performs well even with few and infrequent observations and reduces processing time when compared to re-calibration. Furthermore, updating allows uncertainties in both the simulation and the data assimilated to be addressed [10]. However, it requires modifications in the model itself (i.e., the source code) and not all models allow such interference. The most commonly used updating techniques are the (extended) Kalman filter, particle filter, and the ensemble Kalman filter [11–14].

Following the definition by Kennedy and O'Hagan, model uncertainties may be classified into parameter, parametric, model inadequacy, residual variability, observation, and code uncertainties [15]. In the context of biophysical modeling, the most relevant sources of uncertainties are parameter uncertainty (errors related to suboptimal parameter settings), parametric or input uncertainty (errors in the input data driving the simulation, e.g., daily weather measurements), code uncertainty (approximations and inaccuracies in model implementation), and model inadequacy (e.g., model bias). Uncertainties related to implementations and inadequacies are usually addressed during model development and subsequent calibration and sensitivity studies [16–19]. Parameter and input uncertainties, however, are highly application- and context-dependent and need to be assessed individually.

Most updating approaches are robust and fast, but often lack a detailed representation of such uncertainties. The Kalman filter, for example, approximates uncertainties in the model and the measurement by a simple scalar (e.g., the standard deviation in repeated measurements) or a covariance matrix in the case that multiple variables are updated [20]. This approach does not allow for a detailed handling of different uncertainty sources. Techniques such as the above-mentioned ensemble Kalman or particle filter, may account for uncertainty in parameters and model states stochastically.

Both re-calibration and updating require the solution of an optimization problem, which is usually non-linear. For such kinds of problems, several numerical algorithms can be applied. In our updating technique, we employed particle swarm optimization (PSO) due to its reliable global optimization capacities and flexibility in inputs and objective functions (see Section 2.2.3). PSO has seen various applications in remote sensing, frequently in image segmentation and classification [21–23], but also in agricultural applications. Guo et al., for example, used the algorithm to couple the PROSAIL canopy reflectance model with the WheatGrow crop model based on vegetation indices [24]. Others have

used it in combination with multiple classifiers and algorithms for crop classification [25]. The most frequent application, however, is the (re-)calibration of crop models such as the WO<sup>r</sup>ld FO<sup>o</sup>d ST<sup>u</sup>dies model (WOFOST) [26], the Simple Algorithm for Yield Estimate (SAFY) [27], the Decision Support System for Agrotechnology Transfer (DSSAT) [28,29], or AquaCrop [30,31].

The main objective of this study was therefore to ensure increased flexibility of uncertainty handling. The new technique proposed allows the user to include different uncertainties in the process with minimal limitations on their type and definition. The technique should also be largely independent and self-calibrating to enable direct application with minimal prior adjustments, thus allowing fast assimilation of remote sensing observations.

Although many studies exist that combine remote sensing inputs with dynamic crop models, a direct comparison is difficult to draw. The diverse nature of approaches involving different sensors, input variables, crop types, crop models, calibration settings, application scales (field to national or even continental and global) and varying amounts of prior knowledge (e.g., detailed study plots with regular measurements), aggravate a direct comparison. To demonstrate the potential of the new method, we therefore decided to apply multiple updating schemes to the same datasets with the same model and calibration settings. We compared the results of the new approach to a simple updating scheme (replacing values in the model simulation directly) as well as an extended Kalman filter (EKF). As a case study, we assimilated Sentinel-2 canopy cover (CC) data into the AquaCrop-OS model v5.0a to improve the winter wheat yield estimation.

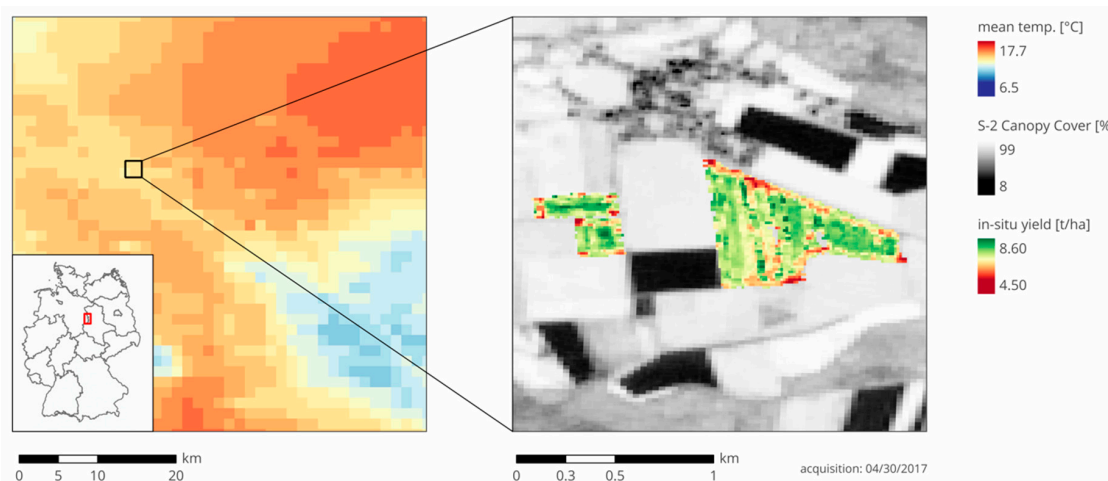
The rest of the paper consists of five parts. Section 2 describes the study area and data used and introduces the methodology. We provide some methodological background first, followed by a description of the updating technique. In Section 3, we describe results and discuss them in Section 4. Finally, Section 5 will give a short conclusion and outlook.

## 2. Materials and Methods

### 2.1. Datasets

#### 2.1.1. Study Area

Our study area is located in central Germany near the border of the states of Niedersachsen and Sachsen-Anhalt (see Figure 1). The climate is temperate/oceanic with warm summers and wet winters (Cfb in Koeppen-Geiger climate classification) [32]. The region is relatively warm and dry with an average temperature of 8.2 °C and an annual precipitation of 538 mm in the climate reference period 1960–1990 [33]. Our weather data of the years 2016 and 2017 revealed both years to be rather warm (9.8 and 10.8 °C), while precipitation was low in 2016 (436 mm) and high in 2017 (679 mm) compared to the long-term average. Soils in the region are typically stagnosols and brown earths originating from sandy and loamy glacial debris. Further, clayey soils from skeletal loam, sandy Loess over limestone, rendzinas, and some podzols also occur [34].



**Figure 1.** Map of study area and example of rasterized weather datasets in the form of mean air temperature (**left**). Example of yield data and canopy cover maps (**right**).

### 2.1.2. Weather Data

The German Weather Service (DWD) delivered daily weather data for the nearby weather station “Ummendorf” (11.18° E, 52.16° N) as well as  $1 \times 1 \text{ km}^2$  rasterized weather datasets for the whole of Germany (see Figure 1). Weather data include daily minimum and maximum temperatures, precipitation sums, and reference evapotranspiration based on the Penman–Monteith equation [35]. The raster datasets were used as input to the model, introducing a limited amount of spatial dynamics.

### 2.1.3. Canopy Cover Data

Our database consisted of atmospherically corrected Sentinel-2 Level-2A scenes between August 2015 and November 2017. We only considered scenes with generally low to moderate cloud cover (up to 50%). The dataset comprised of 116 scenes, covering the full winter wheat growing seasons for both harvest periods of 2016 and 2017. We used the biophysical processor implemented in the ESA Sentinel-2 Toolbox (S2TBX, version 6.0.4) to generate canopy cover (CC) maps from all scenes (see Figure 1). The processor employs artificial neural networks trained on a large dataset of radiative transfer simulations of canopy and leaf properties [36]. The documentation of the SNAP Biophysical Processor provides some theoretical performance indicators. The authors claim a low root mean square error (RMSE) of 0.04 for CC predictions on their validation dataset [36]. During pre-processing, we further performed a multi-threshold cloud and cloud shadow detection for each of our test fields to discard any potentially contaminated observations. The resulting number of observations ranged between three and 12 per growing season, depending on location.

### 2.1.4. Yield Data

Field data were obtained via GPS-based yield measurements on combine harvesters during harvest of 30 fields in 2016 and 32 fields in 2017. We removed outliers outside  $\pm 2.58$  standard deviations (99% threshold in a standard normal distribution), particularly false zero measurements that frequently occurred at the start and end of the harvest procedure. We then aggregated the remaining points to  $10 \times 10 \text{ m}^2$  yield maps matching Sentinel-2 observations (see Figure 1). The resulting mean yields of all fields were in good agreement with the reported department-level yield statistics [37–39]. The observed yield on the pixel-level ranged from 2.38 to 9.60 t/ha, and the mean field yields ranged from 3.90 to 7.63 t/ha. No information on measurement accuracy was provided.

For further analysis, we generated a pixel- and a field-level dataset. We split both randomly into 60% calibration (32 field observations, 23,375 pixel observations) and 40% validation (20 field observations, 15,584 pixel observations) data.

## 2.2. Methodological Background

### 2.2.1. AquaCrop-OS Description

AquaCrop is a dynamic crop model developed by the Food and Agriculture Organization of the United Nations (FAO). It simulates the yield response of herbaceous crops on a homogeneous field, considering water response and various stress effects [40–42]. Inputs for daily simulation are the maximum and minimum temperature data, precipitation sum, and potential evapotranspiration [43]. The simulation is considerably simplified compared to complex model suites such as the Decision Support System for Agrotechnology Transfer (DSSAT) [44,45], focusing on a global model applicability with a potentially limited range of available data.

The central part of the model is a crop productivity function that relates biomass accumulation to water productivity and evapotranspiration to obtain the total cumulative biomass:

$$B_T = K_{sb} \cdot WP^* \cdot \sum_{t=0}^T \left( \frac{Tr_t}{ET_{ot}} \right) \quad (1)$$

where  $B_T$  is the total accumulated biomass from  $t = 0$  days to  $t = T$ ;  $K_{sb}$  is an air temperature stress coefficient;  $WP^*$  is the water productivity normalized to annual mean  $CO_2$  concentration;  $Tr_t$  is daily crop transpiration; and  $ET_{ot}$  is daily potential evapotranspiration (both in mm).

AquaCrop represents the heat, drought, and cold stress effects via stress coefficients that can influence canopy development, stomatal conductance, canopy senescence, or harvest index development. The stress coefficients change with the level of stress following a convex to concave response curve [41,46]:

$$K_s = 1 - \frac{e^{S_{rel} f_{shape}} - 1}{e^{S_{rel}} - 1} \quad (2)$$

where  $K_s$  describes the stress response function;  $S_{rel}$  is the relative stress level ( $\leq 1$ ); and  $f_{shape}$  is a shape factor defining the curvature of the function.

The main state variable in the model is canopy cover (CC; sometimes referred to as Fraction of Vegetation Cover, FVC or FCOVER) that directly influences  $Tr_t$  in Equation (1) via a crop transpiration coefficient:

$$K_{C_{Tr}} = CC^* K_{C_{Tr,x}} \quad (3)$$

$$Tr = K_{sw} K_{C_{Tr}} ET_o \quad (4)$$

where  $CC^*$  is the current canopy cover (adjusted for micro-adjecive effects);  $K_{C_{Tr,x}}$  is the maximum crop transpiration coefficient for well-watered soil and a complete canopy;  $K_{sw}$  represents a soil water stress coefficient; and  $K_{C_{Tr}}$  is the current crop transpiration coefficient obtained. Therefore, CC is an important variable in biomass accumulation in Equation (1), and consequentially, yield as determined via a harvest index (i.e., percentage of biomass at crop maturity).

CC development over the growing season is determined mostly empirically. After crop emergence, CC first increases exponentially up to 50% of the maximum. A slowing growth follows until the maximum is reached. The value of CC stays constant until an exponential decay sets in at the beginning of senescence [46]. This process is summarized in the following equations:

$$\text{Canopy expansion : } CC = CC_0 e^{t CGC} \quad \text{for } CC \leq \frac{CC_x}{2} \quad (5)$$

$$CC = CC_x - 0.25 \frac{(CC_x)^2}{CC_0} e^{-t CGC} \quad \text{for } CC > \frac{CC_x}{2} \quad (6)$$

$$\text{Canopy senescence : } CC = CC_x \left[ 1 - 0.05 \left( e^{\frac{CDC}{CC_x}} - 1 \right) \right] \quad \text{for } CC \leq \frac{CC_x}{2} \quad (7)$$

where  $CC$  is the new canopy cover;  $CC_x$  is the maximum possible canopy cover;  $CC_o$  is the initial canopy cover at the start of growth; and  $CGC$  and  $CDC$  are canopy growth and decline coefficients, respectively. Dry yield is obtained by applying a harvest index (percentage) to the biomass value at maturity.

We used the open source version of the model called AquaCrop-OS, allowing us to make the necessary source code changes for the updating procedures described in Section 2.3 [47].

### 2.2.2. AquaCrop-OS Calibration

Due to the lack of information on winter wheat varieties on our test fields or regular in situ sampling, our prior knowledge for calibration was limited. We therefore relied on an empirical calibration of model parameters. This also made the simulation more general and independent of specific field conditions.

AquaCrop-OS offers a large number of crop parameters, separated into conservative ones that have previously been proven accurate in many different environments and user-specific ones [43]. The former were ignored for the most part in our calibration, except for the Canopy Growth and Decline Coefficients ( $CGC$  and  $CDC$ , see Table 1) due to their particular relevance in this context. We did not consider irrigation management because agriculture in our study area is exclusively rain-fed. Similarly, we assumed that field management follows a “best practice” due to high technological standards and a long tradition of industrialized agriculture in our study area.

**Table 1.** Calibration ranges and obtained calibrated crop parameters in AquaCrop-OS.

Parameter	Description	Calibration Range	Calibrated Value
CDC	Canopy Decline Coefficient	0.0015–0.0065	0.0038
CGC	Canopy Growth Coefficient	0.0025–0.0075	0.006
Emergence	Time from sowing to emergence in Growing Degree Days (GDD)	112–225	188
fshape_b	Shape factor of biomass productivity reduction due to insufficient GDDs	10.36–17.27	14.504
HIstart	Time from sowing to start of yield formation in GDDs	660–1.100	748
Kcb	Maximum crop coefficient at full canopy development	0.825–1.375	1.045
Maturity	Time from sowing to maturity in GDDs	1.800–3.000	2.160
Senescence	Time from sowing to senescence in GDDs	1.275–2.125	1.615

We performed a sensitivity analysis based on iterative changes to individual parameters and observed the influence on predicted yield. It revealed the parameters listed in Table 1 to be those most relevant for winter wheat yield prediction in our study area. We calibrated the model by altering parameters iteratively and minimizing the RMSE of yield. The calibration process improved the RMSE from 2.305 t/ha to 1.324 t/ha on field-level and from 2.264 t/ha to 1.521 t/ha on the pixel-level validation datasets. The optimal parameter settings for the pixel- and field-level were quite similar, so we decided to use the same calibration set on both scales. Table 1 provides a list of calibration settings.

### 2.2.3. Particle Swarm Optimization

Particle swarm optimization is a metaheuristic global optimization algorithm based on swarm intelligence principles of complex intelligent behavior emerging from primitive individual agents.

As such, it is part of the larger family of evolutionary computing [48,49]. Kennedy and Eberhart originally designed the algorithm following previous efforts by Reynolds in simulating realistic movements of bird flocks [50,51].

The particle swarm is a group (“swarm”) of candidate solutions (“particles”) moving in the multidimensional search space over time (i.e., iteration steps). Each particle is initiated as a random vector with a random initial velocity vector representing its movement in the search space. This velocity is updated at each iteration based on certain rules and the new particle fitness is obtained. In the original version, the process is only influenced by the best previous solution of the particle (previous best,  $p_{best}$ ) and the best solution obtained in its neighborhood (neighborhood best,  $n_{best}$ ) [48,49]. This neighborhood is described by the topology representing connections between the particles in the swarm. There are many different topologies used in the literature including local best, global best, and von Neumann topologies, but also dynamic topologies changing throughout the process based on time, Euclidian distance, and fitness–distance ratios, among others [52,53]. For a more detailed discussion, readers may refer to the paper by Poli et al. [54].

The following equations describe the central velocity and position update (all multiplications are element-wise):

$$\vec{v}_i(t) = \vec{v}_i(t-1) + \varphi_1 \epsilon_1 (\vec{p}_i - \vec{x}_i(t-1)) + \varphi_2 \epsilon_2 (\vec{p}_n - \vec{x}_i(t-1)) \quad (8)$$

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t) \quad (9)$$

where  $\vec{v}_i(t)$  is the new (updated) velocity vector of particle  $i$  at time step  $t$  and  $\vec{v}_i(t-1)$  is its previous velocity vector. The previous and new positions are given by  $\vec{x}_i(t-1)$  and  $\vec{x}_i(t)$ , respectively. The previous best solution is represented by  $\vec{p}_i$  and the neighborhood best solution by  $\vec{p}_n$ . The terms  $\varphi_1$  and  $\varphi_2$  refer to  $p_{best}$  and  $n_{best}$  coefficients, respectively, and  $\epsilon_1, \epsilon_2$  are random vectors [48]. One can interpret  $p_{best}$  and  $n_{best}$  coefficients as the tendency of particles to move independently or “toward the swarm”. The two elements are therefore closely related to exploration and exploitation.

Its metaheuristic approach distinguishes PSO from gradient-based optimization techniques. PSO does not use exact or approximated derivative information. It therefore does not need continuous or differentiable objective functions or any prior knowledge about the cost function [55]. This makes it very flexible in handling different types of inputs and even combinations of continuous and discrete functions. PSO is also considered to be reliable in finding global optima, even in highly heterogeneous, complex solution spaces as simulated by test functions like the Ackley or Hölder table functions [48,56]. Moreover, PSO scales very well with high-dimensional inputs as the number of function evaluations is determined by the swarm size, not the number of input variables.

However, PSO is not a deterministic algorithm, but includes stochastic elements. The process is therefore not entirely predictable, even identical starting conditions may lead to different iteration steps and even to different solutions due to the random component of the process [54]. As a result, it is up to the user to determine application-specific parameters (such as swarm size, coefficients, topology, etc.) that ensure a reliable and fast convergence. Furthermore, unlike gradient descent-related algorithms that reach a local minimum under certain assumptions, the convergence in PSO methods is only valid in a stochastic setting.

To ensure fast and reliable optimization results, we compared a number of different PSO variants and settings. This included different swarm sizes, inertia weights, cognitive and social coefficients, static and dynamic topologies (local best, global best, dynamic nearest neighbor, dynamic fitness–distance ratio, among others) as well as different distributions for random vector sampling (uniform, normal, Lévy).

We observed that Clerc’s constriction coefficients [57] were superior to inertia weights or velocity bounds alone. We therefore used  $\varphi_1 = \varphi_2 = 2.05$ , and the constriction coefficient  $\chi$  calculated according to [57]. We further observed that, although not necessarily required when using inertia weights or constriction coefficients, limiting  $v_{max}$  to the dynamic range of the input was advantageous in some

cases, as suggested in [58]. The von Neumann and dynamic nearest neighbor topologies showed quite similar performances with the former being chosen due to higher computational efficiency. Different random distributions showed no significant impact in this context. The swarm size was set to 20, a common value in the literature. Larger numbers of particles were unable to improve convergence significantly, but logically increased the number of function evaluations, slowing down the process. Table 2 presents the fastest and most reliable setup we obtained. In our applications, this implementation usually converged very quickly to the optimum within 10–15 iterations.

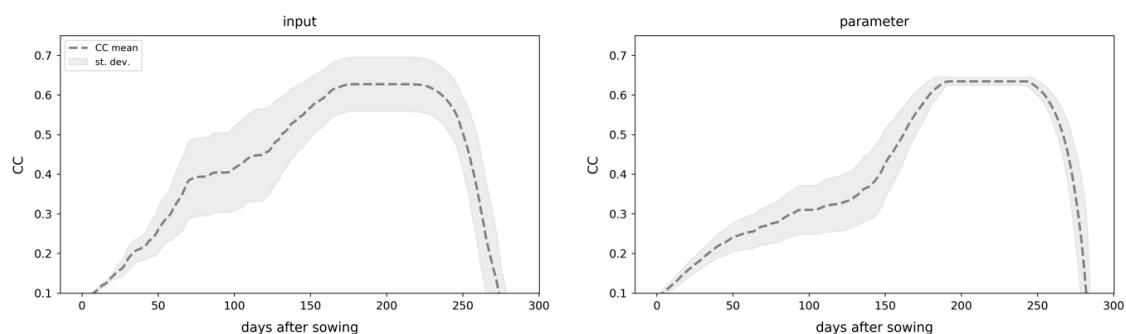
**Table 2.** Settings used for particle swarm optimization algorithm.

Parameter	Value
swarm size ( $n$ )	20
pbest coefficient ( $\varphi_1$ )	2.05
nbest coefficient ( $\varphi_2$ )	2.05
maximum velocity ( $v_{max}$ )	dynamic range
constriction coefficient ( $\chi$ )	$\sim 0.72984$
topology	von Neumann (4 neighbors)
random distribution	uniform

#### 2.2.4. Uncertainty Quantification

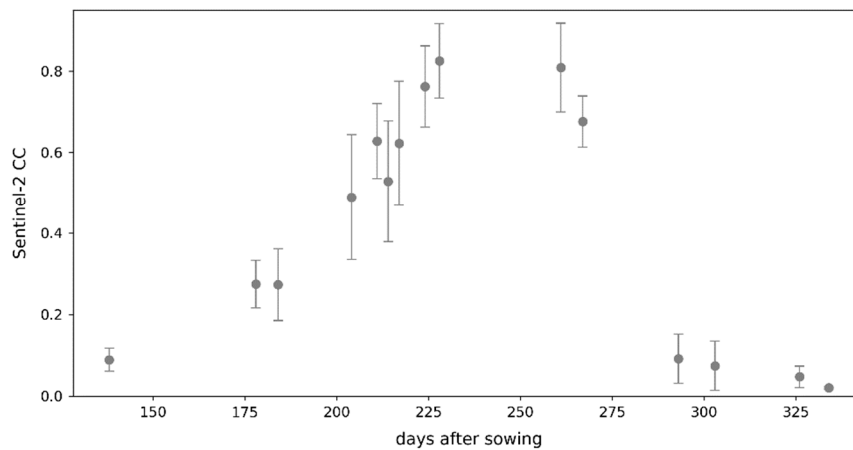
We considered multiple sources of uncertainties, both in the model and the remote sensing data. These need to be quantified before they can be included in the updating procedure. We are unable to account for potential weather measurement errors or instrument-related issues. Still, we are able to quantify the reaction of AquaCrop-OS canopy cover simulations to perturbations in weather inputs and parameter settings.

We achieved this via Monte Carlo simulations. First, we estimated input-related uncertainty by randomly perturbing a 10-year mean weather time series with Gaussian random noise. The magnitude of the noise was determined by the daily variance observed in the same 10-year period. We obtained 10,000 CC time series from AquaCrop-OS simulations on these randomized weather datasets. Second, we assessed parameter uncertainties accordingly by randomly sampling parameter settings from a normal distribution around the calibrated values in Table 1 with a standard deviation of 1/10 of the calibration range. This ensured a sufficient variation within a realistic range around the calibrated settings. We performed 10,000 Monte Carlo simulations randomizing all parameters listed in Table 1. Both model-related uncertainties are illustrated in Figure 2. Third, we estimated uncertainties in the remote sensing data. Here, the procedures on field- and pixel-level are different. On the field level, we used a set of all CC pixel values in a given field at the observation date; on the pixel-level, we used only values in the  $3 \times 3$  pixel neighborhood (see Figure 3).



**Figure 2.** Visualization of observed uncertainties throughout the growing season. Uncertainty from Monte Carlo simulations weather inputs (left) and perturbing parameters (right).





**Figure 3.** Example of uncertainties in Sentinel-2 CC values in a field. Dots refer to the mean of all CC observations (i.e., pixels) in the field, error bars indicate standard deviations.

Using these datasets, we created probability density functions (PDF) representing the probability of all possible CC values (between 0 and 1) of each uncertainty source. We employed kernel density estimation with a symmetric Gaussian kernel. Tests showed that a narrow bandwidth of 0.02 yielded the best results. Finally, we represented the uncertainty in the current simulation by a Gaussian distribution around the currently simulated CC value using a bandwidth of 0.2.

### 2.3. Updating Methodology

#### 2.3.1. Simple Updating

The simple updating scheme we employed replaced simulated CC values directly with remote sensing observations without any additional processing and with no consideration of uncertainties. As a result, the simulated CC values remained unconsidered and errors in the remote sensing data were directly transferred to the model.

#### 2.3.2. Extended Kalman Filter Updating

Since its first description in [11], the Kalman filter has become one of the most common data assimilation techniques [20]. It iteratively updates an estimated value by incorporating information from incoming measured values, taking into account the uncertainty associated with both the measurement and the estimated value. The Kalman filter assumes a linear model. Its extension to non-linear models is the EKF. Here, a linearization of the original non-linear model function is used to update the uncertainty (i.e., the covariance matrix) of the estimate of the model state [14,59].

In our case, we have a non-linear model, but we assimilated the scalar state variable CC directly. Thus, no additional observation is present. Both facts simplify the EKF procedure and make the updating computationally very efficient. Assuming we have the estimates of the state variable  $x_k$  and its uncertainty  $P_k$  at time step instant  $t_k$ . We now obtain a new observation value  $y_{k+1}$  at the next time instant  $t_{k+1}$ . Then, the EKF performs a predictor step for the model state

$$\hat{x}_{k+1} = f(x_k) \quad (10)$$

using the original non-linear model. Additionally, the uncertainty is predicted as:

$$\hat{P}_{k+1} = F_k P_k F_k \quad (11)$$

Here, we assume that the model has no error and uses an approximation  $F_k \approx f'(x_k)$  for the derivative of the model function. In our case, this derivative is also a scalar. Now, the Kalman gain is computed as:

$$G_{k+1} = \frac{\hat{P}_{k+1}}{\hat{P}_{k+1} + R_{k+1}} \quad (12)$$

where  $R_{k+1}$  is the uncertainty in the measurement  $y_{k+1}$ . Now, the correction step computes new estimates of the state and its uncertainty as:

$$x_{k+1} = \hat{x}_{k+1} + G_{k+1}(y_{k+1} - \hat{x}_{k+1}) \quad (13)$$

$$P_{k+1} = (1 - G_{k+1}) \hat{P}_{k+1} \quad (14)$$

We computed the derivative approximation needed in Equation (11) by a finite difference formula:

$$F_k = \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} = \frac{\hat{x}_{k+1} - \hat{x}_k}{x_k - x_{k-1}} \quad (15)$$

This approximation uses only already computed quantities. In the first assimilation step ( $k = 0$ ), a modification is needed to replace the value  $x_{k-1}$  and  $f(x_{k-1})$ .

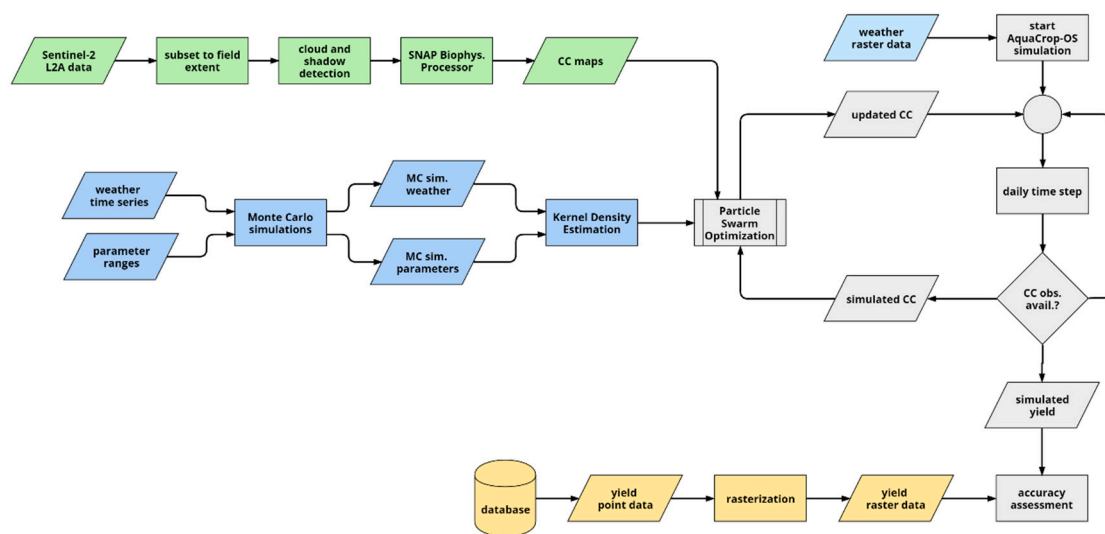
As mentioned previously, after accounting for clouds and cloud shadows, the remaining observations were not too frequent. In case of frequent observations, the updated uncertainty can be propagated continuously throughout the whole EKF process. In our case, however, we often encountered large time gaps in between observations. This implies that the assimilation cannot take place in every time step of the model. Thus, the function  $f$  in Equation (10) represents not just one model step, but rather summarizes a concatenation of model steps between subsequent time instants  $t_k$  and  $t_{k+1}$  where measurements are available. As a consequence, the derivative approximation in Equation (15) is an average of the derivative of the model in the interval  $[t_k, t_{k+1}]$ . As indicated in Section 2.2.1, AquaCrop-OS varies significantly in its simulation procedures depending on growth stages and environmental influences. Thus, this kind of averaging of the derivative seems to be reasonable. As noted above, the derivative for the first assimilation step has to be approximated in a slightly different way. Here, we used a state at a time instant in the interval  $[t = 0, t_1]$  instead of  $x_{k-1}$ .

The uncertainty in the state is initially assumed to be 0.2. The uncertainty in the measured values was estimated as the standard deviation of all CC values at the observed location and time (i.e., all pixels of a field on the field-level and pixels in the  $3 \times 3$  neighborhood on the pixel-level).

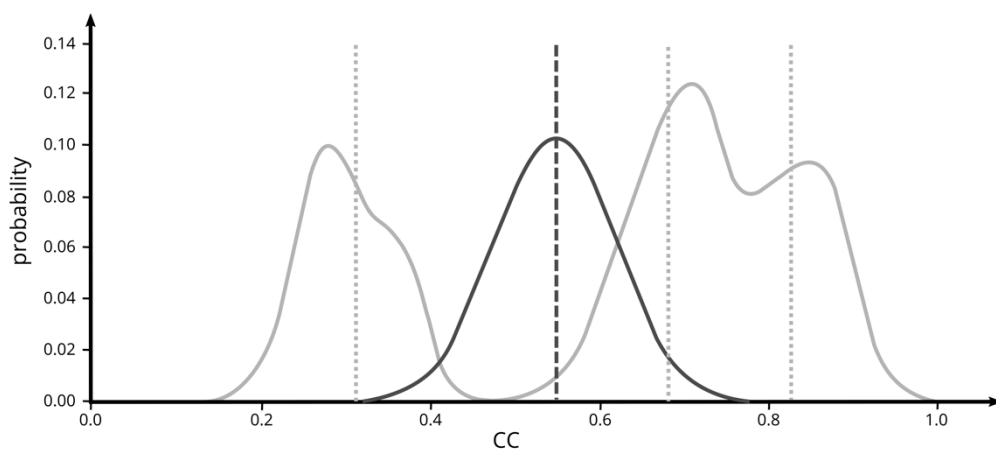
### 2.3.3. New Updating Scheme

Figure 4 demonstrates the main processing steps for our new method. Preparation of CC data (green), weather data (blue), and yield maps (yellow) were discussed in Section 2.1 and uncertainty quantification (dark blue) was covered in Section 2.2.4. In this section and the next, we will explain the details of the actual updating process and accuracy assessment (grey).

The fundamental idea behind our approach is to balance all uncertainties (relating to the model and the CC observations) to obtain the updated value. To do so, we represented all uncertainties as PDFs (see Section 2.2.4). We then obtained a hypothetical optimal Gaussian distribution, as described by a mean  $\mu$  and a standard deviation  $\sigma$ , that balances all uncertainties in terms of statistical distance (see Figure 5). In other words, we assumed that the mean of a distribution minimizing statistical distances to all PDFs will provide us with a better estimate given the available information. We employed PSO to search for the mean and standard deviation of this optimal Gaussian distribution.



**Figure 4.** Flowchart of pre-processing steps and the new updating algorithm. Colors refer to different aspects: Sentinel-2 CC data (green), weather input data (blue), yield data preparation (yellow), uncertainty quantification (dark blue), and the actual simulation and updating process (grey).

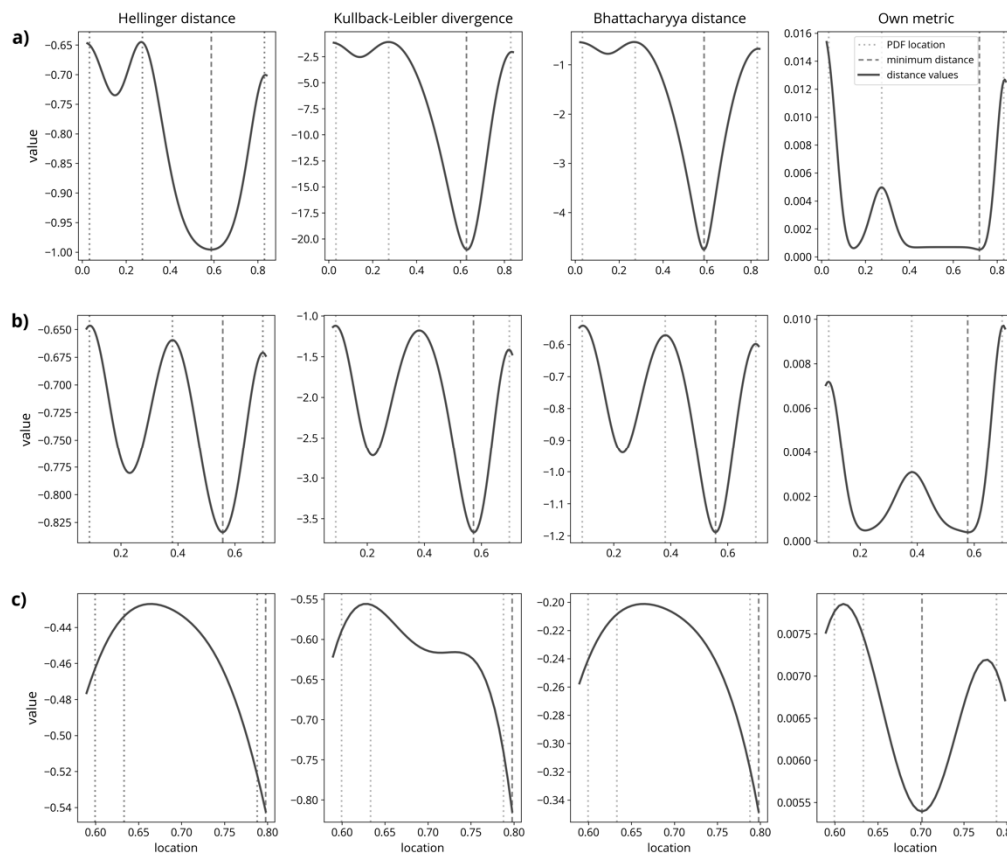


**Figure 5.** Idealized representation of optimal Gaussian distribution (solid dark grey line) and summed uncertainty probability density functions (solid light grey lines). Maximum likelihood estimators (MLE) of the uncertainty probability density functions indicated by vertical dotted lines, updated CC value (i.e., location of optimal Gaussian distribution) indicated by vertical dashed line. This example represents a common case in our tests with model-related uncertainties and remote sensing observations indicating different value ranges for optimal CC.

The central element of this technique is the representation of distance or similarity between probability distributions or their respective PDFs. There are a number of statistical distance and divergence metrics proposed in the literature. Some of the most common ones are the Hellinger distance, the Kullback–Leibler divergence, and Bhattacharyya distance [60–62].

When comparing calculations measuring the statistical distance of the optimal Gaussian distribution to a set of uncertainty PDFs, the different metrics behaved similarly. Figure 6 demonstrates this with example cases where three PDFs at different locations and with different standard deviations are considered. The values shown for the different distance metrics were obtained using a brute force algorithm moving an optimal distribution of standard deviation 0.05 from 0.01 to 0.99 through the search space. These demonstrative cases are, however, drastically simplified as they assume all PDFs to be perfectly symmetric Gaussian distributions and keep the standard deviation of the optimal

distribution fixed. Additionally, this example case only assumes three PDFs while the situation logically becomes much more heterogeneous when more are considered.



**Figure 6.** Three simplified example cases (a–c) of Hellinger distance, Kullback–Leibler divergence, Bhattacharyya distance, and the distance metric of Equation (16) used in this study (scales on the first three inverted). Solid black lines represent the distance of an optimal distribution moving through the search space according to three uncertainty PDFs (locations indicated by dotted grey vertical lines). Vertical dashed lines indicate the minimum value the optimizer would obtain.

Figure 6a,b show that in cases where the PDFs are relatively far apart, the three distance metrics behave similarly. Although magnitudes may differ significantly, the general shapes (number and location of local optima) are quite similar. If we consider the case where the three PDFs are located closely to one another, however, problems arise. Here, as illustrated in Figure 6c, all three distances failed to establish a clear minimum within the search range and instead produced single peaks or plateaus. This produced a situation with two potential minimum solutions at the extremes. In searching for the minimum value, the algorithm would run off to either side. In this special case of only Gaussian normal distributions, this is equivalent to choosing either the upper or the lower bound as the updated value. This may be mitigated, as attempted in a previous implementation [63], by using the mean squared distance of the Maximum Likelihood Estimator (MLE) of the optimal distribution to the MLEs of all uncertainty PDFs. This approach, however, has two major drawbacks. First, using the MLE as an indicator for the position of a distribution is only representative if it is a unimodal, approximately normal distribution. Second, it necessitates an additional processing step to determine the MLEs. Although obtaining MLEs is a straightforward optimization problem (maximizing the sum

of probabilities in all PDFs), it adds to the processing time and introduces a potential error source. Due to this drawback, we decided to employ a different metric described as:

$$D(f_{opt}(\mu, \sigma), f_{sum}) = \sum_{i=1}^N \left( q_i \cdot \frac{(p_i - \mu)}{\sigma} \right)^2 \quad (16)$$

where  $f_{opt}(\mu, \sigma)$  and  $f_{sum}$  are the optimal Gaussian distribution defined by  $\mu$  and  $\sigma$  and the sum of all uncertainty PDFs, respectively. Furthermore,  $q_i$  is the probability value of the summed uncertainty PDFs. Although this definition violates at least two important criteria of a metric as it is neither symmetric nor is it limited to the range (0, 1), we may still refer to it as such for simplicity.

Fundamentally, this metric weights the probabilities of the uncertainty PDFs based on their distance to the optimal distribution measured in standard deviations (z-scores). As shown in Figure 6, using this metric, we can ensure a clear minimum within the search space, even in the case of PDFs located very closely to one another.

Additionally, we addressed the possibility that not all PDFs may have the same relevance for the update. We therefore introduced a weighting, following the assumption that PDFs that are more similar to the currently simulated CC value of the model contain less new information than those that were more dissimilar. To represent this, we employed the Hellinger distance to introduce a weighting of the individual PDFs. We obtained weights by calculating the Hellinger distance for each uncertainty PDF to a narrow Gaussian distribution around the currently simulated CC value, following the equation:

$$H(P, Q) = \frac{1}{\sqrt{2}} \sqrt{\sum_{i=1}^N (\sqrt{p_i} - \sqrt{q_i})^2} \quad (17)$$

where  $P$  and  $Q$  are two distributions with  $p_i$  and  $q_i$  describing the probabilities of the two distributions at point  $i$ . The Hellinger distance ranges between 0 (identical) to 1 (no overlap). We used the resulting distance values to establish normalized exponential weights:

$$w_i = \frac{\exp(\alpha H(f_i, f_{sim}))}{\sum_{j=0}^n \exp(\alpha H(f_j, f_{sim}))} \quad (18)$$

where  $f_i$  is the PDF representing the respective uncertainty and  $f_{sim}$  is the distribution around the simulated CC value. The denominator represents the sum of all Hellinger distances, ensuring summation to unity. This further allows us to introduce  $\alpha$ , a simple multiplicative factor that determines the magnitude of weights with higher values of  $\alpha$  leading to more emphasis on dissimilar PDFs (i.e., those with larger Hellinger distance). Combining Equation (16) with the weights determined in Equation (18) results in an optimization problem of the form:

$$\min \sum_{i=1}^N (D(f_{opt}(\mu, \sigma), f_{sum}) \cdot w_i) \quad (19)$$

This objective function, however, may lead to the process of finding an optimum with a very large standard deviation  $\sigma$ . Although such an extremely flat distribution would indeed minimize statistical distances, it is not a useful solution for our approach. If the distribution is essentially flat, the mean value may be positioned anywhere in the CC range without having any significant effect on statistical distance. In other words, a flat distribution would allow for any CC value to be an optimal solution. To avoid this, we penalized  $\sigma$  of the optimal Gaussian distribution. This leads to our final optimization problem:

$$\min \sum_{i=1}^N (D(f_{opt}(\mu, \sigma), f_{sum}) \cdot w_i) + \sigma \quad (20)$$

We then searched for optimal values of  $\mu$  and  $\sigma$  to minimize Equation (20). A drawback in terms of computation, however, is that values around the optimum tend to be generally very small and the minimum is not as distinct in cases where PDFs are far apart, like in Figure 6a,b. This puts a larger emphasis on the proper settings and reliability of the optimizer used. The theoretically unbounded, but in our observations generally small value range of this metric, however, facilitated the introduction of constraints compared to, for example, the Kullback–Leibler divergence with observed values between 0.02 and over 30.

#### 2.3.4. Performance Analysis

The comparison of updating schemes involved three test situations: field-level, pixel-level, and pixel-to-field aggregated level estimations where we simulated yield on a pixel basis and compared the mean of these individual estimates to the observed mean yield of the field. We performed this analysis on the pixel-level validation dataset and therefore did not use all pixels. Still, the large size of 40% of all pixels ensured a representative sample of pixels for all fields. As stated earlier, we did not have in situ measurements of canopy cover or regular samples of biomass in the field. It was therefore not our goal to obtain realistic CC time series or closely recreate biomass development. Instead, our comparison relies primarily on the capacity for improving yield predictions.

We considered two versions of our method: one with a fixed (user-defined) value for the weighting factor  $\alpha$  and an adaptive one letting PSO automatically determine  $\alpha$  in the process. Previous tests on the calibration data showed quite high values of  $\alpha$  around 5–10 were advantageous for field-level simulations, while on the pixel-level, values around 1–2 were preferable. In our comparison, we compared these settings with the results obtained by the adaptive weighting within the continuous range 1–10.

We also evaluated the capability of our method to incorporate different uncertainties by adding uncertainty sources one at a time and observing the effect on performance. First, we introduced remote sensing data into the update, then parameter-related and, finally, weather-related uncertainty.

We used three metrics to analyze the results. The main metric for overall performance in the results was the RMSE.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (21)$$

where  $y_i$  describes the reference yield value and  $\hat{y}_i$  is the predicted value. To determine bias in the results, we used the mean percentage error (MPE).

$$MPE = 100 \cdot \frac{1}{N} \sum_{i=1}^N \frac{\hat{y}_i - y_i}{y_i} \quad (22)$$

Furthermore, we calculated the  $R^2$  scores.

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2} \quad (23)$$

where  $\bar{y}$  is the mean of the reference data. To incorporate the inherent uncertainty that is present in any sort of yield measurement, we also employed a percentage match metric (i.e., counting the predicted values that fall within a certain range) around the respective reference value (in this case  $\pm 20\%$ ).

$$p_{match} = 100 \cdot \frac{1}{N} \sum_{i=1}^N \begin{cases} |y_i - \hat{y}_i| \leq 0.2 \cdot y_i & 1 \\ else & 0 \end{cases} \quad (24)$$

### 3. Results

In this section, we describe the performance of yield predictions of AquaCrop-OS on the field-level, pixel-level, and pixel-to-field aggregated level. The very low values of  $R^2$  we observed throughout all experiments did not benefit the interpretation and were therefore omitted from descriptions. We analyze and address this topic specifically in Section 3.4 and the discussion (Section 4).

#### 3.1. Field-Level Yield Estimation

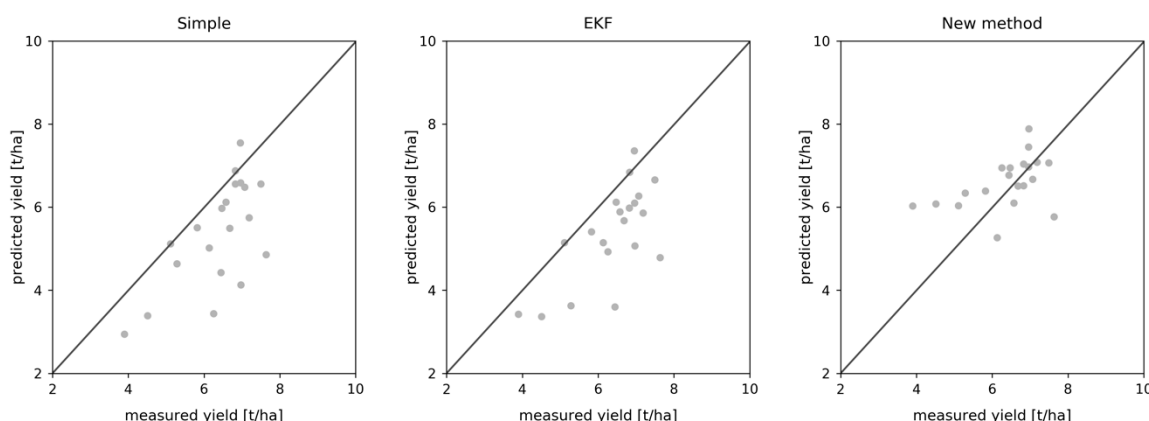
The results in Table 3 show that without assimilation, AquaCrop-OS produced rather poor results with a RMSE of 1.32 t/ha and a quite significant bias, expressed in an MPE of 15.2%, which indicates a tendency of the model to overestimate yield. The simple updating scheme had no effect in terms of accuracy, but inverted the bias to an underestimation. The EKF performed better with a significant reduction of RMSE to 1.20 t/ha and a slightly smaller bias.

**Table 3.** Accuracies of the AquaCrop-OS field-level yield estimation on the validation dataset using different updating schemes. Uncertainties refer to the PDFs considered. RS: remote sensing; pars: parameter-related; weather: weather-related.

Updating	$\alpha$	Uncertainties	RMSE [t/ha]	MPE [%]	$R^2$	Pmatch [%]
none	-	-	1.32	15.2	0.01	70.0
simple	-	-	1.37	-15.6	0.09	70.0
EKF	-	-	1.20	-14.8	0.35	70.0
new method	5	RS	1.14	-10.5	0.08	85.0
new method	5	RS, pars	1.09	4.3	0.04	75.0
new method	5	RS, pars, weather	1.04	5.7	0.06	75.0
new method	10	RS	1.23	-7.5	0.13	80.0
new method	10	RS, pars	1.11	4.2	0.10	85.0
new method	10	RS, pars, weather	1.05	4.0	0.11	80.0
new method	auto	RS	1.25	-8.4	0.12	75.0
new method	auto	RS, pars	1.09	4.5	0.15	75.0
new method	auto	RS, pars, weather	0.90	5.9	0.21	80.0

Our new PSO method performed similar to or better than the EKF with the two  $\alpha$  values of 5 and 10 when using all three uncertainties. Using only remote sensing inputs led to small changes in RMSE, but inverted the bias from positive to negative in all setups, similar to what we observed in the simple and EKF updates. Adding parameter-related uncertainties led to an improvement in both RMSE and MPE, while the subsequent addition of weather-related input affected results only slightly, and sometimes slightly deteriorated MPE. The adaptive version performed comparably and even outperformed the non-adaptive ones occasionally. Overall, when using all three uncertainties, model results improved by up to 0.42 t/ha in the RMSE, 11.2% in MPE, and 15% in terms of pmatch.

All versions of our approach were particularly successful in reducing bias. This is also apparent in the scatter plot in Figure 7. It also indicates a tendency of the new method to reduce the range of predictions by avoiding low results < 5 t/ha.



**Figure 7.** Scatter plots of measured vs. predicted yield on the field-level for (left to right) simple update, EKF update, and the new method (adaptive; all three uncertainties).

### 3.2. Pixel-Level Yield Estimation

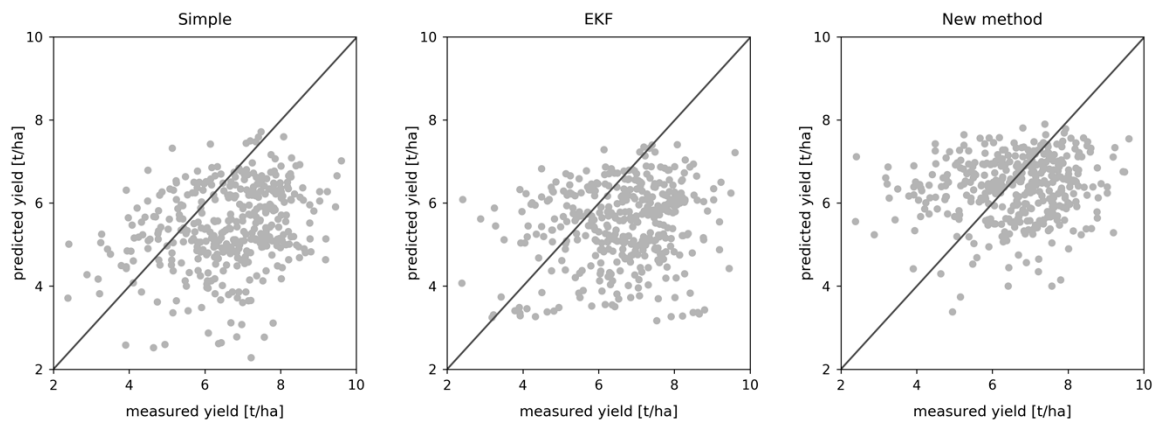
Table 4 shows that on the pixel-level, the model with no updating again produced a high RMSE and showed a bias of 12.7%. Both simple updating and the EKF inverted the bias to an underestimation of  $-14.9\%$  and  $-13.3\%$ , respectively. Interestingly, both techniques increased the inaccuracies.

**Table 4.** Accuracies of AquaCrop-OS pixel-level yield estimation on the validation dataset using different updating schemes. Abbreviations as in Table 3.

Updating	$\alpha$	Uncertainties	RMSE [t/ha]	MPE [%]	R <sup>2</sup>	Pmatch [%]
none	-	-	1.52	12.7	0.08	65.6
simple	-	-	1.88	-14.9	0.07	43.8
EKF	-	-	1.79	-13.3	0.08	45.6
new method	1	RS	1.80	-12.9	0.06	43.6
new method	1	RS, pars	1.43	1.8	0.07	64.4
new method	1	RS, pars, weather	1.47	2.5	0.07	64.9
new method	2	RS	1.72	-12.7	0.05	49.0
new method	2	RS, pars	1.47	0.2	0.06	61.3
new method	2	RS, pars, weather	1.50	3.2	0.07	64.9
new method	auto	RS	1.82	-12.8	0.05	46.2
new method	auto	RS, pars	1.52	-2.9	0.07	59.7
new method	auto	RS, pars, weather	1.48	1.1	0.09	62.1

The new method performed better when using remote sensing and parameter-related uncertainties, while adding weather-related uncertainty did not improve the results consistently. Still, even the best results only reduced the RMSE by about 0.09 t/ha. Despite that, it again managed to reduce the bias significantly. The adaptive version seemed to incorporate the different uncertainties more consistently, as indicated by decreasing RMSE and MPE with each added uncertainty. The scatter plot in Figure 8 supports these findings. As previously mentioned, the new method reduced the bias through slightly higher predictions with a slightly smaller range.





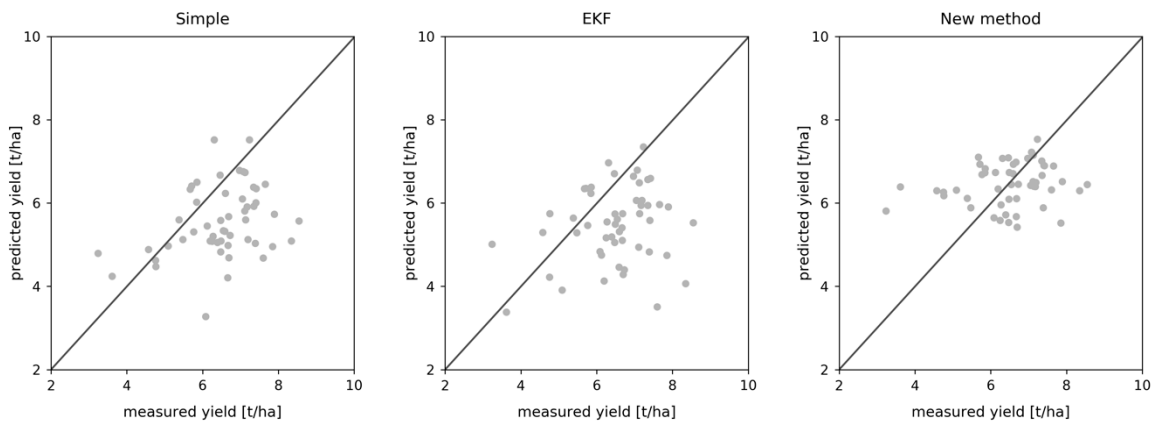
**Figure 8.** Scatter plots of measured vs. predicted yield on the pixel-level for (left to right) simple update, EKF update, and the new method (adaptive; all three uncertainties). To improve visual interpretation, we displayed only a subset of 400 data points.

### 3.3. Pixel-To-Field Aggregated Yield Estimation

In general, the aggregated results (Table 5) are better than those on the pixel-level, indicating a benefit from aggregation. When compared to field-level results, however, differences become apparent. Without an update, the model produced better results on the aggregated than on the field-level, while the simple update showed no significant difference. The EKF, however, performed worse on aggregated scales than on field-wise runs, indicated by a higher RMSE. The new method, in comparison, provides similar or better results on the aggregated compared to the field-level and bias tended to be a bit smaller. Results for the different uncertainty setups generally behaved in accordance to the observations in previous sections. Compared to the model without updating, we observed only small improvements for all updating schemes. Again, the scatter plot in Figure 9 shows a smaller range, but also a reduction in bias in the predictions using the new method compared to simple updating and EKF.

**Table 5.** Accuracies of AquaCrop-OS yield estimation on the validation dataset aggregating pixel-level simulations to the field-level using different updating schemes. Abbreviations as in Table 3.

Updating	$\alpha$	Uncertainties	RMSE [t/ha]	MPE [%]	R <sup>2</sup>	Pmatch [%]
none	-	-	1.12	11.6	0.09	72.2
simple	-	-	1.36	-12.5	0.11	69.4
EKF	-	-	1.33	-13.3	0.09	61.1
new method	1	RS	1.27	-10.9	0.10	72.2
new method	1	RS, pars	0.92	1.6	0.11	86.1
new method	1	RS, pars, weather	0.96	3.2	0.11	80.6
new method	2	RS	1.26	-10.5	0.10	72.2
new method	2	RS, pars	0.95	0.6	0.05	83.3
new method	2	RS, pars, weather	0.97	1.3	0.07	80.6
new method	auto	RS	1.28	-10.4	0.08	69.4
new method	auto	RS, pars	0.96	-2.4	0.09	86.1
new method	auto	RS, pars, weather	0.92	1.5	0.07	86.1

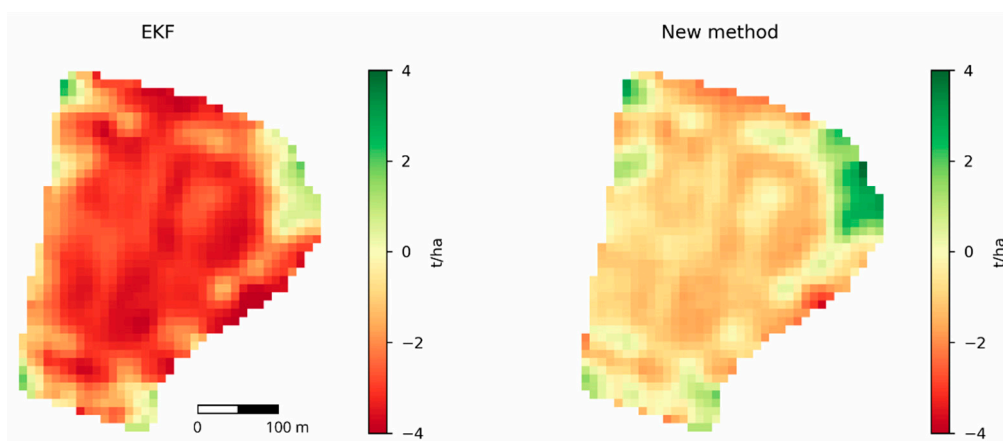


**Figure 9.** Scatter plots of measured vs. predicted yield on the pixel-to-field aggregated level for (left to right) simple update, EKF update, and the new method (adaptive; all three uncertainties).

3.4.  $R^2$  Performance

As mentioned previously, we generally observed low  $R^2$  values in all yield estimations. This is particularly surprising, since even good RMSE and MPE values were associated with low  $R^2$  values. The scatter plots in previous sections also indicate low correlations between the predicted and measured yield values.

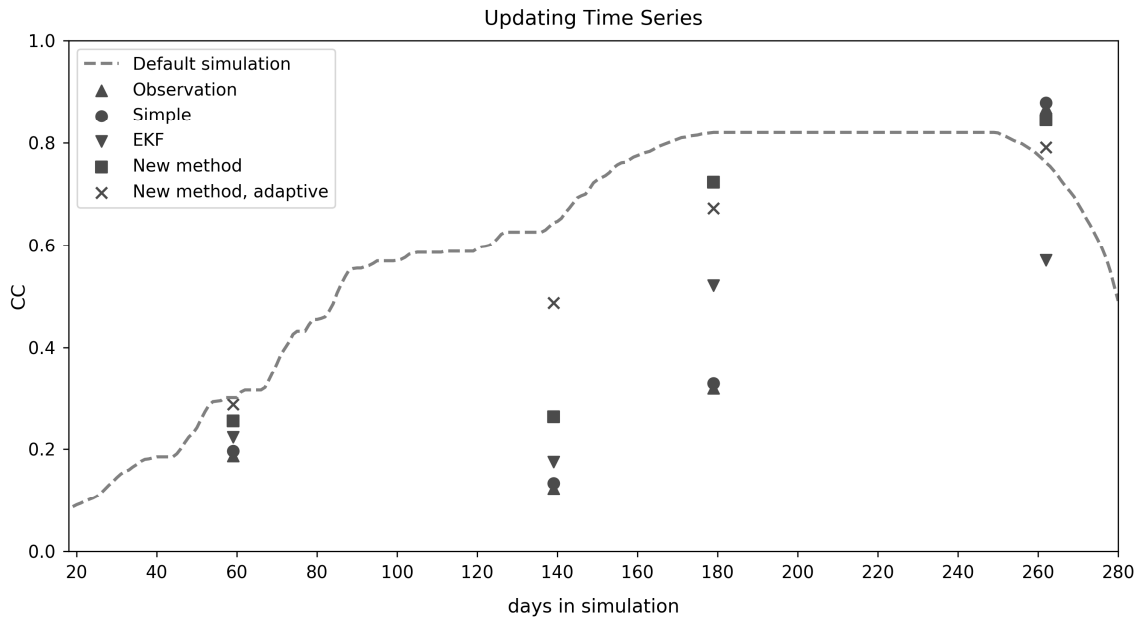
An explanation could be a bias toward drastic underestimation in some cases and overestimation in others. This kind of bias would not manifest itself in a metric like MPE, but may become obvious when looking at outputs individually. We therefore investigated the results of the fields with the worst predictions. Figure 10 shows an example for extreme over- and underestimations in one of the fields. Overall results using the EKF update had a more obvious bias towards underestimations ( $\leq -3$  t/ha). The new method also produced significant errors, but they tended to be slightly more evenly distributed and less dramatic in most cases. Nevertheless, both results demonstrate similar trends with regions of overestimation near the top left and right as well as the bottom left boundaries in Figure 10. The remaining parts were mostly underestimated, in particular near the center and at the right border of the field. These issues may be explained by mixed pixels and often observed low yields near the boundaries of the field that are not captured by the model.



**Figure 10.** Example map of the pixel-wise differences between the simulated and in situ yield values ( $< 0$  underestimation,  $> 0$  overestimation) for EKF updating (left) and adaptive PSO updating (right). Maps are smoothed with a  $3 \times 3$  mean filter to reduce noise.

The time series plot in Figure 11 may hint at causes behind different behavior of the updating techniques as well as the Sentinel-2 CC observations. As mentioned earlier, CC observations were often

very different from what the AquaCrop-OS model simulated by default. In this example, observations were much lower than the CC predicted by the model (without update), especially in the earlier growth stages. The EKF updates were therefore particularly low in many cases, while the new method often performed less dramatic corrections.



**Figure 11.** Example for time series of CC updating on the field-level showing the default simulation without updating and the corresponding updated CC values from simple updating (equivalent to observed CC), EKF updating, and the new method using  $\alpha = 5$  and the adaptive version.

#### 4. Discussion

In this study, we introduced a new updating method based on PSO to simulate wheat yield using the AquaCrop-OS model. As mentioned previously, our comparison addressed the results of different updating schemes applied to the same yield prediction scenario. We compared performances on different scales.

Overall, the new method performed better than a simple update and similar or better than the EKF update approach. It was particularly successful in reducing the bias in the estimation and outperformed both the simple update and the EKF update in this respect. However, the EKF is designed to correct random errors in the model state rather than systematic errors, so state augmentation or bias correction may improve its performance in this respect [64]. Furthermore, other techniques such as the ensemble Kalman filter or particle filter updates may better address non-linearities in the model.

Settings chosen for the update largely determined performance. The weighting factor  $\alpha$  had a significant impact and the optimal value was scale-dependent. This would pose the need for prior testing and calibration by the user, which is not ideal for most applications. Further research may reveal a best practice for the setting of  $\alpha$ , which may depend on the number and/or type of uncertainty PDFs or scale. The adaptive version, however, showed promising results via self-adjusting  $\alpha$ . It performed comparably to other settings on all scales, even though in most cases, it was not the best-performing updating scheme. Instead of guidance for the manual adjusting of  $\alpha$ , it may be preferable to improve the way is involved in the objective function.

Another observation in this context is that higher  $\alpha$  (values of five and higher) seemed to be advantageous on the field-level. One may argue that remote sensing observations are superior to the simulation and therefore a higher weighting automatically led to a significant improvement in predictions. This interpretation, however, contrasts with the poor results obtained when using remote sensing-related uncertainty alone or in the simple update. It is more likely that the reason lies in

model-related uncertainties, which often are distributed (skewed) normally around or close to the simulated CC value, while remote sensing observations differed significantly. Therefore, a larger weight on remote sensing-related uncertainty is required to “outweigh” the two similar model-related distributions. This interpretation is further supported by the fact that the field-level required higher  $\alpha$  values than the pixel-level. In our pixel-level implementation, we used the same model-related uncertainty PDFs (based on 10,000 simulations each), but considered only the immediate neighborhood of the pixel. Naturally, heterogeneity in such a small, spatially constrained sample is much smaller. Therefore, the distribution of the remote sensing-related uncertainties in the pixel neighborhood tends to be narrower than that of the entire field, consequentially leading to a smaller probability of an overlapping with the current simulated CC of AquaCrop-OS. The result is that most remote sensing predictions are assigned a Hellinger distance of 1 and therefore receive higher weights than on the field-level, even with a comparatively small  $\alpha$ . This situation may be addressed by altering the kernel bandwidth or analyzing different scales, numbers of pixels, etc. to possibly observe a relationship between the number of PDFs or sample size and the optimal  $\alpha$  values to choose. The other approach may be the use of an adaptive version that seemed to adjust to the different levels quite effectively, which was demonstrated in its comparable performance to the fixed versions.

Regarding uncertainties, all versions of the new method managed to incorporate different PDFs successfully. Using only remote sensing uncertainty led to poor results, probably because of the way the weighting was handled: in case of having only one uncertainty input, that distribution will always be assigned a weight of 1 (see Equation (18)). Results are expected to be close to those of a simple replacement update because the remote sensing-related PDF is the only one considered. An obtained optimal distribution would then obviously be located closely to the observed pixel or field mean CC value. In case of an actual normal distribution of values, it must be identical.

As expected, adding model-related uncertainty always improved results, except for adding the weather PDF, which was unable to enhance the performance consistently. A possible reason could be the similarity of the two distributions as indicated in Figure 2. Adding parameter-specific uncertainty PDFs rather than an all-in-one PDF may prove a better alternative in the future. This also highlights the importance of choosing the relevant type of uncertainty and its correct quantification. However, further research may be necessary to ensure a proper incorporation of all PDFs. This task would be closely linked to the improvement in optimizing  $\alpha$ , but may also involve increasing the  $3 \times 3$  pixel neighborhood with a larger rectangular or circular one.

Another observation was that  $R^2$  values were generally poor throughout all analyses, methods, and scales. Our investigation revealed that many fields contained regions of significant over- and underestimations. These frequent outliers may cause the low  $R^2$  values, even in cases with relatively good RMSE and MPE values. The new method seemed to be less likely to underestimate yield than EKF updating, but showed similar overall trends.

In this context, we also cannot exclude the possibility of significant bias in the Sentinel-2 CC data itself. The low errors mentioned in the original report on the algorithm suggest good performance [36]. However, the datasets used for both training and validation of the artificial neural networks in the algorithm were obtained from radiative transfer simulations rather than in situ experimental data may therefore not be representative for all situations. The fact that low  $R^2$  values were present in all updating results further indicates that they are probably unrelated to the updating methods themselves, but are connected to issues in the CC inputs or in situ yield data. This is further supported by the facts that pixel-level predictions were much worse and the simple update performed particularly poorly. Data on the pixel-level may therefore be unreliable. By increasing the detail in the process (i.e., the number of individual simulations per field), the broader trend is captured, but individual pixel-wise yield predictions are not matched. Nevertheless, as no external in situ measurements were available in our study area, we could not validate the quality conclusively.

Finally, AquaCrop-OS may also introduce unknown uncertainties. Even with careful calibration, it is difficult to scale such a model to represent conditions in a number of fields distributed over a large area, let alone different spatial scales.

On top of performance-related questions, pre-processing requirements are important for applications. The approach in the implementation described here requires extensive pre-processing. Our technique, however, is flexible regarding the type of uncertainty representation, future applications would not need to rely on a computation-intensive approach like Monte Carlo simulations and kernel density estimation. Instead, if previous knowledge about the characteristics of the model is available, simple functional relationships may represent uncertainties, for example.

## 5. Conclusions

We presented a method for updating model variables during simulation, taking into account uncertainties in both the model and the assimilated data. We described the method for assimilating remote sensing data into a dynamic crop model for improving yield estimation. The method proved to be comparable to other existing updating techniques, but was particularly capable of reducing bias in the estimations and managed to incorporate different sources of uncertainties.

We described the process specifically for the application in our study. The principle is, however, easily transferrable to other models or model variables. Its flexibility regarding the representation of uncertainties would also allow an adaptation to different situations where Monte Carlo simulations may not be feasible. Previous knowledge about the model in question would allow a representation of uncertainties by a simple functional relationship or a set of distributions.

Further research may be needed to analyze the behavior of the technique regarding different numbers of uncertainties and potential improvement in incorporating them into the updating scheme. Additionally, improvements to the distance metrics, objective function, and weighting may also be of interest as the applicability to different remote sensing datasets and pixel neighborhood sizes. It is also possible to analyze improvements regarding the optimal distribution, for example, by adding skewness or kurtosis. Furthermore, additional comparisons to other non-linear updating techniques such as the ensemble Kalman filter may provide valuable insights.

**Author Contributions:** Conceptualization, all authors; Methodology, M.P.W., T.S., A.T.; Validation, and M.P.W.; Formal Analysis, M.P.W.; Writing—Original Draft Preparation, M.P.W.; Writing—Review & Editing, all authors; Visualization, M.P.W.; Supervision, N.O. and T.S.; Project Administration, N.O.; Funding Acquisition, N.O. All authors have read and agreed to the published version of the manuscript.

**Funding:** This study was partly funded by the Federal Office for Agriculture and Food (BLE), Germany (grant no. 281702015). Financial support for publication fees was provided by Land Schleswig-Holstein within the funding program Open Access Publikationsfonds.

**Acknowledgments:** We thank the German Weather Service (DWD) for delivering the station and raster weather datasets. We further thank the anonymous farmers that provided in situ yield data.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Asseng, S.; Ewert, F.; Martre, P.; Rötter, R.P.; Lobell, D.B.; Cammarano, D.; Kimball, B.A.; Ottman, M.J.; Wall, G.W.; White, J.W.; et al. Rising temperatures reduce global wheat production. *Nat. Clim. Chang.* **2015**, *5*, 143–147. [[CrossRef](#)]
2. Food and Agriculture Organization of the United Nations. *Climate Change and Food Systems—Global Assessments and Implications for Food Security and Trade*; Elbehri, A., Ed.; FAO: Rome, Italy, 2015.
3. Alexandratos, N.; Bruinsma, J. *World Agriculture towards 2030/2050: The 2012 Revision*; Food and Agriculture Organization of the United Nations: Rome, Italy, 2012; ISBN 9789251086995.
4. United Nations, Department of Economic and Social Affairs. *Population Division World Population Prospects—The 2017 Revision, Key Findings and Advance Tables*; United Nations, Department of Economic and Social Affairs, Population Division: New York, NY, USA, 2015.

5. Basso, B.; Cammarano, D.; Carfagna, E. Review of Crop Yield Forecasting Methods and Early Warning Systems. In Proceedings of the First Meeting of the Scientific Advisory Committee of the Global Strategy to Improve Agricultural and Rural Statistics, Rome, Italy, 18–19 July 2013.
6. Jones, J.W.; Antle, J.M.; Basso, B.; Boote, K.J.; Conant, R.T.; Foster, I.; Godfray, H.C.J.; Herrero, M.; Howitt, R.E.; Janssen, S.; et al. Toward a new generation of agricultural system data, models, and knowledge products: State of agricultural systems science. *Agric. Syst.* **2017**, *155*, 269–288. [[CrossRef](#)] [[PubMed](#)]
7. Delécolle, R.; Maas, S.J.; Guérif, M.; Baret, F. Remote sensing and crop production models: Present trends. *ISPRS J. Photogramm. Remote Sens.* **1992**, *47*, 145–161. [[CrossRef](#)]
8. Rembold, F.; Atzberger, C.; Savin, I.; Rojas, O. Using low resolution satellite imagery for yield prediction and yield anomaly detection. *Remote Sens.* **2013**, *5*, 1704–1733. [[CrossRef](#)]
9. Maas, S.J. Use of remotely-sensed information in agricultural crop growth models. *Ecol. Modell.* **1988**, *41*, 247–268. [[CrossRef](#)]
10. Dorigo, W.A.; Zurita-Milla, R.; de Wit, A.J.W.; Brazile, J.; Singh, R.; Schaepman, M.E. A review on reflective remote sensing and data assimilation techniques for enhanced agroecosystem modeling. *Int. J. Appl. Earth Obs. Geoinf.* **2007**, *9*, 165–193. [[CrossRef](#)]
11. Kalman, R.E. A New Approach to Linear Filtering and Prediction Problems. *Trans. ASME—J. Basic Eng.* **1960**, *82*, 35–45. [[CrossRef](#)]
12. Evensen, G. The Ensemble Kalman Filter: Theoretical formulation and practical implementation. *Ocean Dyn.* **2003**, *53*, 343–367. [[CrossRef](#)]
13. Del Moral, P. Nonlinear filtering: Interacting particle resolution. *C. R. l'Académie des Sci.—Ser. I Math.* **1997**, *325*, 653–658. [[CrossRef](#)]
14. Kalman, R.E.; Bucy, R.S. New Results in Linear Filtering and Prediction Theory. *J. Basic Eng.* **1961**, *83*, 95–108. [[CrossRef](#)]
15. Kennedy, M.C.; O'Hagan, A. Bayesian calibration of computer models. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2001**, *63*, 425–464. [[CrossRef](#)]
16. Asseng, S.; Ewert, F.; Rosenzweig, C.; Jones, J.W.; Hatfield, J.L.; Ruane, A.C.; Boote, K.J.; Thorburn, P.J.; Rötter, R.P.; Cammarano, D.; et al. Uncertainty in simulating wheat yields under climate change. *Nat. Clim. Chang.* **2013**, *3*, 827–832. [[CrossRef](#)]
17. Warszawski, L.; Frieler, K.; Huber, V.; Piontek, F.; Serdeczny, O.; Schewe, J. The Inter-Sectoral Impact Model Intercomparison Project (ISI-MIP): Project framework. *Proc. Natl. Acad. Sci. USA* **2014**, *111*, 3228–3232. [[CrossRef](#)] [[PubMed](#)]
18. Hoffmann, H.; Zhao, G.; van Bussel LG, J.; Enders, A.; Specka, X.; Sosa, C.; Yeluripati, J.; Tao, F.; Constantin, J.; Raynal, H.; et al. Variability of effects of spatial climate data aggregation on regional yield simulation by crop models. *Clim. Res.* **2015**, *65*, 53–69. [[CrossRef](#)]
19. van Bussel LG, J.; Ewert, F.; Zhao, G.; Hoffmann, H.; Enders, A.; Wallach, D.; Asseng, S.; Baigorria, G.A.; Basso, B.; Biernath, C.; et al. Spatial sampling of weather data for regional crop yield simulations. *Agric. For. Meteorol.* **2016**, *220*, 101–115. [[CrossRef](#)]
20. Reichle, R.H. Data assimilation methods in the Earth sciences. *Adv. Water Resour.* **2008**, *31*, 1411–1418. [[CrossRef](#)]
21. Liu, X.P.; Li, X.; Peng, X.J.; Li, H.B.; He, J.Q. Swarm intelligence for classification of remote sensing data. *Sci. China Ser. D Earth Sci.* **2008**, *51*, 79–87. [[CrossRef](#)]
22. Shen, L.; Huang, X.; Fan, C. Double-group particle swarm optimization and its application in remote sensing image segmentation. *Sensors* **2018**, *18*, 1393. [[CrossRef](#)]
23. Bansal, S.; Gupta, D.; Panchal, V.K.; Kumar, S. Swarm intelligence inspired classifiers in comparison with fuzzy and rough classifiers: A remote sensing approach. *Commun. Comput. Inf. Sci.* **2009**, *40*, 284–294. [[CrossRef](#)]
24. Guo, C.; Zhang, L.; Zhou, X.; Zhu, Y.; Cao, W.; Qiu, X.; Cheng, T.; Tian, Y. Integrating remote sensing information with crop model to monitor wheat growth and yield based on simulation zone partitioning. *Precis. Agric.* **2017**, *19*, 1–24. [[CrossRef](#)]
25. Omkar, S.N.; Senthilnath, J.; Mudigere, D.; Manoj Kumar, M. Crop Classification using Biologically-inspired Techniques with High Resolution Satellite Image. *J. Indian Soc. Remote Sens.* **2008**, *36*, 175–182. [[CrossRef](#)]

26. Jin, M.; Liu, X.; Wu, L.; Liu, M. An improved assimilation method with stress factors incorporated in the WOFOST model for the efficient assessment of heavy metal stress levels in rice. *Int. J. Appl. Earth Obs. Geoinf.* **2015**, *41*, 118–129. [[CrossRef](#)]
27. Silvestro, P.C.; Pignatti, S.; Pascucci, S.; Yang, H.; Li, Z.; Yang, G.; Huang, W.; Casa, R. Estimating Wheat Yield in China at the Field and District Scale from the Assimilation of Satellite Data into the Aquacrop and Simple Algorithm for Yield (SAFY) Models. *Remote Sens.* **2017**, *9*, 509. [[CrossRef](#)]
28. Li, Z.; Wang, J.; Xu, X.; Zhao, C.; Jin, X.; Yang, G.; Feng, H. Assimilation of Two Variables Derived from Hyperspectral Data into the DSSAT-CERES Model for Grain Yield and Quality Estimation. *Remote Sens.* **2015**, *7*, 12400–12418. [[CrossRef](#)]
29. Son, N.T.; Chen, C.F.; Chen, C.R.; Chang, L.Y.; Chiang, S.H. Rice yield estimation through assimilating satellite data into a crop simulation model. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.—ISPRS Arch.* **2016**, *41*, 993–996. [[CrossRef](#)]
30. Jin, X.; Kumar, L.; Li, Z.; Xu, X.; Yang, G.; Wang, J. Estimation of winter wheat biomass and yield by combining the aquacrop model and field hyperspectral data. *Remote Sens.* **2016**, *8*, 972. [[CrossRef](#)]
31. Jibo, Y.; Haikuan, F.; Xiudong, Q. Monitor key parameters of winter wheat using Crop model. *IOP Conf. Ser. Earth Environ. Sci.* **2016**, *46*. [[CrossRef](#)]
32. Kottek, M.; Grieser, J.; Beck, C.; Rudolf, B.; Rubel, F. World Map of the Köppen-Geiger climate classification updated. *Meteorol. Zeitschrift* **2006**, *15*, 259–263. [[CrossRef](#)]
33. German Weather Service (DWD) DWD Climate Data Center. Available online: <https://cdc.dwd.de/portal/> (accessed on 17 July 2019).
34. Bundesanstalt für Geowissenschaften und Rohstoffe (BGR). *Bodenübersichtskarte der Bundesrepublik Deutschland 1:200000*; Bundesanstalt für Geowissenschaften und Rohstoffe (BGR): Hanover, Germany, 2018.
35. Doorenbos, J.; Pruitt, W.O. *Guidelines for Predicting Crop Water Requirements*; Food and Agriculture Organisation: Rome, Italy, 1977.
36. Weiss, M.; Baret, F. *S2ToolBox Level 2 Products: LAI, FAPAR, FCOVER—Version 1.1*; Institut National de la Recherche Agronomique (INRA): Paris, France, 2016.
37. Statistisches Landesamt Sachsen-Anhalt Tabellen Land- und Forstwirtschaft, Fischerei. Available online: <https://statistik.sachsen-anhalt.de/themen/wirtschaftsbereiche/land-und-forstwirtschaft-fischerei/tabelle-land-und-forstwirtschaft-fischerei/> (accessed on 17 July 2019).
38. Niedersachsen, L. *Bodennutzung und Ernte 2016*; Landesamt für Statistik Niedersachsen (LSN): Hanover, Germany, 2018; Volume 2016, p. 21.
39. Niedersachsen, L. *Bodennutzung und Ernte 2017*; Landesamt für Statistik Niedersachsen (LSN): Hanover, Germany, 2018; Volume 2017, p. 21.
40. Steduto, P.; Hsiao, T.C.; Raes, D.; Fereres, E. AquaCrop—The FAO crop model to simulate yield response to water: I. concepts and underlying principles. *Agron. J.* **2009**, *101*, 426–437. [[CrossRef](#)]
41. Raes, D.; Steduto, P.; Hsiao, T.C.; Fereres, E. AquaCrop—The FAO crop model to simulate yield response to water: II. main algorithms and software description. *Agron. J.* **2009**, *101*, 438–447. [[CrossRef](#)]
42. Hsiao, T.C.; Heng, L.; Steduto, P.; Rojas-Lara, B.; Raes, D.; Fereres, E. AquaCrop—The FAO crop model to simulate yield response to water: III. Parameterization and testing for maize. *Agron. J.* **2009**, *101*, 448–459. [[CrossRef](#)]
43. Steduto, P.; Hsiao, T.C.; Fereres, E.; Raes, D. Crop yield response to water. *FAO Irrig. Drain. Pap.* **2012**, *66*, 500.
44. Ines, A.V.M.; Das, N.N.; Hansen, J.W.; Njoku, E.G. Assimilation of remotely sensed soil moisture and vegetation with a crop simulation model for maize yield prediction. *Remote Sens. Environ.* **2013**, *138*, 149–164. [[CrossRef](#)]
45. Jones, J.W.; Hoogenboom, G.; Porter, C.H.; Boote, K.J.; Batchelor, W.D.; Hunt, L.A.; Wilkens, P.W.; Singh, U.; Gijsman, A.J.; Ritchie, J.T. The DSSAT cropping system model. *Eur. J. Agron.* **2003**, *18*, 235–265. [[CrossRef](#)]
46. Raes, D.; Steduto, P.; Hsiao, T.C.; Fereres, E. Chapter 3: Calculation Procedures. In *AquaCrop Reference Manual Version 6.0-6.1*; FAO: Rome, Italy, 2018.
47. Foster, T.; Brozović, N.; Butler, A.P.; Neale, C.M.U.; Raes, D.; Steduto, P.; Fereres, E.; Hsiao, T.C. AquaCrop-OS: An open source version of FAO’s crop water productivity model. *Agric. Water Manag.* **2017**, *181*, 18–22. [[CrossRef](#)]
48. Kennedy, J.; Eberhart, R.C.; Shi, Y. *Swarm Intelligence*, 1st ed.; Morgan Kaufmann: Burlington, MA, USA, 2001; ISBN 9781558605954.

49. Yang, X.-S. *Nature-Inspired Optimization Algorithms*, 1st ed.; Elsevier: London, UK, 2014; ISBN 978-0128100608.
50. Kennedy, J.; Eberhart, R. Particle Swarm Optimization. In Proceedings of the ICNN'95—International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; pp. 1942–1948.
51. Reynolds, C.W. Flocks, herds and schools: A distributed behavioral model. In Proceedings of the ACM SIGGRAPH '87 Conference, Anaheim, CA, USA, 27–31 July 1987; Volume 21, pp. 25–34.
52. Peram, T.; Veeramachaneni, K.; Mohan, C.K. Fitness-distance-ratio based particle swarm optimization. In Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No.03EX706), Indianapolis, IN, USA, 24–26 April 2003; pp. 174–181.
53. Mendes, R.; Kennedy, J.; Neves, J. Watch thy neighbor or how the swarm can learn from its environment. In Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No.03EX706), Indianapolis, IN, USA, 24–26 April 2003; pp. 88–94.
54. Poli, R.; Kennedy, J.; Blackwell, T. Particle swarm optimization, An overview. *Swarm Intell.* **2007**, *1*, 33–57. [[CrossRef](#)]
55. del Valle, Y.; Venayagamoorthy, G.K.; Mohagheghi, S.; Hernandez, J.-C.; Harley, R.G. Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power Systems. *IEEE Trans. Evol. Comput.* **2008**, *12*, 171–195. [[CrossRef](#)]
56. Liu, Y.; Ling, X.; Shi, Z.; Lv, M.; Fang, J.; Zhang, L. A survey on particle swarm optimization algorithms for multimodal function optimization. *J. Softw.* **2011**, *6*, 2449–2455. [[CrossRef](#)]
57. Clerc, M.; Kennedy, J. The Particle Swarm - Explosion, Stability, and Convergence in a Multidimensional Complex Space. *IEEE Trans. Evol. Comput.* **2002**, *6*, 58–73. [[CrossRef](#)]
58. Eberhart, R.C.; Shi, Y. Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization. In Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No.00TH8512), La Jolla, CA, USA, 16–19 July 2000; pp. 84–88.
59. Grewal, M.S.; Andrews, A.P. *Kalman Filtering—Theory and Practice Using MATLAB*, 4th ed.; John Wiley & Sons: Hoboken, NJ, USA, 2015; ISBN 9781118984963.
60. Hellinger, E. Neue Begründung der Theorie quadratischer Formen von unendlichvielen Veränderlichen. *J. für die reine und Angew. Math.* **1909**, *1909*, 210–271. [[CrossRef](#)]
61. Kullback, S.; Leibler, R.A. On Information and Sufficiency. *Ann. Math. Stat.* **1951**, *22*, 79–86. [[CrossRef](#)]
62. Bhattacharyya, A. On a measure of divergence between two statistical populations defined by their probability distributions. *Indian J. Stat.* **1946**, *7*, 401–406.
63. Wagner, M.P.; Taravat, A.; Oppelt, N. Particle swarm optimization for assimilation of remote sensing data in dynamic crop models. In *Proceedings of the SPIE 11149, Remote Sensing for Agriculture, Ecosystems, and Hydrology XXI, SPIE Remote Sensing 2019, Strasbourg, France, 9–12 September 2019*; Neale, C.M.U., Maltese, A., Eds.; SPIE: Bellingham, WA, USA, 2019.
64. De Lannoy, G.J.M.; Reichle, R.H.; Houser, P.R.; Pauwels, V.R.N.; Verhoest, N.E.C. Correcting for forecast bias in soil moisture assimilation with the ensemble Kalman filter. *Water Resour. Res.* **2007**, *43*. [[CrossRef](#)]

