2013 European Control Conference (ECC)
July 17-19, 2013, Zürich, Switzerland.

Moving Horizon Observation for Autonomous Operation of Agricultural Vehicles*

Janick V. Frasch[1], Tom Kraus[2], Wouter Saeys[2], and Moritz Diehl[1]

Abstract— **Varying terrain conditions influencing the ground friction challenge model-based control methods for precise autonomous driving of agricultural vehicles on off-road terrains. We apply moving horizon estimation (MHE) to cope with these uncertainties in a predictive control framework for autonomous driving of a sensor-equipped tractor using a (nonlinear) rigid-body dynamic model featuring a simple tire slip model. Using the ACADO Code Generation tool feedback times in the range of few miliseconds are achieved. Estimation results on real-world experimental data are presented.**

I. Introduction

Economic and other reasons lead to a desire for automated machine operation in the agricultural industry. While many work stages in agriculture are already automated, vehicle navigation in difficult conditions still relies on a human operator (think for example of a combine harvester). Modern GPS units providing precise position measurements pave the ground for automation of navigation [28]. Various control techniques have been applied to this problem; [27] gives a good overview of the most common approaches for path following in this context and compares their performance. These methods include geometric approaches, explicit control laws derived from kinematic models, and model-based linear quadratic regulators. According to [27] following non-straight paths still presents a challenge for the considered methods. Very recently the application of nonlinear model predictive control (MPC) schemes for autonomous navigation has been investigated in [1]. Using a combination of sensor systems including laser scanners, inertial measurement units, and a high-precision GPS, a navigation performance in the accuracy range of a human operator is reported. The nonlinear MPC scheme in that paper is based on a simple kinematic vehicle model, while state estimation is based on an Extended Kalman Filter (EKF).

Based on a similar kinematic model a combined nonlinear MHE/MPC scheme was presented in [9]. The authors use highly structure-exploiting autogenerated algorithms based on the Real-Time Iteration (RTI) scheme [5] for Bock's multiple shooting method [3]. An implementation is provided in the open source ACADO Code Generation tool [9], [13]. The computational aspects of this approach seem promising due to the efficiency of the exported code, which allows for short computation times and even significantly shorter feedback delays. As the exported C code is self contained and uses only static memory allocation, it is well suited for application on embedded hardware. [9] however uses a simplistic vehicle dynamics model and presents only simulation results.

We extend [9] by (a) augmenting the estimation/control vehicle model to a rigid body dynamics model and (b) presenting online moving horizon state and parameter estimation results on real-world data from an autonomously driving tractor in this paper.

The extended vehicle model merges dynamic effects typically considered in autonomous road vehicle guidance [15], [11] with the observed offroad driving behavior. Since the EKF (which is chosen for state observation, e.g., in [1]) can be considered a special case of the more general MHE algorithm for horizon length 1, estimation results from an MHE approach can in general be expected to be at least as good as the ones from an EKF approach. Recent computational advances like [18], [9] lead to real-time feasibility of MHE approaches even for medium-scale problems with fast-evolving dynamics, as they are present in many mechanical systems. Particularly due to potential synergies with an MPC controller based on the RTI scheme, MHE seems a well suited approach for the autonomous vehicle operation problem. Whether or not it is superior to other estimation techniques, like, e.g., particle filtering methods however is beyond the scope of this paper.

The structure of the paper reads as follows: Section II describes the tractor used for demonstration purposes and proposes a bicycle model for its control. Section III presents the algorithmic framework to solve this problem in real-time on vehicle embedded hardware. Section IV elaborates on the appropriate choice of weighting matrices for this estimation problem. Section V shows real-world estimation results from an autonomously offroad driving tractor, and section VI concludes the paper.

II. Tractor Model Formulation

A. The tractor

This paper presents an algorithm for state and parameter monitoring in autonomously driving agricultural vehicles. The control task is performed by an MPC algorithm based on the identical model as presented in Section II-B. MPC

* This research was supported by Research Council KUL: PFV/10/002 Optimization in Engineering Center OPTEC, GOA/10/09 MaNet and GOA/10/11 Global real- time optimal control of autonomous robots and mechatronic systems. Flemish Government: IOF/KP/SCORES4CHEM, FWO: PhD/postdoc grants and projects: G.0320.08 (convex MPC), G.0377.09 (Mechatronics MPC); IWT: PhD Grants, projects: SBO LeCoPro; Belgian Federal Science Policy Office: IUAP P7 (DYSCO, Dynamical systems, control and optimization, 2012-2017); EU: FP7-EMBOCON (ICT-248940), FP7-SADCO (MC ITN-264735), ERC ST HIGHWIND (259 166), Eurostars SMART, ACCM.

[1]Department of Electrical Engineering, KU Leuven. Kasteelpark Arenberg 10, B-3001 Leuven, Belgium

[2]Department of Mechatronics, Biostatistics and Sensors, KU Leuven. Kasteelpark Arenberg 30, B-3001 Leuven, Belgium



Fig. 1. The modified New Holland TZ25DA.

Parameter	Value	Description
m	730 kg	vehicle mass
I_z	350 kg m^2	moment of inertia
a	0.8	distance front axle to CoG
b	0.6	distance rear axle to CoG
c	0.95	distance GPS to CoG

TABLE I
IMPORTANT TRACTOR PARAMETERS.

controllers based on such models have been observed to perform well in semi-autonomous vehicle guidance settings, cf. [11]. To recover the full system state and environment-dependent parameters, we make use of an MHE scheme. To evaluate the performance of the proposed approach we use a modified demonstrator tractor on basis of a New Holland TZ25DA available at the Department for Mechatronics, Biostatistics and Sensors of KU Leuven, see Figure 1. A National Instruments NI PXI-8110 real-time computing unit featuring a 2.26 GHz Intel Core 2 processor has been added for embedded estimation and control. Further, the tractor has been equipped with a Septentrio AsteRx2eH PRO GPS for position and orientation measurements. Encoders measuring the rotational wheel speed at the rear tires are mounted on the tractor. A steering rate controller and a rear wheel speed controller following constant input references are available for control of the tractor. The front wheels are not driven in the considered configuration. Note that the steering rate controller was observed to react almost instantaneously, while the rear wheel speed reference is assumed to be constant for our purposes.

A list of parameters relevant for the proposed model in Section II-B that could be identified offline can be found in Table I.

B. Proposed Dynamic Model

We represent the vehicle's motion by its position and orientation in a global $X-Y$ coordinate frame, by its velocities in a local $x-y$ coordinate frame, and by its yaw rate, cf. Figure 2. A summary of all vehicle states,

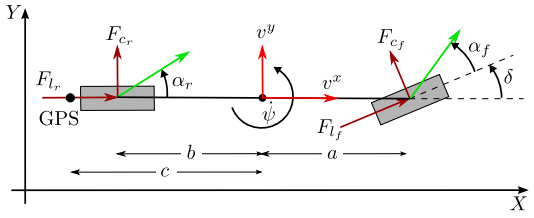


Fig. 2. Coordinate frame, forces, and slip angles of the bicycle vehicle model.

parameters, and control inputs is given in Table II. Roll, heave, and pitch motion of the chassis are neglected. This representation is commonly used in autonomous vehicle guidance, see, e.g., [11], [7], [16], [15]. Accordingly, the chassis equations of motion used in this paper read

$$\dot{X} = v^x \cos\psi - v^y \sin\psi, \tag{1a}$$
$$\dot{Y} = v^x \sin\psi + v^y \cos\psi, \tag{1b}$$
$$\dot{\psi} = \omega^z, \tag{1c}$$
$$\dot{v}^x = v^y \cdot \omega^z + \frac{1}{m}(2 \cdot F_{\mathrm{f}}^x + 2 \cdot F_{\mathrm{r}}^x), \tag{1d}$$
$$\dot{v}^y = -v^x \cdot \omega^z + \frac{1}{m}(2 \cdot F_{\mathrm{f}}^y + 2 \cdot F_{\mathrm{r}}^y), \tag{1e}$$
$$\dot{\omega}^z = \frac{1}{I^z}(a \cdot 2 \cdot F_{\mathrm{f}}^y - b \cdot 2 \cdot F_{\mathrm{r}}^y). \tag{1f}$$

The used parameters are explained in Table I. The factor 2 in front of the forces in Equations (1d)-(1f) stems from the simplification of a four-wheel to a bicycle model for computational efficiency. Note that the difference between the models is negligible under the targeted operating conditions of the tractor, cf. [14].

For the rear wheel forces it holds

$$F_{\mathrm{r}}^x = F_{\mathrm{r}}^{\mathrm{l}}, \qquad F_{\mathrm{r}}^y = F_{\mathrm{r}}^{\mathrm{c}},$$

while the front wheel forces are composed as

$$F_{\mathrm{f}}^x = F_{\mathrm{f}}^{\mathrm{l}} \cos\delta - F_{\mathrm{f}}^{\mathrm{c}} \sin\delta,$$
$$F_{\mathrm{f}}^y = F_{\mathrm{f}}^{\mathrm{l}} \sin\delta - F_{\mathrm{f}}^{\mathrm{c}} \cos\delta.$$

Here, $F_{\cdot}^{\mathrm{l}}$ and $F_{\cdot}^{\mathrm{c}}$ denote longitudinal and cornering forces, respectively; Section II-C gives more insight on the chosen tire model.

The steering angle δ is obtained by integration of the steering rate ω_δ, which is a control input to the system:

$$\dot{\delta} = \omega_\delta.$$

Since the GPS unit for position and orientation measurement is located behind the vehicle's center of gravity (CoG), we need to account for this in the measurements. It holds

$$X_{\mathrm{GPS}} = X - c \cdot \cos\psi \tag{2}$$
$$Y_{\mathrm{GPS}} = Y - c \cdot \sin\psi \tag{3}$$

The full list of measured states and inputs thus reads $[X_{\mathrm{GPS}}, Y_{\mathrm{GPS}}, \psi, \delta, \omega_\delta, v_{\mathrm{w}}]$. The MHE observer is used to recover the current velocities in the local coordinate frame and the tire-ground interaction parameters, as well as to filter the raw measurements against sensor noise.

State	Unit	Description
X	m	Global X position of vehicle's CoG
Y	m	Global Y position of vehicle's CoG
ψ	rad	Global Vehicle orientation
v^x	$\frac{\text{m}}{\text{s}}$	Total longitudinal velocity of vehicle
v^y	$\frac{\text{m}}{\text{s}}$	Total lateral velocity of vehicle
ω^z	$\frac{\text{rad}}{\text{s}}$	Vehicle's yaw rate
δ	rad	Steering angle

Parameter	Unit	Description
C_l	$\frac{\text{rad}}{\text{s}}$	Longitudinal tire stiffness
C_c	$\frac{\text{m}}{\text{s}}$	Cornering tire stiffness

Control	Range	Unit	Description
ω_δ	$[-0.8, 0.8]$	$\frac{\text{rad}}{\text{s}}$	Front steering angle
v_w	$[0.01, 10]$	$\frac{\text{m}}{\text{s}}$	Driving speed of rear wheels

TABLE II

STATES, PARAMETERS, AND CONTROL INPUTS OF THE TRACTOR MODEL.

C. Tire – Ground Interaction

Since we assume a rear-wheel driven vehicle the longitudinal tire forces are computed as

$$\begin{aligned} F_\text{f}^l &= 0, \\ F_\text{r}^l &= C_\text{l} \cdot \kappa = C_\text{l} \frac{v_\text{w} - v^x}{v^x}, \end{aligned}$$

where v_w is the driving speed of the rear wheels and C_l is the longitudinal tire stiffness, which can be seen as a linearization of a Pacejka-type tire model, cf. [4]; see [23] for details on tire modeling. Note that κ is typically referred to as the slip ratio.

The cornering tire forces are as well computed from a linearized tire model:

$$\begin{aligned} F_\text{f}^\text{c} &= -C_\text{f} \cdot \tan \alpha_\text{f}, \\ F_\text{r}^\text{c} &= -C_\text{r} \cdot \tan \alpha_\text{r}, \end{aligned}$$

where the side slip angles $\alpha_\cdot$ are computed as

$$\begin{aligned} \tan \alpha_\text{f} &= \frac{(v_y + l_f \cdot \omega_z) \cdot \cos \delta - v_x \cdot \sin \delta}{(v_y + l_f \cdot \omega_z) \sin \delta + v_x \cdot \cos \delta}, \\ \tan \alpha_\text{r} &= \frac{-l_r \cdot \omega_z + v_y}{v_x}, \end{aligned}$$

and the front and rear cornering stiffnesses are denoted by C_f and C_r, respectively. Experimental validation showed that under the targeted operating conditions the cornering stiffness is mostly influenced by the soil characteristics. We therefore assume a constant ratio between the front and rear cornering stiffness to increase observability. It holds $C_\text{r} := C_\text{c}$ and $C_\text{f} := C_\text{c}/r$, where r was identified to be approximately 3 for the considered hardware setup.

We choose a linearized tire model over a full Pacejka model for two reasons. Firstly, the latter one is a semi-empirical model and thus requires several parameters characterizing the tire which are simply not available for the considered tractor's tires. This includes parameters which do not get much excitation in the tractor's typical operating conditions and thus cannot be estimated well online either. Secondly, the Pacejka tire model is developed analyzing tire-road interactions [23]. Basic assumptions, most importantly the force generation by tire deformation on a rigid driving surface, typically do not hold in off-road driving. To the best of our knowledge it is not clear how well a Pacejka-type model reflects offroad driving characteristics.

[19] on the other hand proposes a tire-soil interaction model for offroad vehicles that only considers ground deformation. We however assume that a tractor's driving behavior will express characteristics from both tire and soil deformation. Since it is typically operated at moderate speeds we believe a tire force linearization sufficiently covers all appearing effects.

Even though we present a time-dependent vehicle model in this paper, it should be noted that for a practical application on commercial agricultural vehicles a spacial reformulation of the vehicle dynamics is desirable in a similar fashion as presented in [10].

III. Algorithmic Solution Framework

A. MHE Problem Formulation

We summarize differential states of the dynamic system 1 by $\xi : \mathbb{R} \to \mathbb{R}^{n_x}$, control inputs by $\nu : \mathbb{R} \to \mathbb{R}^{n_u}$ and parameters by $p \in \mathbb{R}^{n_p}$.

The dynamic system is then given by

$$\dot{\xi}(t) = f(\,\xi(t),\, \nu(t),\, p\,), \qquad t \in \mathcal{T} := [t_0, t_f]. \tag{4}$$

The observed system quantities $y(t)$ are summarized introducing a (possibly nonlinear) measurement function h. It holds

$$y(t) = h(\,\xi(t),\, \nu(t),\, p\,), \qquad t \in \mathcal{T}. \tag{5}$$

New observations $\tilde{y}^{t_k} \approx y(t_k)$ are assumed to arrive at discrete time points t_k. At each sampling time the MHE only takes a sequence of N consecutive, typically equidistant observations, w.l.o.g. called $t_0 < t_1 < \ldots < t_N =: t_f$, explicitly into account. These observations naturally define a receding time window $\mathcal{T} = [t_0, t_f]$ for the MHE, which is shifted forward at each sampling time. We accordingly summarize the finite horizon estimation problem to be solved repeatedly online as

$$\begin{aligned} \min_{\substack{\xi(\cdot),p,\\ \nu(\cdot)}} \quad & \left\| \begin{matrix} \xi(t_0) - \bar{\xi}^{t_0} \\ p - \bar{p}^{\,t_0} \end{matrix} \right\|_{P^{t_0}}^2 + \sum_{k=1}^{N} \left\| \tilde{y}^{t_k} - h(\xi(t_k), \nu(t_k), p) \right\|_{V^{t_k}}^2 \\ \text{s.t.} \quad & \dot{\xi}(t) = f\left(\xi(t), \nu(t), p\right) && \forall t \in \mathcal{T}, \\ & \xi^l(t) \le \xi(t) \le \xi^u(t) && \forall t \in \mathcal{T}, \\ & \nu^l(t) \le \nu(t) \le \nu^u(t) && \forall t \in \mathcal{T}, \\ & p^l \le p \le p^u. \end{aligned} \tag{6}$$

The objective function strives to minimize the deviation of the process output from the observed measurements, summarizing knowledge about states and parameters from observations before t_0 in a regularizing arrival cost that is computed independently, see Section IV. The norm $\|a\|_{\{A\}} := a^{\mathrm{T}} A\, a$ denotes the Euclidean norm. Symmetric positive semi-definite matrices are given by $P^{t_0} \in \mathbb{R}^{(n_x+n_p)\times(n_x+n_p)}$ and $V^{t_k} \in \mathbb{R}^{n_y \times n_y}$, respectively. Note that V^{t_k} may vary over the MHE horizon; more details on the choice of the weighting matrices will be given in Section IV. Functions $\xi^l(\cdot), \nu^l(\cdot), p^l(\cdot)$ denote (possibly time varying) lower bounds on states, controls and parameters, while $\xi^u(\cdot), \nu^u(\cdot), p^u(\cdot)$ denote the corresponding upper bounds.

B. Multiple Shooting Discretization

The time grid induced by $\{t_0, \ldots, t_N\}$ is used as a multiple shooting discretization grid for the dynamic optimization problem 6, as proposed in [2]. On each time interval $[t_i, t_{i+1}],\ 0 \leq i < N$, given an initial guess ξ_i^0, an initial value problem (IVP) is solved to obtain $\xi(t_{i+1})$. We choose constant base functions for control discretization on the shooting intervals in this paper. Additionally, state and control bounds are relaxed to the sampling time grid. Note that the analogon of this procedure for optimal control problems is known as Direct Multiple Shooting, see [3].

After discretization, we obtain a least-squares nonlinear programming (NLP) problem, which is solved by applying the generalized Gauss-Newton method. For an efficient numerical computation, the proposed solution algorithm relies heavily on structure exploitation, most importantly on a condensing technique for dimensionality reduction (cf. [20]).

C. The Real-Time Iteration Scheme for MHE

In an MHE context estimation problem 6 needs to be solved repeatedly for a sliding time window $\mathcal{T}$. A new state observation $\tilde{y}^{t_f}$ is added, while $\tilde{y}^{t_0}$ is aggregated into the arrival cost; all intermediate observations are merely shifted.

This similarity of subsequent problems can be exploited for efficiency by the so called Real-Time Iteration (RTI) scheme. Originally proposed for nonlinear MPC in [5] it has recently been adapted for nonlinear MHE problems [17], [18]. Building on Bock's Multiple Shooting method the RTI scheme performs only one Gauss-Newton iteration per sampling time, thus permitting real-time execution even for fast sampling times. Moreover, even before obtaining the most current observation $\tilde{y}^{t_f}$, the next estimation problem is prepared in the so-called preparation phase. This includes integration and sensitivity generation (typically the computationally most expensive part) as well as the condesing step. Once $\tilde{y}^{t_f}$ becomes available, only a condensed QP, typically similar to the one solved in the previous iteration, needs to be solved to obtain the full current state and parameter estimate $[\,\xi(t_f),\, p\,]$. Exploiting the QP similarity by the so-called parametric active set strategy (cf. [8]), the time delay between the availability of the process observation and the estimation result typically is very short, both in relation to the sampling time and even more in relation to a conventional multiple shooting based estimation approach. More details on the RTI scheme can be found in [5]; theoretical properties, including a proof of contractivity, are shown in [6] in the context of nonlinear MPC; more details on the specific implementation used in this paper are found in [9].

D. Code Generation

In order to provide sufficiently fast feedback for the real-time implementation of the proposed MPC problem, we make use of automatic source code generation [22], [21], [12]. We use an implementation for auto-generation of nonlinear MPC algorithms, the ACADO Code Generation tool [12]. This open source software for automatic control has recently been extended to support code generation for MHE problems [9]. It exports plain and self-contained C code based on a symbolic representation of the estimation problem, removing all redundant computations. Exploiting sparsity patterns and fixing dimensions, the exported code uses static references only, making it well suited for fast application on vehicle embedded hardware.

IV. Choice of Weighting Matrices

In the presence of model-plant mismatch and sensor noise the measurements $\tilde{y}^{t_k}$ can in general not be fitted exactly by the underlying system (4,5). To achieve a stable estimation performance weighting matrices P^{t_0} and V^{t_k} need to be chosen with particular care.

A well established approach is to choose V^{t_k} with respect to the sensor uncertainties, characterized by their standard deviation $R(t_k)$, as $V^{t_k} := \left(R(t_k)^{\mathrm{T}}\, R(t_k)\right)^{-1}$, and to choose P^{t_0} with respect to the observed state range and parameter variation, characterized by their standard deviations $Q_x(t_0)$ and $Q_{\mathrm{p}}(t_0)$, as $P^{t_0} = (\mathrm{diag}[Q_x(t_0), Q_{\mathrm{p}}(t_0)])^{-1}$. Under the assumption that (a) the sensor noise is a normal distributed random variable with covariance $R(t_k)^{\mathrm{T}}\, R(t_k)$ around the true system output $y(t_k)$, and (b) the initial system state $\xi(t_0)$ and the parameters p are normal distributed with uncorrelated covariances $Q_\xi(t_0)^{\mathrm{T}}\, Q_\xi(t_0)$ and $Q_{\mathrm{p}}(t_0)^{\mathrm{T}}\, Q_{\mathrm{p}}(t_0)$ around $\bar{\xi}^{t_0}$ and $\bar{p}^{t_0}$, it has been shown that the solution of (6) is the Maximum Likelihood (ML) estimator [18], even in the presence of active constraints, cf. [26].

Note that this choice of weighting matrices particularly implies that obviously wrong or missing measurements are simply ignored, as a sensor uncertainty of ∞ means a zero weight in the corresponding V^{t_k} matrix.

We choose to update the arrival cost information $\bar{\xi}^{t_0}, \bar{p}^{\,t_0}$ and P^{t_0} summarizing knowledge of and confidence in the past measurements using so-called smoothed EKF updates [18], [25] based on sensitivity information computed during MHE preparation phase. This is motivated by the observation that the classical EKF is equivalent to MHE on a horizon of length one with smoothed EKF-updates [25], [17].

V. Experimental Results

We demonstrate the capabilities of the proposed estimation approach in an offroad driving scenario of the tractor presented in Section II-A. The trajectory driven by the tractor

can be seen from Figure 3. The speed controller reference value of the tractor was constantly set to 6 km/h and the steering input was defined offline. All computations were performed within NI LabVIEW. The actual MHE module was run as compiled exported C-code called through LabVIEW's DLL-Interface.

In accordance with Section IV weighting matrices V^{t_k} were chosen in corresponding to sensor a standard deviation matrix $R = \text{diag}[0.02, 0.02, 0.01745, 0.05, 0.18, 0.1745]$. The arrival cost matrix P^0 was chosen with respect to an initial state and parameter uncertainty matrices $Q_\xi = \text{diag}[10, 10, 3.14, 1.5, 0.3, 0.5, 0.4]$ and $Q_\text{p} = \text{diag}[1, 0.7]$. The smooting matrices for the smoothed EKF were chosen as $W_\xi = \text{diag}[5, 5, 0.1745, 0.2, 0.2, 0.2, 0.09]$ and $W_\text{p} = \text{diag}[0.1, 0.1]$.

An estimation horizon of 20 intervals with a total length of 4 s at a sampling time of 200 ms is used. Integration of the dynamic system and the corresponding variational differential equations is done by a implicit Runge-Kutta integrator of order 2 (see [24]) on a grid of 80 integrator steps. To ensure a stable observer performance, 5 RTI-type iterations were performed per sampling time. Still, the proposed approach was observed to be well real-time feasible for the chosen sampling time of 200 ms with total execution times of the auto-generated code lying clearly below 5 ms.

Figures 3 and 4 show the estimation results. Position, orientation and steering angle are recovered well by the estimation scheme. A constant cornering stiffness parameter different from the arbitrarily chosen initialisation point was found when observable. Note that on straight parts this value drops off slightly due to unobservability arising from non-excitation. A longitudinal stiffness parameter could as well be found oscillating about a constant value of roughly 5300 N.

We observed that the estimation results from the arrival cost alone (which corresponds to a pure EKF) exhibited a significantly worse estimation performance, likely due to the short estimation horizon. Drawing a general conclusion from this is however impossible, since the estimation performance is influenced strongly by the EKF tuning.

VI. Conclusions

In this paper we presented a nonlinear MHE scheme for the combined state and parameter estimation of autonomously driving agricultural vehicles. A simplified model covering the most important dynamic effects of driving in varying terrain conditions is described. Estimation results on real-world data for an autonomously driving tractor were shown to yield an accurate and stable estimation performance. Closed loop simulations including a nonlinear MPC controller based on the same model as presented in Section II are subject of ongoing research.

Acknowledgment

The authors thank Mario Zanon and Sebastién Gros for fruitful discussions on the research that lead to this paper.

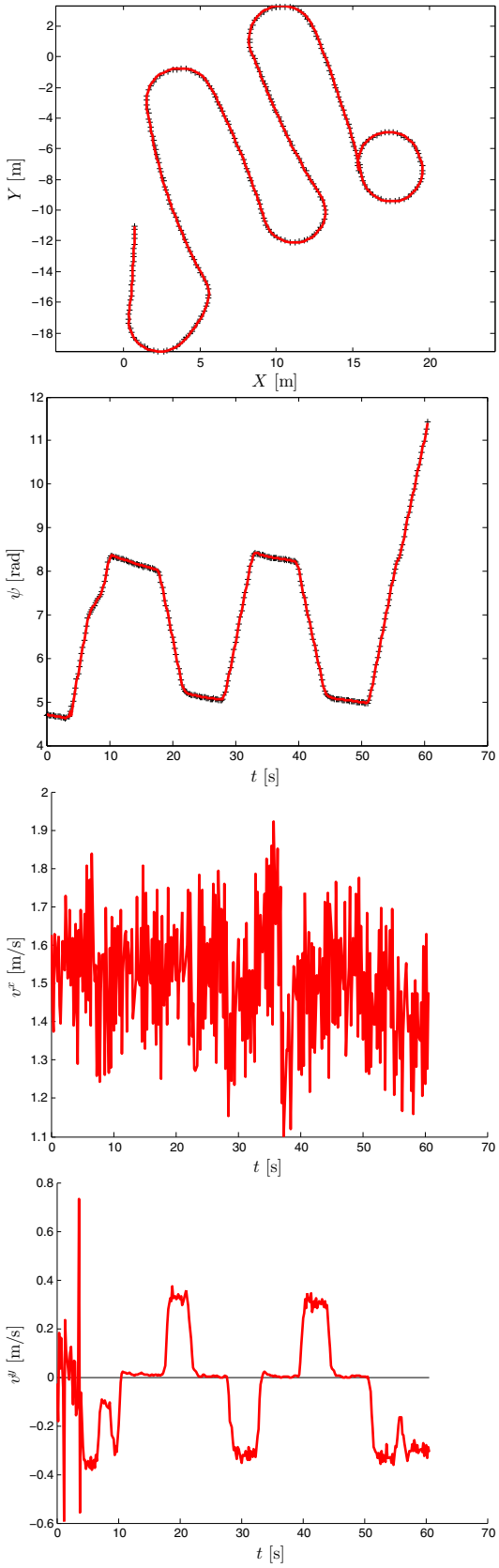


Fig. 3. State and parameter estimation from a U-turn scenario (I).

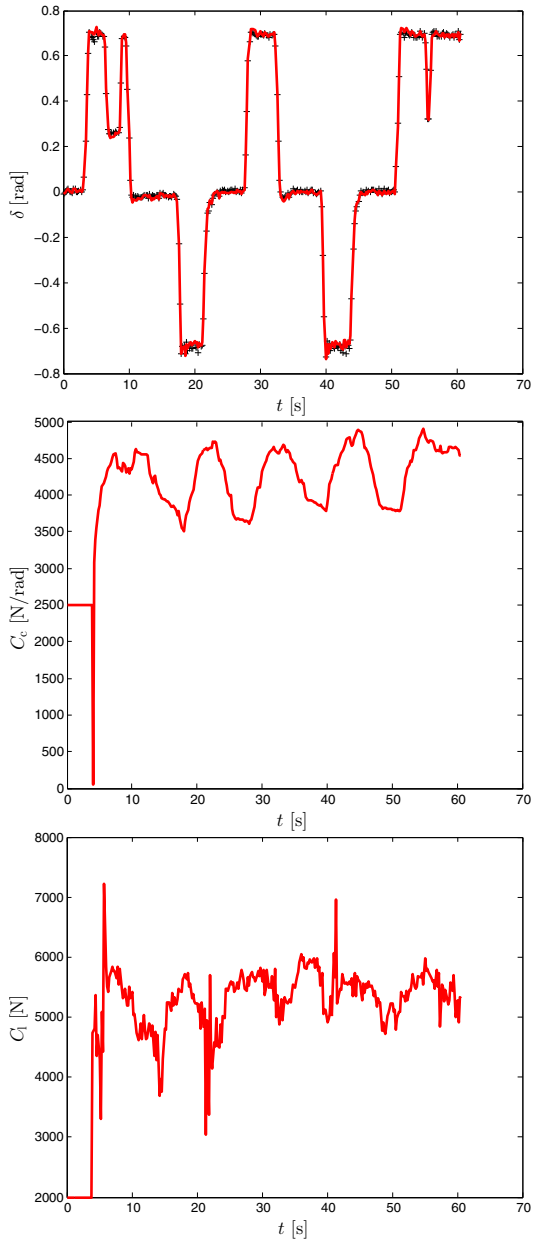


Fig. 4. State and parameter estimation from a U-turn scenario (II).

REFERENCES

[1] J. Backman, T. Oksanen, and A. Visala. Navigation system for agricultural machines: Nonlinear model predictive path tracking. *Computers and Electronics in Agriculture*, 82:33–43, 2012.

[2] H.G. Bock. Recent advances in parameter identification techniques for ODE. In P. Deuflhard and E. Hairer, editors, *Numerical Treatment of Inverse Problems in Differential and Integral Equations*. Birkhäuser, Boston, 1983.

[3] H.G. Bock and K.J. Plitt. A multiple shooting algorithm for direct solution of optimal control problems. In *Proceedings 9th IFAC World Congress Budapest*, pages 243–247. Pergamon Press, 1984.

[4] C.R. Carlson and J.C. Gerdes. Consistent nonlinear estimation of longitudinal tire stiffness and effective radius. *IEEE Transactions on Control Systems Technology*, 13(6):1010–1020, 2005.

[5] M. Diehl, H.G. Bock, J.P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer. Real-time optimization and Nonlinear Model Predictive Control of Processes governed by differential-algebraic equations. *Journal of Process Control*, 12(4):577–585, 2002.

[6] M. Diehl, R. Findeisen, and F. Allgöwer. A Stabilizing Real-time Implementation of Nonlinear Model Predictive Control. In L. Biegler, O. Ghattas, M. Heinkenschloss, D. Keyes, and B. van Bloemen Waanders, editors, *Real-Time and Online PDE-Constrained Optimization*, pages 23–52. SIAM, 2007.

[7] P. Falcone, F. Borrelli, J. Asgari, H.E. Tseng, and D. Hrovat. Low complexity MPC schemes for integrated vehicle dynamics control problems. 9^{th} *International Symposium on Advanced Vehicle Control*, 2008.

[8] H. J. Ferreau, H. G. Bock, and M. Diehl. An online active set strategy to overcome the limitations of explicit MPC. *International Journal of Robust and Nonlinear Control*, 18(8):816–830, 2008.

[9] H.J. Ferreau, T. Kraus, M. Vukov, W. Saeys, and M. Diehl. High-speed moving horizon estimation based on automatic code generation. In *Proceedings of the 51th IEEE Conference on Decision and Control (CDC 2012)*, 2012.

[10] J. V. Frasch, A. J. Gray, M. Zanon, H. J. Ferreau, S. Sager, F. Borrelli, and M. Diehl. An Auto-generated Nonlinear MPC Algorithm for Real-Time Obstacle Avoidance of Ground Vehicles. In *Proceedings of the European Control Conference*, 2013.

[11] Y. Gao, A. Gray, J. V. Frasch, T. Lin, E. Tseng, J.K. Hedrick, and F. Borrelli. Spatial predictive control for agile semi-autonomous ground vehicles. In *Proceedings of the 11th International Symposium on Advanced Vehicle Control*, 2012.

[12] B. Houska, H.J. Ferreau, and M. Diehl. ACADO Toolkit – An Open Source Framework for Automatic Control and Dynamic Optimization. *Optimal Control Applications and Methods*, 32(3):298–312, 2011.

[13] B. Houska, H.J. Ferreau, and M. Diehl. An Auto-Generated Real-Time Iteration Algorithm for Nonlinear MPC in the Microsecond Range. *Automatica*, 47(10):2279–2285, 2011.

[14] F. Kehrle, J. V. Frasch, C. Kirches, and S. Sager. Optimal control of formula 1 race cars in a VDrift based virtual environment. In S. Bittanti, A. Cenedese, and S. Zampieri, editors, *Proceedings of the 18th IFAC World Congress*, pages 11907–11912, 2011.

[15] U. Kiencke and L. Nielsen. *Automotive Control Systems*. Springer, 2005.

[16] C. Kirches, H.G. Bock, J.P. Schlöder, and S. Sager. Block structured quadratic programming for the direct multiple shooting method for optimal control. *Optimization Methods and Software*, 26(2):239–257, April 2010.

[17] T. Kraus. Real-Time State And Parameter Estimation for NMPC-Based Feedback Control With Application To The Tennessee Eastman Benchmark Process. Master's thesis, University of Heidelberg, 2007.

[18] P. Kühl, M. Diehl, T. Kraus, J. P. Schlöder, and H. G. Bock. A real-time algorithm for moving horizon state and parameter estimation. *Computers & Chemical Engineering*, 35(1):71–83, 2011.

[19] A.T. Le, D.C. Rye, and H.F. Durrant-Whyte. Estimation of track-soil interactions for autonomous tracked vehicles. In *Proceedings of the International Conference in Robotics and Automation*, 1997.

[20] D.B. Leineweber, I. Bauer, A.A.S. Schäfer, H.G. Bock, and J.P. Schlöder. An Efficient Multiple Shooting Based Reduced SQP Strategy for Large-Scale Dynamic Process Optimization (Parts I and II). *Computers and Chemical Engineering*, 27:157–174, 2003.

[21] J. Mattingley and S. Boyd. *Convex Optimization in Signal Processing and Communications*, chapter Automatic Code Generation for Real-Time Convex Optimization. Cambridge University Press, 2009.

[22] T. Ohtsuka and A. Kodama. Automatic code generation system for nonlinear receding horizon control. *Transactions of the Society of Instrument and Control Engineers*, 38(7):617–623, 2002.

[23] H. B. Pacejka. *Tyre and Vehicle Dynamics*. Elsevier, 2006.

[24] R. Quirynen, M. Vukov, and M. Diehl. Auto Generation of Implicit Integrators for Embedded NMPC with Microsecond Sampling Times. In Mircea Lazar and Frank Allgöwer, editors, *Proceedings of the 4th IFAC Nonlinear Model Predictive Control Conference*, 2012.

[25] D. Robertson. *Development and Statistical Interpretation of Tools for Nonlinear Estimation*. PhD thesis, Auburn University, 1996.

[26] D.G. Robertson and J.H. Lee. A moving horizon approach for least-squares estimation. *AIChe Journal*, 42:2209–2224, 1996.

[27] J.M. Snider. Automatic steering methods for autonomous automobile path tracking. *Technical Report CMU-RI-TR-09-08, Robotics Institute, Carnegie Mellon University*, 2009.

[28] J.N. Wilson. Guidance of agricultural vehicles a historical perspective. *Computers and Electronics in Agriculture*, 25:3–9, 2000.