ON-THE-GO CONDITION MAPPING FOR HARVESTING MACHINERY

T. Coen, J. De Baerdemaeker, W. Saeys

MeBioS-BIOSYST
Katholieke Universiteit Leuven
Leuven, Belgium

ABSTRACT

In recent years control systems have been used to alleviate the task of harvesting machinery operators. Automation allows the operator to spend more time on other tasks such as coordinating transport. Moreover, such control systems guarantee constant performance throughout the day whereas an operator gets tired.
The perfect control system anticipates on the harvest condition, just like an experienced operator would. The operator makes a visual assessment of the condition in terms of amongst others crop density in front of the machine, and acts upon it. At present automatic control systems do not have this information, which makes it hard to compete with experienced operators.
A lot of research has been performed into sensors to measure the harvest condition in front of the machine. In this paper a different approach is taken. The harvest properties in front of the machine are predicted based on earlier measurements near the current location. The technique is based on the assumption that the variation of these properties is continuous. As new measurements become available, the model of the field is updated.

Keywords: mapping, online, model, field, crop

INTRODUCTION

In recent years a lot of efforts have been made to implement capacity control systems on combine harvesters (Coen et al., 2008a, Coen, 2009). One of the approaches proposed for capacity control is Stochastic Model-based Predictive Control (Coen et al., 2010a,b). An advantage of this approach is that it can easily include predictions of process variables in its control strategy. The predictions generated in this paper can thus be easily used in the capacity controller. However, in order to design a capacity control system that has the same performance as the ideal operator, the system has to have access to the same information as the operator. The operator also sees the field in front of the combine harvester, and remembers how the combine harvester behaved in the field next to the current position. Some work has been done by Saeys et al. (2009) to measure the crop density in front of the combine with a laser scanner. Next to laser sensor approaches, it is also possible to use a pendulum, as described by

Hammen et al. (1991). The deviation of the pendulum from its equilibrium position is a measurement of the crop density. One of the disadvantages of this system is that it measures the crop density very close to the combine. This measurement is not much sooner than the measurement obtained from the CNH feed rate sensor, which measures the torque on the straw elevator and the header.

This paper focuses on adding a location specific memory to the control system. This way measurement such as crop density performed by the combine harvester on one location can be used to predict the crop and field conditions on a neighboring location. Based on these earlier measurements the field condition in front of the combine harvester will be predicted online. Examples of relevant field characteristics are:

- Field slope: It has been shown that the effective combine capacity decreases uphill, because the losses increase. If the slope information is available, the combine can slow down before the losses actually increase. If the system waits till the losses increase, it would always respond too late. Because of the delay it would take quite some time before the losses return to their normal levels.
- Crop density: The crop density varies throughout the field because of varying sunlight conditions, different soil types and varying fertilizer distribution. If the crop density in front of the machine is known, the controller can anticipate on crop density variations just like an operator would.
- Moisture content: The moisture content of the crop has significant impact on threshing and cleaning process. Normally the moisture content can only be measured in the straw elevator, at the end of the cleaning process. This means that it is impossible to adjust the threshing and cleaning system on time. By predicting the moisture content based on earlier measurements, it is possible to adjust the machine settings on time.

These are just a few examples of biological variables that can be predicted based on earlier measurements. The prediction is based on the assumption that all these variables can only vary continuously.

FIELD MODEL

The purpose of this field model is to predict biological characteristics of the crop and the field. These variables always show the following characteristics:

- The measurements are not exact, but corrupted by noise.
- To be predicted variables are linked to the field, and not to the machine state. In other words, if a variable has a certain value at spot A, it will have a similar value at all spots B which are in the vicinity of A.
- The variables vary continuously; no sudden changes of the variables are possible.
- The variable is a measurement of some biological quantity, which implies that one can assume a smooth first and second order derivative.

On the machine GPS position information is available such that the measurements can be related to a position in the field. The above requirements are important to select a suitable model structure. In order to use the prediction of a variable online, the model also needs to be calculated and adapted online. Computational requirements thus should be minimized. For this work the GPS location and the GPS altitude measurements are used. Based on these measurements it should be possible to predict the slope of the field. Note that the accuracy of any GPS system is much lower in the vertical direction than in the horizontal direction. The performance of the model may be improved by including the inclination measurements on the machine.

Model concept

A lot of different techniques can be used to use the previous measurements to predict the field characteristics in front of the machine. One could for instance store all the measured data in a database, and calculate a weighted average of the measurements surrounding the spot in front of the combine. The weighting function would than depend on the distance of each of the measurements to the location to predict. An alternative is to train a neural network to predict the variable based on the surrounding measurements. However, neither of these techniques takes into account that we are dealing with biological properties. The large variation of the distance to the nearest measurement may also have an adverse effect on the quality of the prediction.

The approach taken in this paper is to fit a model to the field measurements, and to update this model online. Since the shape of such a biological variable across a field may be very capricious, a numerical (instead of an analytical) model is chosen. First of all a grid is defined, with for instance a resolution of 1 meter. This leads to a matrix **F**, of which each element is a model value at one point on the grid. This is a very flexible model, since this can take on any desired shape.

Of course these model values need to be calculated. The calculation of the model values takes the following effects into account:

- Measurements: The measurements which have already been made, pull the model values in the direction of the measured values. Each measurement is attributed to one position on the grid (and thus in the matrix).
- Minimal derivative: In nature the curve of any object varies as little as possible. Things tend to level out. This also applies to the crop density and the terrain slope. During the modeling phase the derivative between two neighboring model values is penalized. In other words, the model surface can only deviate from a plane if measurements pull sufficiently hard.
- Minimal second derivative: In nature curves also vary smoothly. So, the first derivative should not change unless it is strictly necessary.

It is clear that all the above mentioned effects are important, but of course, they need to be balanced with each other, which is a tuning exercise.

Mathematical description

Let us now formalize the above presented approach in a mathematical description. First of all the measured values are discretized into a matrix. This transformation from geographically continuous measurement value to matrix element consists of the following steps:

1. Convert the GPS coordinates (latitude, longitude) into the WGS84 Geodetic Coordinate system. This way the position is expressed in meters.
2. The matrix dimensions are aligned with the North-South and East-West axis of the globe. The first value's x- and y-component are stored as *offset*$_x$ and *offset*$_y$. This only happens when the first measurement value is added.
3. A new coordinate $(x_0, y_0) = (x_0, y_0) - (offset_x, offset_y)$ is defined.
4. The continuous coordinates are then transformed into discrete coordinates. In this work the resolution of the measurement matrix is set to 5 meters. Of course this is a trade-off between computation time and accuracy. However most combine harvesters will normally not come closer than 5 meters to each other since the headers are often wider than 5 meters. If several measurements correspond with the same matrix element, the average measurement for that matrix element is taken.
5. The measurement matrix is automatically enlarged if measurements outside the current measurement matrix are encountered.

A model matrix, which contains the model values, with the same size as the measurement matrix is created. The model values are obtained from the following optimization problem:

$$
\begin{aligned}
\min f(a_{i,j}) = \; & \frac{\beta_t}{\gamma_t} \sum_{i=0}^{N_i} \sum_{j=0}^{N_j} c_{i,j} \left(a_{i,j} - t_{i,j} \right)^2 \\
& + \frac{\beta_{\delta,i}}{\gamma_\delta} \sum_{i=-1}^{N_i} \sum_{j=-1}^{N_j} \left(\frac{\delta a_{i,j}}{\delta i} \right)^2 + \frac{\beta_{\delta,j}}{\gamma_\delta} \sum_{i=-1}^{N_i} \sum_{j=-1}^{N_j} \left(\frac{\delta a_{i,j}}{\delta j} \right)^2 \\
& + \frac{\beta_{\delta^2,i}}{\gamma_{\delta^2}} \sum_{i=0}^{N_i} \sum_{j=0}^{N_j} \left(\frac{\delta^2 a_{i,j}}{\delta i^2} \right)^2 + \frac{\beta_{\delta^2,j}}{\gamma_{\delta^2}} \sum_{i=0}^{N_i} \sum_{j=0}^{N_j} \left(\frac{\delta^2 a_{i,j}}{\delta j^2} \right)^2
\end{aligned} \tag{1}
$$

with

$$
\begin{aligned}
\gamma_t &= \sum_{i=0}^{N_i} \sum_{j=0}^{N_j} c_{i,j} \\
\gamma_\delta &= \left(N_i + 2 \right)\left(N_j + 2 \right) \\
\gamma_{\delta^2} &= \left(N_i + 1 \right)\left(N_j + 1 \right)
\end{aligned} \tag{2}
$$

where $a_{i,j}$ the model value at position (i, j), $t_{i,j}$ the measured value at the same position and $c_{i,j}$ indicates if a measurement is available at the position (i, j). Of course, all these different terms need to be weighted, which is done by the parameters β_t, $\beta_{\delta,i}$, $\beta_{\delta,j}$, $\beta_{\delta^2,i}$ and $\beta_{\delta^2,j}$. Normally the weighting factors are taken identical for both directions (i and j). Thus there are actually three parameters to be tuned, namely β_t to weigh the measured values, β_δ to weigh the first order derivative and β_{δ^2} to weigh the second order derivative. Since the number of target value terms is unknown, all terms are first normalized with γ_t, γ_δ and γ_{δ^2}, and then weighted with their respective factors. One could add an additional requirement to these parameters without loss of generality, namely:

$$\beta_t + \beta_\delta + \beta_{\delta^2} = 1 \tag{3}$$

which reduces the number of parameters to be tuned to two. The first derivatives of equation (1) are defined as:

$$\frac{\delta a_{i,j}}{\delta i}(k,l) = \frac{a_{k+1,l} - a_{k,l}}{2} \tag{4}$$

$$\frac{\delta a_{i,j}}{\delta j}(k,l) = \frac{a_{k,l+1} - a_{k,l}}{2} \tag{5}$$

The second order derivatives are defined as:

$$\frac{\delta^2 a_{i,j}}{\delta i^2}(k,l) = \frac{1}{2}\left(\frac{a_{k+1,l} - a_{k,l}}{2} - \frac{a_{k,l} - a_{k-1,l}}{2}\right) \tag{6}$$

$$\frac{\delta^2 a_{i,j}}{\delta j^2}(k,l) = \frac{1}{2}\left(\frac{a_{k,l+1} - a_{k,l}}{2} - \frac{a_{k,l} - a_{k,l-1}}{2}\right) \tag{7}$$

Note in equation (1) that the first order derivative is also calculated for points surrounding the measurement (or model) matrix. In general, from equation (4), (5) , (6) and (7) follows that the model values near the area to be optimized, are also taken into account. Special precautions are needed to either:

- Use the model values outside the current measurement matrix
- Or, modify the objective function such that the derivatives only take values inside the measurement matrix into account. This is the default situation if no model values outside the measurement matrix are known.

Minimizing the problem shown in equation (1) comes down to solving the following system of equations:

$$\begin{cases} \dfrac{\delta f(a_{i,j})}{\delta a_{1,1}} & = & 0 \\ \dfrac{\delta f(a_{i,j})}{\delta a_{1,2}} & = & 0 \\ \vdots & & \\ \dfrac{\delta f(a_{i,j})}{\delta a_{2,1}} & = & 0 \\ \vdots & & \\ \dfrac{\delta f(a_{i,j})}{\delta a_{N_i,N_j}} & = & 0 \end{cases} \tag{8}$$

In other words, for each model value, a linear equation is added to the system. Take for instance the second order derivative along the *i* direction of equation (6). If one use the following notational convention:

$$\begin{aligned} a_l &= a_{k+1,l} \\ a_c &= a_{k,l} \\ a_r &= a_{k-1,l} \end{aligned} \tag{9}$$

where a_l stands for the left element, a_c the central element and a_r the right element. For every element of the model matrix, there is a term like equation (6) in equation (1). As shown in equation (8), the derivative of equation (1) needs to be calculated for each model value. If one only takes the second derivative along the *i* direction into account, this already leads to three components in the derivative of the model value $a_{i,j}$. After all, $a_{i,j}$ can be the left, central or right element of equation (6). First of all, calculate the derivatives of equation (6) to a_l, a_c and a_r, this yields:

Table 1: The contribution of equation (6) to the derivative of equation (1) to $a_{i,j}$. The contributions of a_l, a_c and a_r in equation (10) are listed separately.

	$a_{k-2,l}$	$a_{k-1,l}$	$a_{k,l}$	$a_{k+1,l}$	$a_{k+2,l}$
a_l			0.5	-1	0.5
a_c		-1	2	-1	
a_r	0.5	-1	0.5		
TOTAL	0.5	-2	3	-2	0.5

$$\frac{\delta}{a_l}\left(\left(\frac{\delta^2 a_{i,j}}{\delta i^2}\right)^2\right) = 0.5a_l - a_c + 0.5a_r$$

$$\frac{\delta}{a_c}\left(\left(\frac{\delta^2 a_{i,j}}{\delta i^2}\right)^2\right) = 2a_c - a_l - a_r \tag{10}$$

$$\frac{\delta}{a_r}\left(\left(\frac{\delta^2 a_{i,j}}{\delta i^2}\right)^2\right) = 0.5a_l - a_c + 0.5a_r$$

Now, let’s calculate the contribution of equation (6) to the derivative of equation (1) to $a_{i,j}$, with $a_{i,j}$ a model value in the middle of the model matrix (no boundary effects). This can be summarized in Table 1. The effect of the other terms of equation (1) is calculated analogously.

Example

To illustrate the algorithm GPS data from a field of a little over 1 ha in the neighborhood of Leuven is used. The field profile is modeled based on the coordinates and the altitude measurement of the GPS. Of course the same strategy can be used to model for instance crop density. The measured altitudes are shown in Figure 1.

If the resolution of the model grid is set to 5 meters, the measured field leads to a matrix of about 15 by 40 elements. To show the ability of the model to extrapolate, an area of 50 by 50 elements is modeled. This means this problem contains 2500 model values to be optimized (50 times 50). Given the resolution of 5 meters this corresponds to a field of about 6.25 ha. This problem takes 43 minutes to solve. It should however be noted that no efforts whatsoever were made to make a numerically and computationally optimal implementation. A significant speed-up is probably possible. The result is shown in Figure 2.

Note the model capability to make a smooth extrapolation of the field towards unvisited areas. Of course, this is only the best guess since there are no measurements, but it gives an idea what to expect in that area. This may be very useful to optimize the operation of the combine harvester. The model is not as smooth as it should be in the areas where there are measurements present. This can be improved by changing the tuning parameter of equation (1). In the example of the next section the effect of the tuning parameters is illustrated.

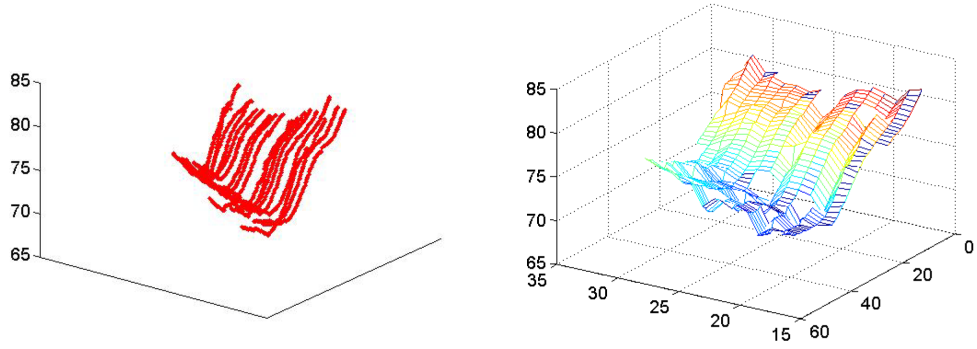


Figure 1: The measurements of the field used in the example. The field size is about 1ha, with a clear sink in the middle. On the left the measured points are plotted, on the right a mesh presentation is shown.

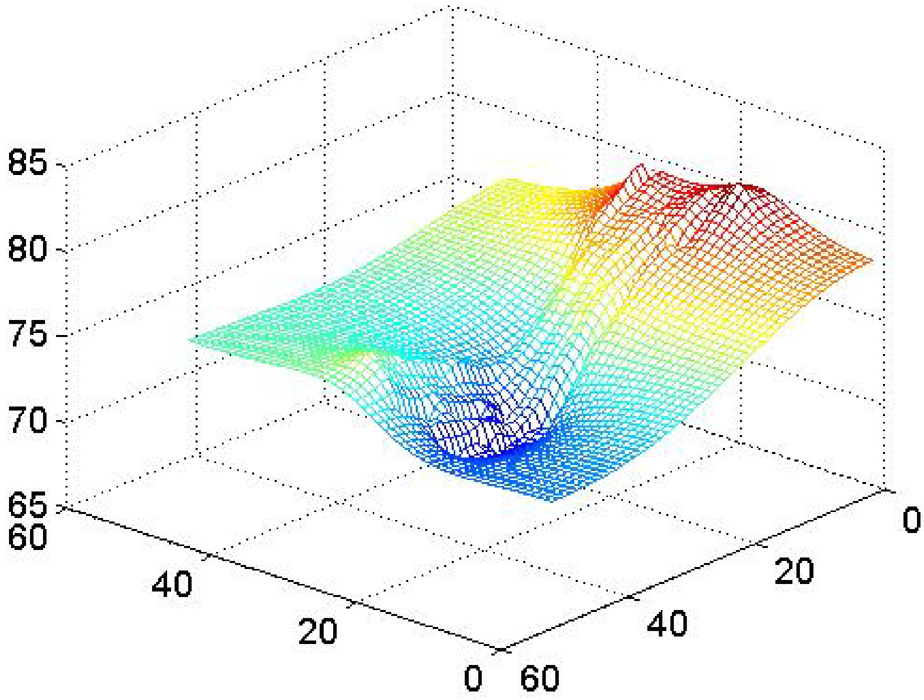


Figure 2: The calculated field model (off-line implementation)

ONLINE MODEL UPDATE

Online application of this technique adds two extra requirements to the algorithm, namely:

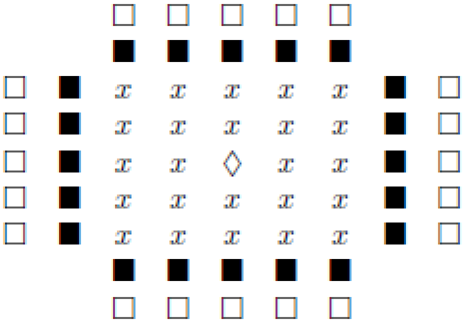
- The computational complexity has to be acceptable. Online application requires the algorithm to run at regular time intervals, for instance once every 10 seconds. Thus it has to be feasible to do so on hardware available on a machine.

- A system of updates has to be devised, to allow adding new information to the existing model.

Online model

The major computational cost is solving the system of equations. In order to be able to perform the model update online, the system size will have to be limited. The most logical approach is to only optimize the model values over part of the field. In other words, model values in a neighborhood around the current position are updated, and the other model values are kept as boundary conditions.

For the second order derivative to work correctly, at least 2 values in all directions around the current position need to be taken into account. This is illustrated below:



where ‘◊’ indicates the current combine position. ‘x’ are positions where the measurement value is taken into account, if present. In this case a range of 2 values around the current position is taken. ‘■’ denotes positions of which the model value (if present) is used as boundary conditions for the first and second order derivatives. ‘□’ denotes positions of which the model value (if present) is only used as boundary conditions for the second order derivative.

To have an idea of the feasibility of this model in an online environment, the execution times are recorded for different ranges of the local model. These tests were performed in Matlab on a Windows XP machine (Intel 2.0Gz, 1GB RAM). Note that no efforts were made to optimize the implementation of the algorithm. The results are shown in Table 2. Note that the execution time is recorded with Matlab, which implies that very low times are not accurately measured. This is probably the reason that the standard deviation remains constant for the first few model spans. The computation time is acceptable, but increases rapidly with the model span. The average execution time doubles for each increase (with 1) of the model span. This is to be expected, since the problem size (number of variables) increases quadratically with the model span, and the solution time of a system of equation is known to be O(x^3) with x the number of variables. The average execution times are plotted in Figure 3.

Tests have shown that a model span of 3 already yields quite good results (with a position resolution of 5 meters). Of course, if the resolution is increased, a larger model span will be needed to reach the same model characteristics.

Since the dataset contains a little over 2200 measurements, the online model calculation with updates takes less time to calculate the entire field model, than the offline version (the offline version takes a little over 1 second per measurement).

Table 2: Computation time required for one update step of the algorithm for different model spans. A model span 2 means that two measurement values in every direction around the combine position are taken into account.

Model	Time (s)				
Span	Mean	Std	Median	Min	Max
2	0.0108	0.0073	0.0150	0	0.0150
3	0.0230	0.0078	0.0160	0.0150	0.0320
4	0.0502	0.0068	0.0470	0.0460	0.0780
5	0.1253	0.0078	0.1250	0.1090	0.1410
6	0.2816	0.0135	0.2810	0.2500	0.3130
7	0.7439	0.0315	0.7500	0.6560	0.8130

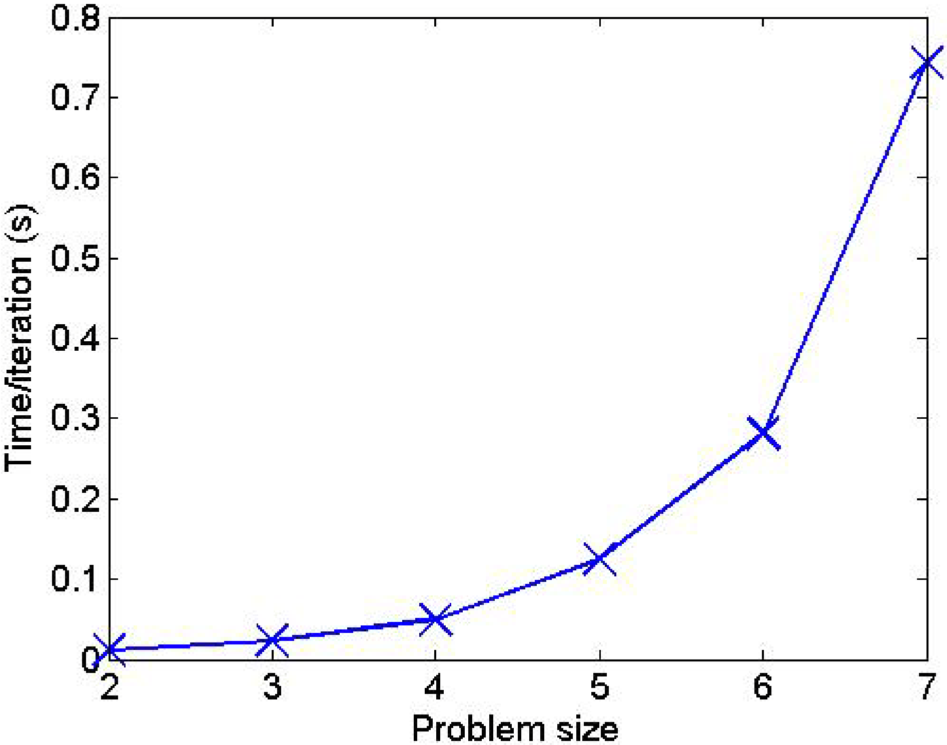


Figure 3: The average execution times as presented in Table 2. Note the steep increase in computation time in function of the problem size.

Example

For the online implementation a span 3 values is used. This means that in every time step the model values in an area of 7×7 $(2\times3+1)$ are optimized. There is little or no benefit in taking larger equation spans. This can be expected since the mathematical description defines the field behavior very locally. This is necessary to obtain a heavily nonlinear surface as shown in Figure 2.

The second order term is actually the most important term to describe the field. It makes sure that the field profile is smooth. However, the second order term also may give rise to instability. If one has two values, and one is larger than the other, the second order term will predict a (sometimes steep) slope. The first order term guarantees stability. This is illustrated in Figure 4, in which the model is shown with all the field data incorporated. On the left a very small first derivative weight is chosen, on the right a 5 times larger weight is used. The left model is clearly unstable, the right model predicts the field like it should.

Of course, another important component is the relative weight of the measurements. As mentioned above, the surface still appears a bit too sharp. Figure 5 shows the model with a reduced weight of the measurements. This clearly yields a smoother surface.

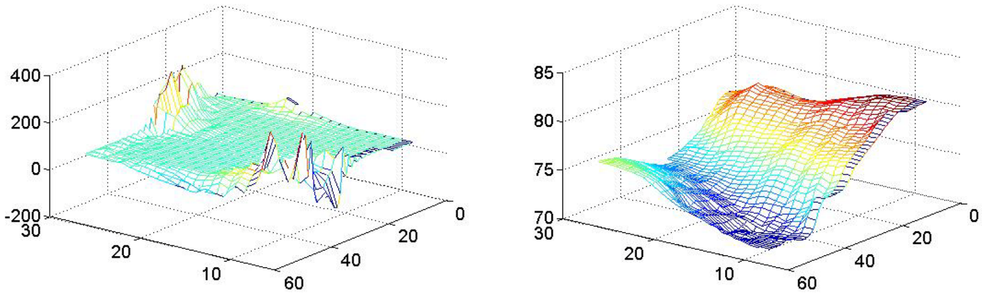


Figure 4: Two field models, left with a small first derivative term in the objective function, right with a larger first derivative term. The first derivative term clearly stabilizes the model.

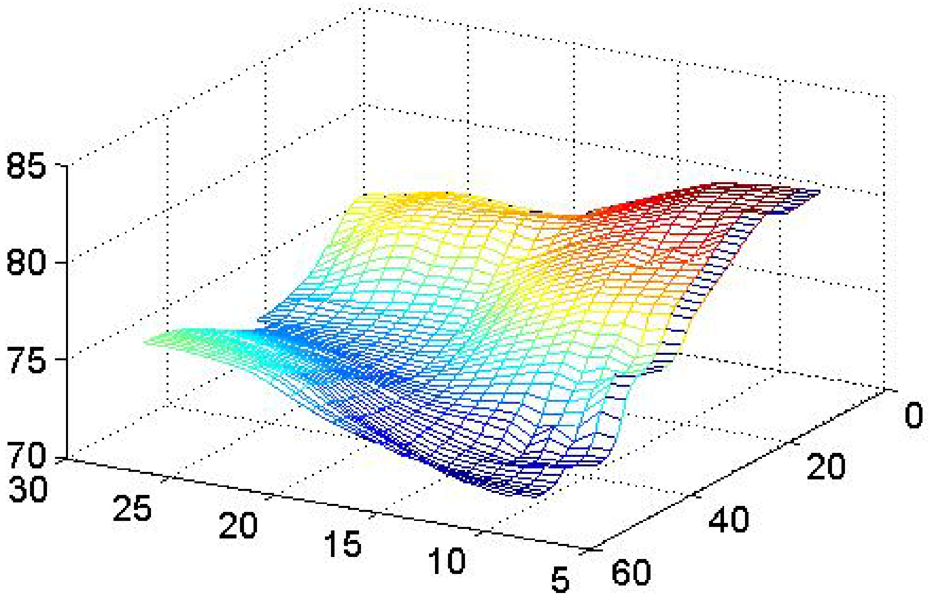


Figure 5: The field model with a reduced weight for the measurements. This leads to a smoother model

CONCLUSION

This paper presents a new approach to use the online measurements of crop and field properties to predict the crop and field condition in front of the machine. This model can also be updated recursively as more data becomes available. As this type of model makes more information available, it enables the development of more advanced controllers.

ACKNOWLEDGEMENTS

The authors would like to thank CNH Belgium for their support during this research. This work was performed at the division MeBioS of the Katholieke Universiteit Leuven. Tom Coen was supported by a scholarship of I.W.T.-Flanders (Institute for the Promotion of Innovation through Science and Technology) during this work, and is now managing director at *IN*DUCT. Josse De Baerdemaeker is a full professor at the Katholieke Universiteit Leuven. Wouter Saeys is funded as a Postdoctoral Fellow of the Research Foundation Flanders (FWO).

REFERENCES

Coen T., Saeys W., Missotten B., De Baerdemaeker J. 2008a. Cruise control on a combine harvester using MPC. Biosystems Engineering. 99(1), 47-55.

Coen T., Vanrenterghem A., Saeys W., De Baerdemaeker J. 2008b. Autopilot for a combine harvester. Computers and Electronics in Agriculture. 63(1):57-64.

Coen T. 2009. Model-based predictive control to increase the capacity of harvesting machinery in uncertain conditions. PhD dissertation, Faculty of Bioscience Engineering, KULeuven, Leuven, Belgium.

Coen T., De Baerdemaeker J., Saeys W. 2010a. Throughput control on a combine harvester using model-based predictive control, part I: process model. Submitted to Biosystems Engineering.

Coen T., De Baerdemaeker J., Saeys W. 2010b. Throughput control on a combine harvester using model-based predictive control, part II: control system. Submitted to Biosystems Engineering.

Hammen V., Ehlert D., Volkert C. 1991. Online-ertragsmessung in kleegras mit dem pendulum meter. Landtechnik. 54:156–163.

Saeys W., Lenaerts B., Craessaerts G., De Baerdemaeker J. 2009. Estimation of the crop density of small grains using LiDAR sensors. Biosystems Engineering. 102:22-30