# Kalman and particle filtering methods for full vehicle and tyre identification

## Karol Bogdanski & Matthew C. Best

Published online: 14 Jun 2017.

Submit your article to this journal ⃗

View related articles ⃗

View Crossmark data ⃗

IA*VSD*

Taylor & Francis
Taylor & Francis Group

🔓 OPEN ACCESS  Check for updates

# Kalman and particle filtering methods for full vehicle and tyre identification

Karol Bogdanski and Matthew C. Best

Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough, UK

**ABSTRACT**

This paper considers identification of all significant vehicle handling dynamics of a test vehicle, including identification of a combined-slip tyre model, using only those sensors currently available on most vehicle controller area network buses. Using an appropriately simple but efficient model structure, all of the independent parameters are found from test vehicle data, with the resulting model accuracy demonstrated on independent validation data. The paper extends previous work on augmented Kalman Filter state estimators to concentrate wholly on parameter identification. It also serves as a review of three alternative filtering methods; identifying forms of the unscented Kalman filter, extended Kalman filter and particle filter are proposed and compared for effectiveness, complexity and computational efficiency. All three filters are suited to applications of system identification and the Kalman Filters can also operate in real-time in on-line model predictive controllers or estimators.

## Nomenclature

### *Abbreviations*

CAN    Controller Area Network
CG    Centre of gravity
$E$    Percentage explanation of accuracy (match between modelled and measured variables)
EKF    Extended Kalman filter
$h^+$    Over-parametrised identification case (IEKF)
IEKF    Identifying extended Kalman filter
IN    Inertial Navigation
PF    Particle filter
UKF    Unscented Kalman filter

### *Identifying filters*

**f**    Continuous nonlinear system state derivatives vector (model)
**F**    State derivative (system model) Jacobian

---

**CONTACT** Matthew C. Best ✉ m.c.best@lboro.ac.uk

| | |
|---|---|
| **h** | Continuous nonlinear system output vector (model) |
| **H** | Output model Jacobian |
| $n_p$ | Number of particles in PF (tuning parameter) |
| **P** | State error covariance matrix |
| $\mathbf{P}_{xy}$ | State-output cross correlation matrix in UKF |
| $\mathbf{P}_{yy}$ | Output error covariance matrix in UKF |
| **Q** | Model error covariance matrix |
| **R** | Output (measurement) covariance matrix |
| $T$ | Discrete time step |
| $u[a,b]$ | Uniform distribution over range $a$, $b$ for particle generation |
| **u** | Input vector of continuous nonlinear state-space system |
| $W$ | Weighting of sigma point (UKF) or Particle (PF) |
| **x** | State vector of continuous nonlinear state-space system |
| **y** | Output (measurement) vector of continuous nonlinear state-space system |
| **z** | Combined state and parameter vector |
| $\varphi$ | Output error probability density function (PF) |
| $\kappa$ | UKF parameter |
| **θ** | Model parameter vector of continuous nonlinear state-space system |
| $\rho$ | EKF and UKF tuning parameter |
| $\sigma$ | Output error variance (PF) |
| $\upsilon$ | Output (measurement) error vector |
| **ω** | State propagation and modelling error vector |
| $\Upsilon$ | Output estimate from Sigma point (UKF) or Particle (PF) |
| $\chi$ | Sigma point (UKF) or Particle (PF) |

## *Vehicle model*

| | |
|---|---|
| $a$, $b$ | Front, rear axle distances from vehicle CG (m) |
| $B_\phi$ | Roll damping (Nms/rad) |
| $c$ | Vehicle half-track (m) |
| $C_\alpha$ | Tyre cornering stiffness (N/rad) |
| $C, D, E$ | Pacejka magic formula coefficients |
| $F_{tx}, F_{ty}$ | Longitudinal, lateral tyre forces in tyre axis system (N) |
| $F_x, F_y$ | Longitudinal, lateral tyre forces in vehicle body axis system (N) |
| $F_z$ | Vertical tyre loads (N) |
| $h$ | Height of vehicle CG above roll axis (m) |
| $h_G$ | Vehicle CG height (m) |
| $h_R$ | Roll centre height (front, rear) (m) |
| $I_{xx}, I_{yy}$ | Vehicle roll, yaw inertia (kg m$^2$) |
| **k** | Normalised slip vector |
| $K_\phi$ | Roll stiffness (Nm/rad) |
| $K_x$ | Tyre longitudinal stiffness |
| $L$ | Vehicle wheelbase (m) |
| $M$ | Vehicle mass (kg) |
| $p, r$ | Vehicle body roll, yaw angular velocities (SAE) (rad/s) |
| $P(|\mathbf{k}|)$ | Simplified Pacejka magic formula |

| $S_i$ | Longitudinal slip ratio, $i = 1$–$4$ (front left, front right, rear left, rear right) |
| $U, V$ | Vehicle body longitudinal, lateral velocities (SAE axes) (m/s) |
| $U_{xi}, U_{yi}$ | Wheel-oriented longitudinal, lateral velocities, $i = 1$–$4$ (front left, front right, rear left, rear right) (m/s) |
| $W$ | Vehicle weight (front, rear) (N) |
| $\alpha_i$ | Slip angle, $i = 1$–$4$ (front left, front right, rear left, rear right) (rad) |
| $\delta$ | Steered wheel angle (rad) |
| $\Delta_{\text{lat}}$ | Lateral load transferred (N) |
| $\phi$ | Vehicle body roll angle (rad) |
| $\omega_i$ | Wheel angular velocities, $i = 1$–$4$ (front left, front right, rear left, rear right) (rad/s) |

## *Qualifications*

| $^\star$ | Updated/propagated vector/matrix |
| $\hat{\phantom{x}}$ | Optimal estimate of vector |
| $k$ | Time step |
| f | Front |
| r | Rear |

## 1. Introduction

Advanced automotive control requires accurate dynamic models to predict vehicle response. In recent years, dynamic modelling has also become increasingly important in vehicle development, due to obvious time and cost saving advantages over traditional prototype testing [1]. Regardless of the application, a compromise between accuracy and complexity needs to be sought, as the ability to run in real time is a major goal in both control and simulation. Hence, the employed models need to be simple and efficient, rather than complex and heavy. The validity of these necessarily simplified models depends on many fixed or estimated parameters. Some of these, such as the centre of gravity (CG) height or the moment of inertia are difficult and expensive to measure. Others, as pointed out in [2] depend on external factors: tyre adherence boundaries for example are heavily influenced by asphalt conditions and precise lab measurements often don't correlate well with the real world. Furthermore, even if these values are physically accurately set, the simplified model can be made to perform better if they are adaptively tuned or completely identified.

Most models in literature are either derived from first principles or generated with multibody techniques. Structured identification, also referred to as grey-box parametrisation can then be employed to identify one or more parameters of this perfectly known mathematical or multibody structure. Models based on Neural Networks are also found [3], but these are black-box and have the disadvantage of not giving any engineering insight into the system.

Common identification tools include the Least Squares, Maximum Likelihood, and Recursive Prediction Error method [4]. And the well-known Kalman Filter [5] which prevails in a large portion of literature, principally in its nonlinear form, the extended Kalman filter (EKF). Several authors have successfully employed EKF to identify a limited number

of model parameters, which are concatenated to the state vector and estimated simultaneously with the true states [6,7]. This approach has later been extended to wholly concentrate on parameter estimation [8] and recent findings suggest that Kalman filter methods can be applied to identify all the parameters of any well-conditioned model structure [9]. The Unscented Kalman filter (UKF) has emerged in the last two decades as the main alternative to EKF. First developed for nonlinear state estimation by researchers at Oxford University [10], it is based on the idea that approximating a nonlinear statistical distribution is easier and more accurate than linearising a nonlinear function, as done by the EKF. It avoids the need to calculate Jacobians and is computationally less expensive and easier to implement. It is also easily extended to system identification and dual estimation problems [11,12], in a similar fashion to the EKF. An example of adaptive parameter estimation in vehicle dynamics has been developed in [13] and a recent study [14] includes tyre parameter estimation, though this is carried out using simulation only. Other recent UKF studies consider state estimation only. The particle filter (PF) belongs to the group of recursive Monte Carlo methods and is particularly suited to harsh nonlinearities and non-Gaussian applications [15]. It approximates the posterior probability density function (pdf) of the state vector in a similar way to the UKF, but uses a much larger set of samples, which are randomly selected from an initial uniform distribution. In certain applications the PF has proven to be a promising alternative to gradient-based Kalman Filters [16], since the sampling representation of the pdf is in general a better approximation of non-Gaussian distributions caused by nonlinear model functions. However, examples in literature usually rely on high computational power and low sampling rates, especially for real-time solutions [17].

The focus of this paper is on grey-box automotive parameter identification and three of the most common and promising identification methods are used for the purpose: the EKF, the UKF and the PF. Several past publications have considered the identification of *individual* parameters of a vehicle model, particularly for tyre and friction coefficient estimation [6,18]. Longitudinal, lateral and vertical dynamics have been considered separately, to estimate a different parameter from each model [19]. And in [20] a method is presented for simultaneous identification of a larger set of parameters, though this method relies on a-priori knowledge of Pacejka tyre coefficients. Here we extend the findings in [21], identifying all the independent parameters of a whole vehicle handling model simultaneously, including longitudinal, yaw and roll freedoms, and with independent combined-slip load dependent tyres. We consider data collected from a test vehicle carrying out medium to high magnitude manoeuvres including wheel-spin and terminal understeer, in order to build a model which is valid over the whole range of the tyres.

In the next sections each filter is first introduced and described in its identifying form. A simplified case is then presented – identification of the front and rear tyre stiffness of a linear single-track model, based on test vehicle data. This example serves as an immediate and straightforward comparison of the different techniques, illustrating how they are tuned for performance and efficiency. The major study of identification and validation of the more complex four degree of freedom full vehicle handling model then completes the paper.

## 2. Identifying EKF

The standard state estimating EKF operates on nonlinear system and sensor models $\mathbf{f}$ and $\mathbf{h}$, which relate the state vector $\mathbf{x}$, measured sensor set $\mathbf{y}$, known inputs $\mathbf{u}$ and model

parameters $\theta$ through

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \theta) + \omega, \tag{1}$$

$$\mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{u}, \theta) + \upsilon, \tag{2}$$

$\omega$ represents state propagation and modelling error, $\upsilon$ is the sensor error, and an optimal filter can be derived using estimates or expectations of the error covariance matrices:

$$\mathbf{Q} = E(\omega\omega^{\mathrm{T}}) \quad \mathbf{R} = E(\upsilon\upsilon^{\mathrm{T}}). \tag{3}$$

The EKF also requires model Jacobians to be evaluated at each time step, defined

$$\mathbf{F} = \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{u}, \theta)}{\partial \mathbf{x}} \quad \mathbf{H} = \frac{\partial \mathbf{h}(\mathbf{x}, \mathbf{u}, \theta)}{\partial \mathbf{x}}. \tag{4}$$

For further details of the full EKF see [6].

The identifying EKF (IEKF) takes advantage of the fact that $\mathbf{f}$ and $\mathbf{h}$ are general nonlinear functions of $\mathbf{x}$ and $\theta$, defining an extended state vector $\mathbf{z}$ with extended state derivatives set zero for the parameter states:

$$\dot{\mathbf{z}} = \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \mathbf{f}(\mathbf{z}, \mathbf{u}) \\ \mathbf{0} \end{bmatrix}. \tag{5}$$

It is computed using a sequence of equations which develop a time-varying estimate of state error covariance, $\mathbf{P}_k$ and Kalman gain $\mathbf{K}_k$; at each time step of the recorded time histories, compute

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_k \mathbf{H}_k^{\mathrm{T}} [\mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^{\mathrm{T}} + \mathbf{R}]^{-1}, \\ \mathbf{P}_k^* &= [\mathbf{I} - \mathbf{K}_k \mathbf{H}_k] \mathbf{P}_k, \\ \mathbf{P}_{k+1} &= \mathbf{P}_k^* + T[\mathbf{F}_k \mathbf{P}_k^* + \mathbf{P}_k^* \mathbf{F}_k^{\mathrm{T}} + \mathbf{Q}], \\ \hat{\mathbf{z}}_{k+1} &= \hat{\mathbf{z}}_k + T\mathbf{f}_k + \mathbf{K}_k(\mathbf{y}_k - \mathbf{h}_k). \end{aligned} \tag{6}$$

The final equation in set (6) combines Euler integration of the system using time step $T$ with state and parameter adaptation driven by the output error (known as the *innovation* sequence).

## 3. Unscented Kalman filter

The UKF identifies its own error statistics at each iteration, and hence avoids the need to use Jacobians. According to Julier and Uhlmann [10] and Wu et al. [22], a sample of $(2n + 1)$ so called sigma points $\chi$ are selected around the $n$th order state vector, at each instant $k$ in time:

$$\begin{aligned} \chi_{0k} &= \hat{\mathbf{x}}_k, \\ \chi_{ik} &= \hat{\mathbf{x}}_k + \{\sqrt{(n + \kappa)\mathbf{P}_k}\}_i, \\ \chi_{(i+n)k} &= \hat{\mathbf{x}}_k - \{\sqrt{(n + \kappa)\mathbf{P}_k}\}_i. \end{aligned} \tag{7}$$

where $\{\sqrt{(n + \kappa)\mathbf{P}_k}\}_i$ is the $i$th column of the matrix square root of $(n + \kappa)\mathbf{P}_k$ (obtained here using Cholesky decomposition) and $\mathbf{P}_k$ is the current estimate of state error covariance. These sigma points are propagated by the model, here as for IEKF, using Euler

approximation:

$$\chi_{i(k+1)} = T\mathbf{f}(\chi_{ik}, \mathbf{u}_k, \boldsymbol{\theta}) \tag{8}$$

and intermediate estimates for the propagated state and covariance matrix are computed by weighted averages:

$$\hat{\mathbf{x}}_{k+1}^* = \sum_{i=0-2n} W_i \chi_{i(k+1)},$$

$$\mathbf{P}_{k+1}^* = \sum_{i=0-2n} W_i \{\chi_{i(k+1)} - \hat{\mathbf{x}}_{(k+1)}\}\{\chi_{i(k+1)} - \hat{\mathbf{x}}_{(k+1)}\}^T + T\mathbf{Q}. \tag{9}$$

With $W_0 = \kappa/(n+\kappa)$ and for all other $i$, $W_i = 1/2(n+\kappa)$

Similarly, average outputs are obtained according to the output model:

$$\Upsilon_{i(k+1)} = \mathbf{h}(\chi_{ik}, \mathbf{u}_k, \boldsymbol{\theta}),$$

$$\hat{\mathbf{y}}_{k+1} = \sum_{i=0-2n} W_i \Upsilon_{i(k+1)}. \tag{10}$$

The UKF then propagates output error covariance according to the transformed sigma points:

$$\mathbf{P}_{yy} = \sum_{i=0-2n} W_i \{\Upsilon_{i(k+1)} - \hat{\mathbf{y}}_{(k+1)}\}\{\Upsilon_{i(k+1)} - \hat{\mathbf{y}}_{(k+1)}\}^T + \mathbf{R} \tag{11}$$

and uses this together with a cross correlation estimate

$$\mathbf{P}_{xy} = \sum_{i=0-2n} W_i \{\chi_{i(k+1)} - \hat{\mathbf{x}}_{(k+1)}\}\{\Upsilon_{i(k+1)} - \hat{\mathbf{y}}_{(k+1)}\}^T \tag{12}$$

to find the Kalman gain by

$$\mathbf{K}_{k+1} = \mathbf{P}_{xy}\mathbf{P}_{yy}^{-1}.$$

State and covariance estimates are then updated using the innovation sequence, in a similar way to IEKF:

$$\mathbf{P}_{k+1} = \mathbf{P}_{k+1}^* - \mathbf{K}_{k+1}\mathbf{P}_{yy}\mathbf{K}_{k+1}^T,$$

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_{k+1}^* + \mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1}). \tag{13}$$

The above UKF derivation is common to all implementations of state and parameter estimation, so the simple substitution of $\mathbf{x}$ in the above, with $\mathbf{z}$ from Equation (5) provides the identifying filter.

## 4. Particle filter

The PF is a recursive Bayesian estimator based on Monte-Carlo simulations. According to Gustafsson et al. [15] and Gordon et al. [17], the posterior density of the state vector

is approximated through a set of $n_p$ so called particles, initially generated from a uniform distribution on a finite interval $[\mathbf{a,b}]$:

$$\chi_{i0} \in u[\mathbf{a, b}] \quad i = 1 \ldots n_p. \tag{14}$$

So each particle $\chi_{i0}$ is an $n$th order state vector at time $k = 0$ with each of its elements selected on a uniform distribution with range $[a_j, b_j]$ selected appropriate to the minimum and maximum expected value of that state. Each particle is propagated by the process and output models at every instant $k$, according to:

$$\chi_{i(k+1)} = \chi_{ik} + Tf(\chi_{ik}, \mathbf{u}_k, \boldsymbol{\theta}), \tag{15}$$

$$\Upsilon_{ik} = \mathbf{h}(\chi_{ik}, \mathbf{u}_k, \boldsymbol{\theta}), \tag{16}$$

where Euler integration is used in Equation (15), as for IEKF/UKF.

A weight $W_i$ is then assigned to each of the particles, based on the error between the propagated output (16) and the true output of the system. These errors are assumed to have a known distribution and zero mean, hence:

$$W_i = \prod_{j=1-n_y} \varphi_j(\mathbf{y}_k(j) - \Upsilon_{ik}(j)|0, \sigma_j^{2}), \tag{17}$$

where $n_y$ is the number of outputs and $\varphi_j$ is the assumed pdf of the $j$th output innovation sequence, with zero mean and variance $\sigma_j$. Here we assume a normal distribution of innovations and hence use

$$W_i = \prod_{j=1-n_y} \frac{1}{\sqrt{2\sigma_j^2 \pi}} \mathrm{e}^{-(\mathbf{y}_k(j)-\Upsilon_{ik}(j))^2/\sqrt{2\sigma_j^2\pi}}. \tag{18}$$

The weights are normalised so that their sum equals unity:

$$W_i = W_i \bigg/ \sum_{i=1-np} W_i. \tag{19}$$

State and output estimates are then computed by a weighted sum of the propagated particles:

$$\hat{\mathbf{x}}_k = \sum_{i=1-n_p} W_i \chi_{ik}, \tag{20}$$

$$\hat{\mathbf{y}}_k = \sum_{i=1-n_p} W_i \Upsilon_{ik}. \tag{21}$$

The filter then resamples $n_p$ particles from the original set using $W_i$ as the probability of res-election. In this way, where a new full set of $n_p$ particles is generated, these include repeated instances of successful particles and exclude low-weighted particles. The new set of particles for the next iteration is then finalised by randomising the state combinations from the pool of these resampled particles. Equations (15)–(21) are then recursively repeated through the data points.

The PF is typically used for state estimation, for which all $n_\mathrm{p}$ particles are necessarily propagated, weighted and resampled at least once at each time step $k$. For identification this has the disadvantage that convergence occurs over a very short section of data (though $n_\mathrm{p}$ is typically high – around $10^5$ – so the process is still computationally intensive). To enable parameter optimisation over a more representative, longer section of identification data (or for continuous identification in real-time) it is sensible to propagate just one particle per time step ($i = k$). Since the PF operates without a state covariance estimate there is then a further advantage that the true states, $\mathbf{x}$ need not be included in the identification set. In converting the PF for identification we therefore select $\boldsymbol{\chi} \equiv \boldsymbol{\theta}$ so Equations (15)–(16) become

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_k + T\mathbf{f}(\hat{\mathbf{x}}_k, \mathbf{u}_k, \boldsymbol{\chi}_{i+1}), \tag{22}$$

$$\Upsilon_{i+1} = \mathbf{h}(\hat{\mathbf{x}}_{k+1}, \mathbf{u}_{k+1}, \boldsymbol{\chi}_{i+1}). \tag{23}$$

Identification using batch data then progresses in an obvious way, using multiple passes through the available data to exercise all $n_\mathrm{p}$ particles. Weighting and resampling operations occur after each $n_\mathrm{p}$ time steps as in the original method.

## 5. Implementation in a simplified case – identification of a linear model

Test data were obtained from a '08 MY Jaguar XF equipped with an OXTS 3200 inertial navigation (IN) device, driven on a dry proving ground. A range of high and low magnitude, separated and combined slip manoeuvres were conducted, exciting both dynamic and steady-state vehicle responses. The controller area network (CAN) data was inter-sampled at a constant 100 Hz to match the IN (setting $T = 0.01$) and all data was then digitally filtered at 10 Hz to remove higher frequency noise; this is necessary to prevent instability in the filters, as they are propagated by Euler integration. Although here we perform a batch data process, the identifying filters are designed to work equally well in real-time, and real-time parameter tuning could easily be achieved using a combination of dynamic low pass filters and/or higher sampling interval $T$.

To illustrate execution, convergence and computational effort of the filters, consider first the identification of a linear bicycle handling model with states $\mathbf{x} = [V, r]^\mathrm{T}$

$$\dot{\mathbf{x}} = \begin{bmatrix} -(C_{\alpha f} + C_{\alpha r})/(MU) & (bC_{\alpha r} - aC_{\alpha f})/(MU) - U \\ (bC_{\alpha r} - aC_{\alpha f})/(I_{zz}U) & -(a^2 C_{\alpha f} + b^2 C_{\alpha r})/(I_{zz}U) \end{bmatrix} \mathbf{x} + \begin{bmatrix} C_{\alpha f}/M \\ aC_{\alpha f}/I_{zz} \end{bmatrix} u \tag{24}$$

60 s of low magnitude random steer data from a constant speed test is applied to each filter using the recorded steer as input, and yaw rate as output

$$u = \delta, \quad y = r$$

to identify front and rear axle cornering stiffnesses

$$\boldsymbol{\theta}_{\mathrm{id}} = [C_{\alpha f}, C_{\alpha r}]$$

with other model parameters fixed through knowledge/approximation of the vehicle and test speed, as

$$\boldsymbol{\theta}_{\mathrm{fix}} = [a = 1.38, b = 1.53, M = 1855, I_{zz} = 2000, U = 12.9].$$

Identification is achieved iteratively by 'rinsing' the data repeatedly through each filter, starting with an initial, nominal parameter set. Both Kalman filters require covariance estimates and here $\mathbf{Q}$ is fixed throughout as $\mathbf{Q} = \rho \mathbf{I}$. $\rho$ is the only tuning parameter in both Kalman filters; it weighs the expectation of accuracy of the model $\mathbf{f}$, and particularly the assumption $\dot{\boldsymbol{\theta}} = \mathbf{0}$, so it controls the variation rate of the parameters (see [9] for more detail). For any given estimate $\hat{\boldsymbol{\theta}}$ of the identified parameters, $\mathbf{R}$ can be obtained numerically from the covariance of $\upsilon$ using Equation (2); here, $\mathbf{R}$ is recomputed after each iteration. Finally, $\mathbf{P_0} = \mathbf{Q}$, and for the UKF $\kappa = 1$. Convergence and accuracy of the PF is affected by the choice of $n_p$.

Figure 1 shows the effect on accuracy and computational effort of varying these core tuning parameters $\rho$ and $n_p$ (upper plots). The lower plots then show the convergence
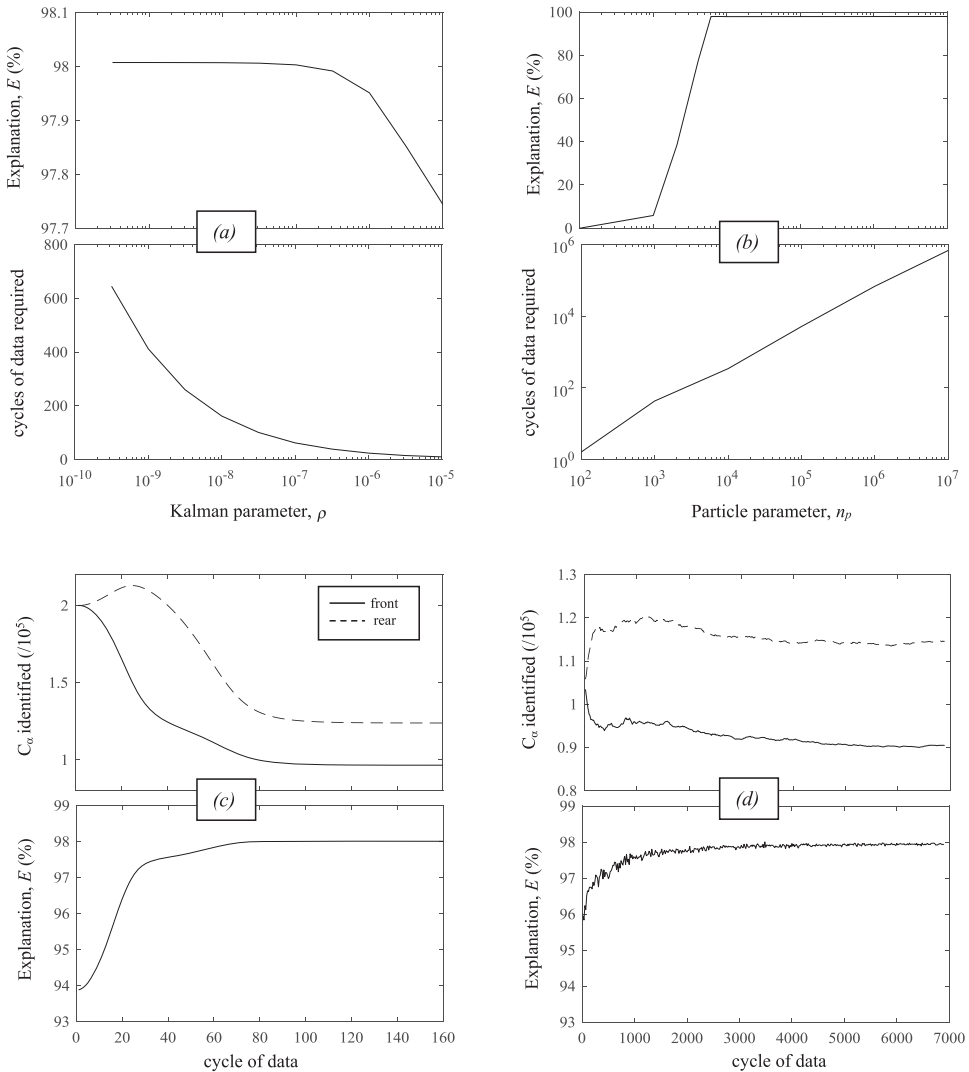


**Figure 1.** Tuning and comparing the filters on a simple identification case (left = Kalman, right = particle).

of the identified parameters and their corresponding accuracy for a single optimisation using suitable choices, $\rho = 10^{-8}$ and $n_p = 10^6$. Accuracy here is quantified by *percentage explanation* of any model time history $\hat{y}$ estimating measurement $y$:

$$E = \left(1 - \frac{\sum_{k=1}^{N}(y_k - \hat{y}_k)^2}{\sum_{k=1}^{N} y_k^2}\right) \times 100. \tag{25}$$

Interestingly, both Kalman Filter methods yield identical results (and we will see indistinguishable difference in the more complex full vehicle case later). Although computed using different techniques, both state covariance and parameter estimates match within a few seconds of filtered data. As the method of computation varies however, so does the processing time; 100 cycles of data takes the UKF 23 s whereas the EKF takes just 9 s (on the mid-range PC used here). The Kalman filters also converge smoothly and definitively at all settings of $\rho$. $E$ does not rise for $\rho < 10^{-8}$ and even at extremely low $\rho$, these filters yield the best possible accuracy from this data, $E = 98.0\%$ in under 3 min.

The PF is effective at optimising $C_\alpha$, achieving $E = 97.9\%$ in all cases where $n_p$ is sufficiently high. However, convergence of the parameters is less definitive, Figure 1(d), and the computational cost becomes high as performance reaches acceptable levels. The final result for $n_p = 10^5$ takes 10 min to converge, and the illustrated $n_p = 10^6$ case takes 90 minutes. For such a small unknown parameter set these are disappointing results; although the PF is effective, it is not competitive with either Kalman filter at delivering well converged, timely results.

## 6. Full vehicle model

For full vehicle and tyre identification a suitably complex model structure is the well-known three degree of freedom handling model, simulating yaw, roll, and sideslip using a load dependent, combined-slip Pacejka tyre model. A fourth, longitudinal degree of freedom is required to exercise both longitudinal and lateral tyre slip regimes. This model structure was previously employed in [23]; the principal equations of motion are

$$\text{longitudinal}: \quad M\dot{U} = \sum_{i=1-4} F_{xi} + MrV + Mhrp, \tag{26}$$

$$\text{lateral}: \quad M\dot{V} + Mh\dot{p} = \sum_{i=1-4} F_{yi} - MUr, \tag{27}$$

$$\text{yaw}: \quad I_{zz}\dot{r} = a\sum_{i=1,2} F_{yi} - b\sum_{i=3,4} F_{yi} + c\sum_{i=1,3} F_{xi} - c\sum_{i=2,4} F_{xi}, \tag{28}$$

$$\text{roll}: \quad Mh\dot{V} + I_{xx}\dot{p} = -MhUr + Mgh\phi - h_0\sum_{i=1-4} F_{yi} + c\sum_{i=1,3} F_z - c\sum_{i=2,4} F_z, \tag{29}$$

$$\text{roll kinematics}: \quad \dot{\phi} = p. \tag{30}$$

Standard SAE axes are used, fixed relative to the vehicle wheelbase, and the wheels are labelled (1–4) in ascending order as (front-left, front-right, rear-left, rear-right). Equal half-track $c$ is assumed, with axles separated distances $a$ and $b$ from the front and rear axles,

respectively, and vertical suspension geometry is based on fixed roll centres $h_{R(f,r)}$ and CG height $h_G$ such that

$$h_0 = h_{Rf} + \frac{W_r}{Mg}(h_{Rr} - h_{Rf}) \quad h = h_G - h_0. \tag{31}$$

The forces controlling the vehicle body motion $(F_{xi}, F_{yi})$ allow for large steer angles

$$
\begin{aligned}
F_{x1,2} &= F^*_{t\,x1,2} \cos\delta - F^*_{t\,y1,2} \sin\delta \quad F_{x3,4} = F^*_{t\,x3,4} \\
F_{y1,2} &= F^*_{t\,y1,2} \cos\delta - F^*_{t\,x1,2} \sin\delta \quad F_{y3,4} = F^*_{t\,y3,4}
\end{aligned}
\tag{32}
$$

based on lagged tyre forces, where each of the 8 elements are lagged to simulate relaxation within the tyre

$$\dot{F}^*_{tx/y,i} = \tau^{-1}(F_{tx/y,i} - F^*_{tx/y,i}). \tag{33}$$

The tyre forces $(F_{t\,xi}, F_{t\,yi})$ are modelled according to a slightly simplified Pacejka magic formula

$$P(|\mathbf{k}|) = D \sin\left(C\tan^{-1}\left(\frac{|\mathbf{k}|}{C} - E\left(\frac{|\mathbf{k}|}{C} - \tan^{-1}\frac{|\mathbf{k}|}{C}\right)\right)\right) \tag{34}$$

using normalised slip and isotropic similarity scaling [24,25]. The normalised slip vector is

$$\mathbf{k} = \begin{pmatrix} k_x \\ k_y \end{pmatrix} = \frac{C_\alpha}{\mu F_z}\begin{pmatrix} K_x S_i \\ \tan\alpha_i \end{pmatrix}, \tag{35}$$

where $S$ is the longitudinal slip ratio, and $\alpha$ is the slip angle at each tyre contact patch,

$$S_i = \frac{r_r\omega_i - U_{xi}}{U_{xi}} \quad \tan\alpha_i = \frac{-U_{yi}}{U_{xi}} \tag{36}$$

based on wheel-oriented velocities

$$
\begin{aligned}
U_{xi} &= U^*_{xi}\cos\delta_i + U^*_{yi}\sin\delta_i \\
U_{yi} &= U^*_{yi}\cos\delta_i - U^*_{xi}\sin\delta_i
\end{aligned}
\tag{37}
$$

and

$$
\begin{aligned}
U^*_{x1,3} &= U + cr \quad U^*_{x2,4} = U - cr, \\
U^*_{y1,2} &= V + ar \quad U^*_{y3,4} = V - br.
\end{aligned}
\tag{38}
$$

The resulting tyre force vector is then

$$\begin{pmatrix} F_{tx} \\ F_{ty} \end{pmatrix} = P(|\mathbf{k}|)\frac{\mu F_z}{|\mathbf{k}|}\begin{pmatrix} k_x \\ k_y \end{pmatrix}. \tag{39}$$

Vertical tyre loads $F_z$ are calculated from static weight distribution, modified to accommodate lateral load transfer using separate front/rear distributions according to:

$$\Delta_{lat}F_{z(f/r)} = \frac{\sum F_{y(f/r)}h_{R(f/r)} + K_{\phi(f/r)}\phi + B_{\phi(f/r)}p}{2c} \tag{40}$$

and longitudinal load transfer, according to a rigid, zero pitch approximation,

$$\Delta_{\text{long}}F_z = \frac{\sum\limits_{i=1-4} F_{xi}h_G}{(a+b)}.$$

(41)

## 7. Full vehicle identification

For full vehicle and tyre identification, input data is comprised of steer angle and the four wheel speeds; the outputs are CG longitudinal and lateral accelerations and body roll rate, modelled as

$$\mathbf{u} = [\delta, \omega_1, \omega_2, \omega_3, \omega_4]^{\text{T}},$$

$$\mathbf{y} = \left[ \frac{1}{M} \sum_{i=1-4} F_{xi}, \frac{1}{M} \sum_{i=1-4} F_{xi}, p \right]^{\text{T}}.$$

(42)

All the above measurements with the exception of body roll rate were taken from the vehicle CAN; although roll was measured using the IN, this sensor may already be available in other CAN sets, or would be cheap to add. IN measured vehicle speed, roll angle, yaw rate and lateral velocity are also used for validation of the identified model. One 100 second test including dynamic and steady-state excitations of steering, brake and acceleration inputs was used as the identification set. This test employed random steer inputs carried out on a circuit of the proving ground track, including some extreme high slip inputs in combination and separated from medium to high acceleration and braking applications. A variety of other tests using combined and separated slip were recorded for validation.

The only difficult task is the decision of which parameters to adapt in the identification, and which to fix. The filters can identify any number of parameters, but if they are not independent in their influence on the recorded outputs, the parameters will diverge. After various trials the following identification set was established as the minimum non-divergent set:

$$\boldsymbol{\theta}_{\text{id}} = [h_G, I_{zz}, K_\phi, B_\phi, K_{xf}, K_{xr}, C_{\alpha f}, C_{\alpha r}, C_f, D_f, E_f, C_r, D_r, E_r].$$

This allows full identification of separate front and rear individual tyre-suspension models but with roll stiffness and damping applied at a constant ratio, known from manufacturer supplied data. The remaining fixed parameters constrain the weight balance, roll inertia and essential geometry ($L$ = wheelbase), and are set

$$\boldsymbol{\theta}_{\text{fix}} = [W_f = 9567, W_r = 8635, I_{xx} = 696, h_{\text{Rf}} = 0.08, h_{\text{Rr}} = 0.125, c = 0.79, L = 2.91].$$

Friction is identified via the tyre $D$ parameters, so we set $\mu = 1$. To illustrate the effect of identifying too many parameters, we will also consider the case of adding roll centre heights $h_{\text{Rf}}$ and $h_{\text{Rr}}$ to the identified set; these are referred to below as the h+ case.

Attempts to identify this full vehicle case using the PF were unsuccessful. Despite lengthy optimisations with high $n_p$ the larger set of parameters proved impossible to identify. Results with both Kalman filters are encouraging however; Figure 2(a–c) illustrate the convergence behaviour of IEKF, UKF and $h+$ (IEKF) for $\rho = 10^{-8}$ over 200 cycles (iterations)
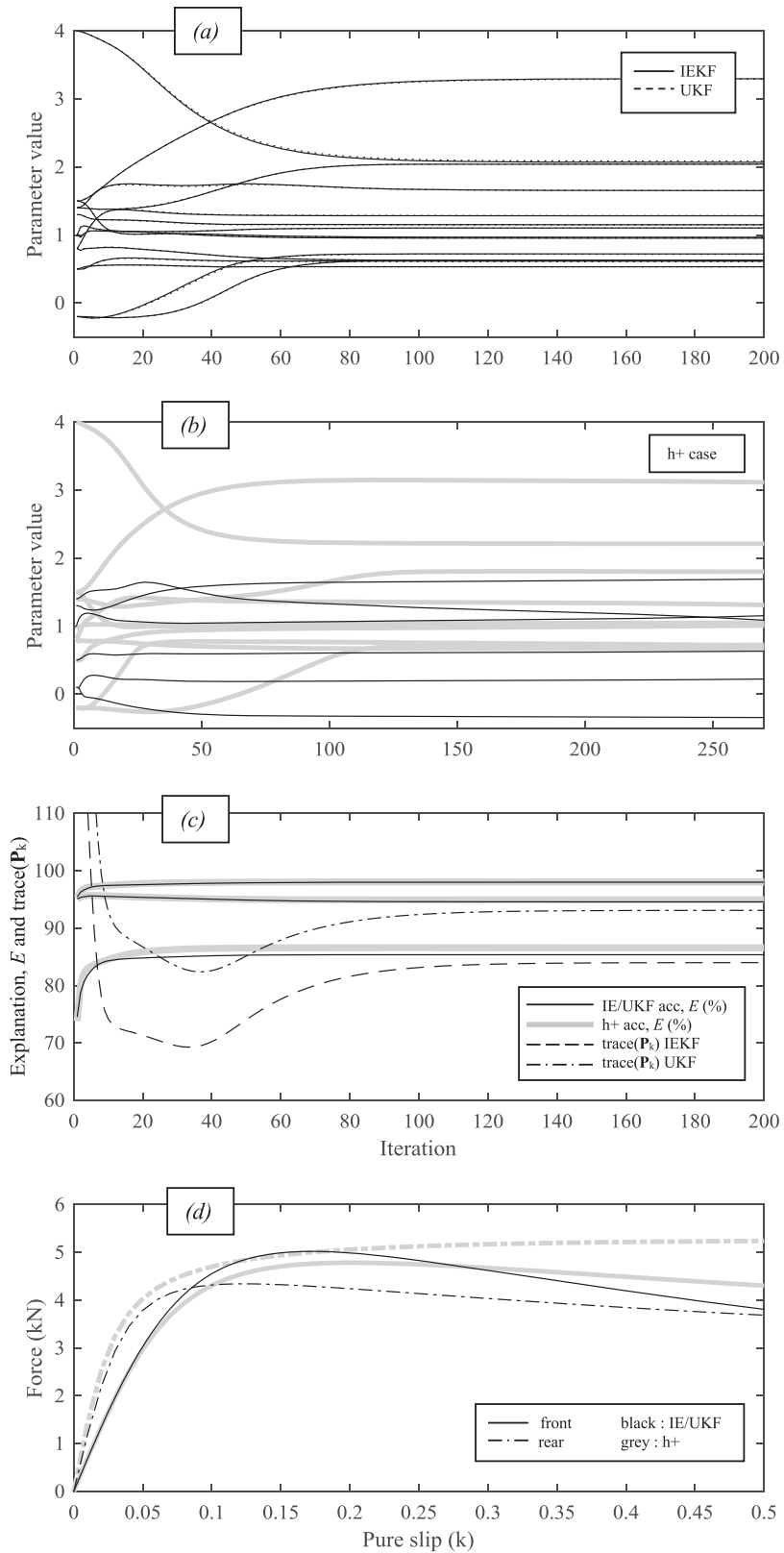
**Figure 2.** Identification convergence and resulting tyre.

of the identification data. Note how all 14 parameters converge completely in Figure 2(a), as do the values for performance and trace($\mathbf{P_k}$), in plot (c). The standard IEKF and UKF results are almost identical; it is just possible to make out tiny differences in the parameters as they converge in Figure 2(a). Final parameter values are given in Table 1 and the tyre/suspension model is illustrated in terms of pure slip, in Figure 2(d); note that the absence of suspension derivatives in the simplified model means that the tyre model here is actually mapping the combined tyre/suspension characteristic. All values are in the expected range; the front tyre has a lower stiffness and quicker saturation than the rear, which is consistent with expected front steer compliance seen in an earlier study on the same vehicle [26]. Although it achieves slightly higher output explanations – Figure 2(c), the $h+$ case results in a less well-conditioned model, with unphysical negative $h_{Rf}$ and high $K_\phi$ and $h_G$. Note that the result in Table 1 is a snapshot at the 200th iteration; Figure 2(b) highlights the divergent parameters.

As these results optimise parameters for the given simplified model structure, we are not seeking parameters which exactly match manufacturer's data for the test vehicle. However it is a test of model conditioning that the identified parameters appear to be feasible, so manufacturer's vehicle data is also given in Table 1 where direct comparison is meaningful. The only significant difference is in yaw inertia; interestingly the identified model employs a lower front/right difference in tyre forces coupled with lower than expected yaw inertia. A further validation check of steady-state metrics also shows a good match; the model roll rate is 5.2 deg/g compared with manufacturers data of 4.5 deg/g and linear range understeer gradient is 4.9 deg/g which compares favourably with separate steady-state tests carried out on this vehicle found in the range 5–7 deg/g.

Figure 3 shows a section of data from the identification set, illustrating the final performance explanations achieved on each of the three output variables. The range of amplitudes in the inputs is also clear here; seeking to characterise the full range of the tyre, over the 100 s longitudinal accelerations range from $-7$ to $+4$ m/s$^2$, occasionally inducing wheelspin, and lateral accelerations of $\pm 8$ m/s$^2$ cover the full range through saturation. Note also the inclusion of lower amplitudes and the combination of dynamic and steady-state inputs.

**Table 1.** Identified parameters and selected vehicle parameters.

| Parameter | IEKF | UKF | $h^+$ | Vehicle |
|---|---|---|---|---|
| $h_G$ (m) | 0.533 | 0.532 | 0.633 | 0.530 |
| $h_{Rf}$ (m) | (0.08) | (0.08) | $-0.344$ | 0.08 |
| $h_{Rr}$ (m) | (0.125) | (0.125) | 0.223 | 0.125 |
| $I_{zz}$ ($\times 1000$ kgm$^2$) | 2.069 | 2.086 | 2.214 | 3.596 |
| $K_\phi$ ($\times 100$ kNm/rad) | 1.152 | 1.147 | 1.692 | 1.325 |
| $B_\phi$ ($\times 10$ kNms/rad) | 0.615 | 0.608 | 1.010 | – |
| $K_{xf}$ | 3.299 | 3.291 | 3.116 | – |
| $K_{xr}$ | 0.966 | 0.966 | 1.036 | – |
| $C_{\alpha f}$ ($\times 100$ kN/rad) | 0.625 | 0.627 | 0.665 | – |
| $C_{\alpha r}$ ($\times 100$ kN/rad) | 1.283 | 1.286 | 1.311 | – |
| $C_f$ | 2.042 | 2.043 | 1.801 | – |
| $D_f$ | 1.103 | 1.102 | 1.051 | – |
| $E_f$ | 0.630 | 0.632 | 0.724 | – |
| $C_r$ | 1.653 | 1.652 | 1.085 | – |
| $D_r$ | 0.954 | 0.955 | 1.155 | – |
| $E_r$ | 0.720 | 0.714 | 0.711 | – |
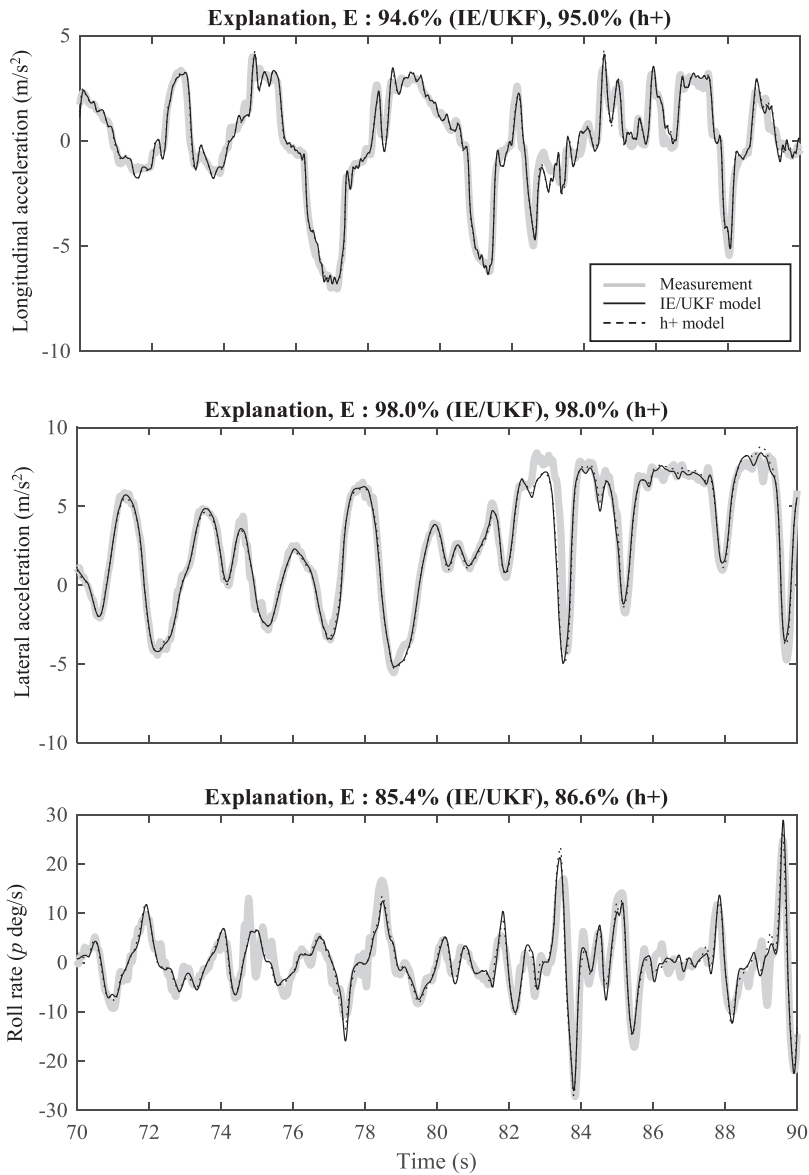
**Figure 3.** Model fit to measured, fitted identification data.

Evidence of success in the identification is then given in Figure 4 which shows the response of four additional model outputs not included in the identification filter, over the same section of data. We would expect the excellent longitudinal velocity match, since wheel speeds are inputs to the model, but well matched roll angle and in particular lateral velocity histories show that the parameters actually capture the tyre/suspension and roll modes very well.

The standard, physically appropriate model is also validated using independent test data in Figures 56–7. Note that in earlier testing [21], poorly calibrated instrumentation meant
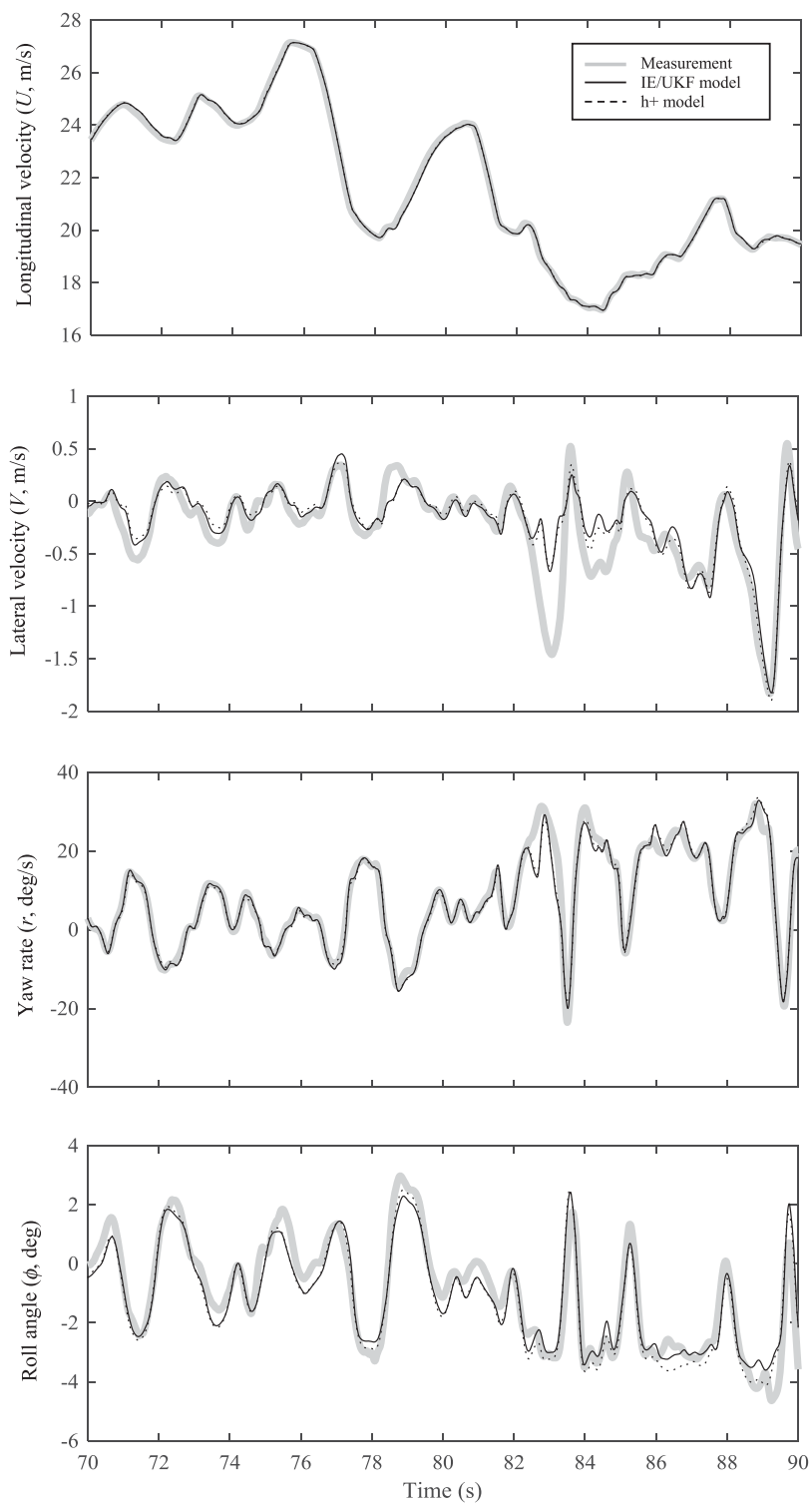
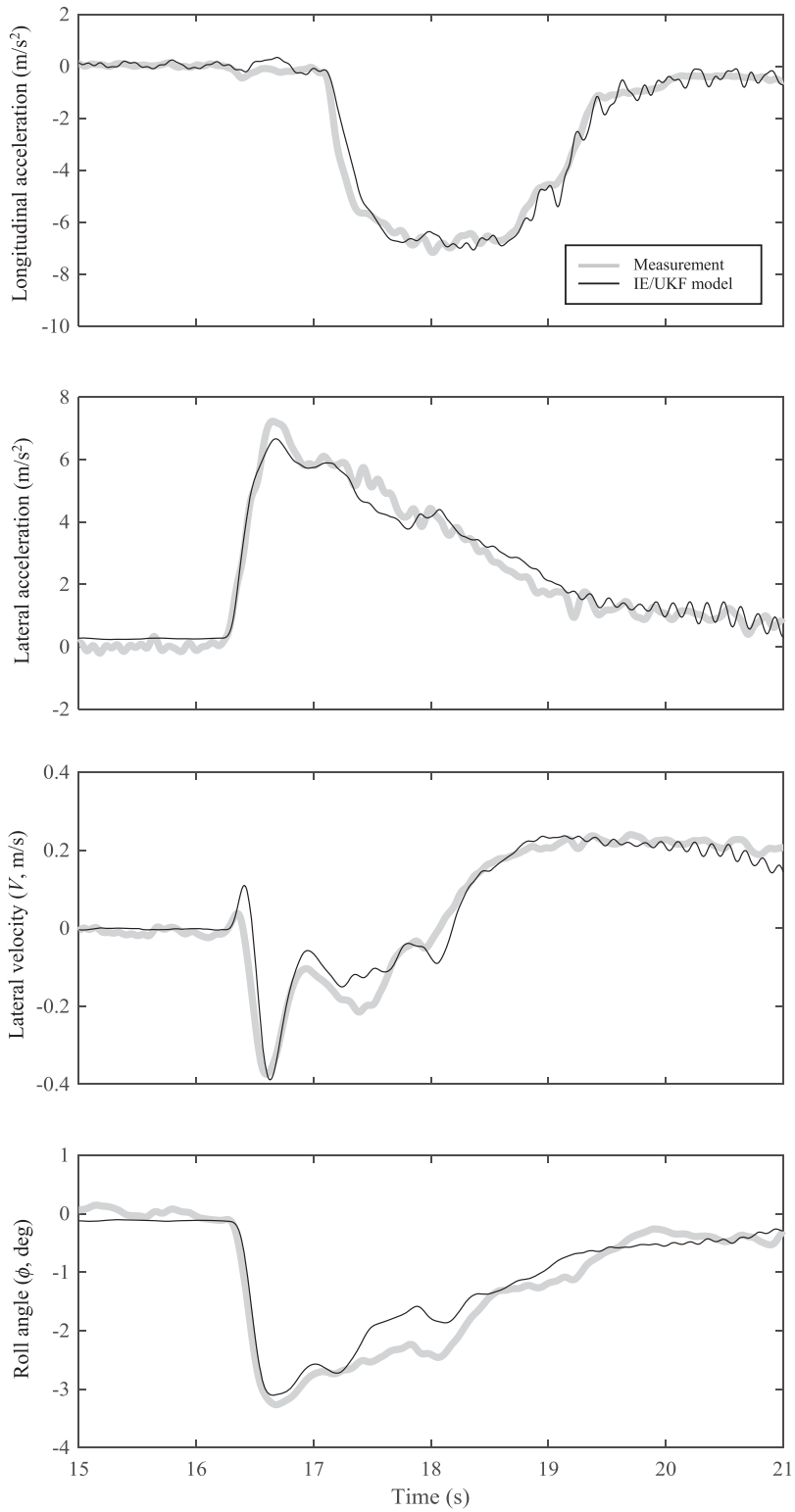**Figure 4.** Model fit to unfitted measurements in the identification data.

**Figure 5.** Identified model fit to combined brake/steer validation test.
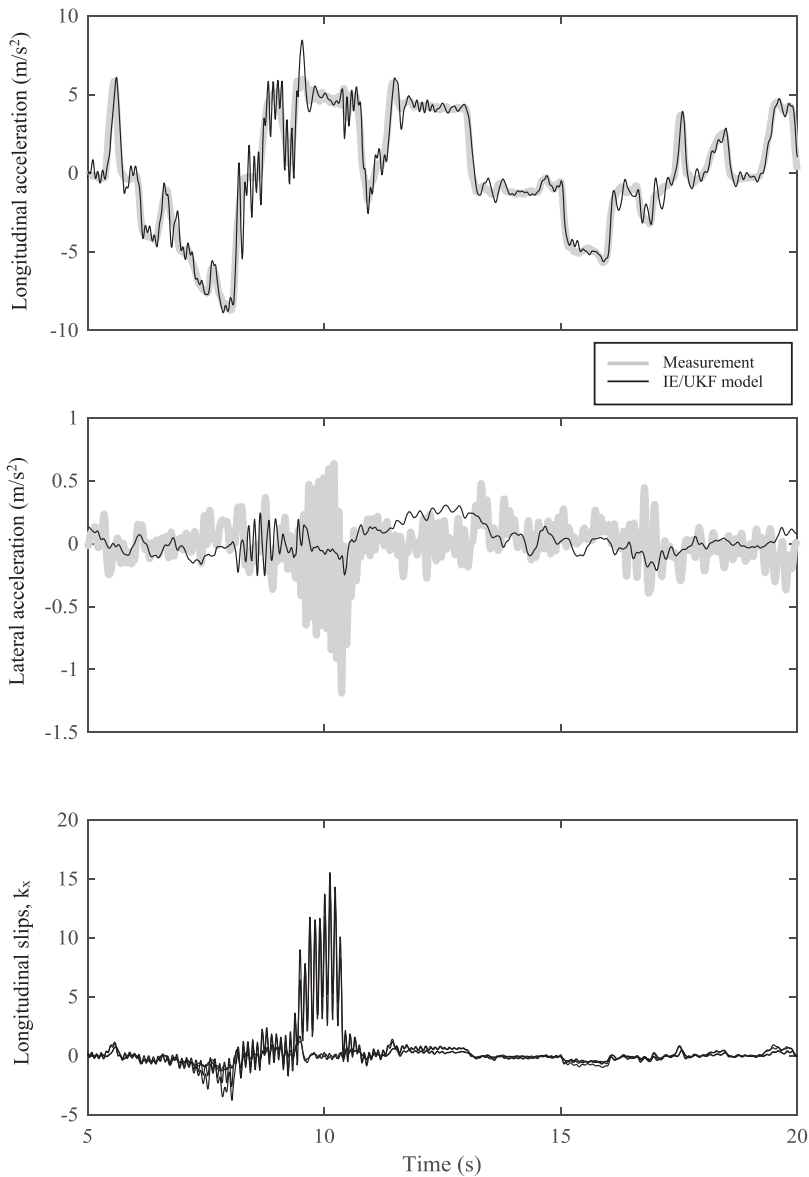
**Figure 6.** Model fit to straight-line brake/accel validation test.

that lateral velocity drift was present. These validation results were therefore generated after recalibration of the INS, 6 months after the original identification tests, on the same track and in similar weather conditions. These results thus also demonstrate repeatability and robustness of the identified model, over time.

Figure 5 shows the result of a step steer manoeuvre which is held while heavy braking is applied approx one second later. Again in addition to accurate results in the identified acceleration outputs, the model is also capable of matching roll angle and lateral velocity very well.
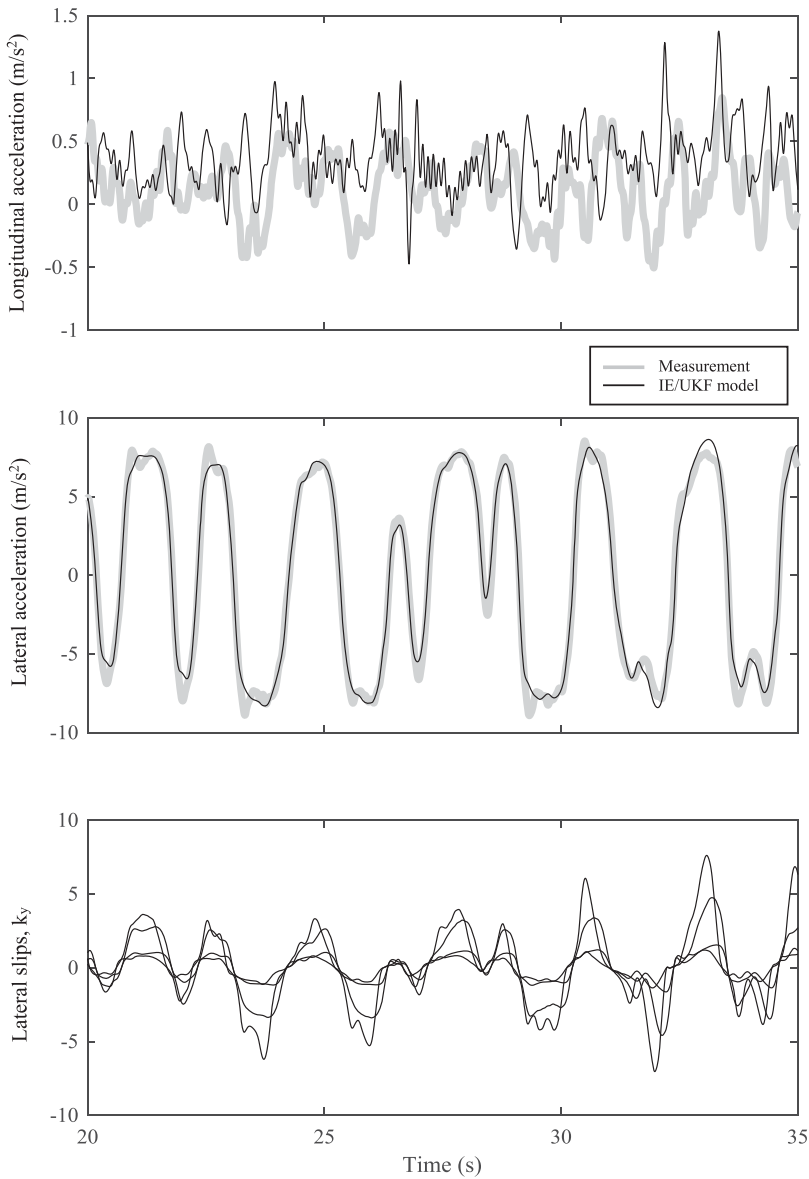
**Figure 7.** Model fit to constant speed, random steer validation test.

Finally, Figures 6 and 7 show further positive results, validating the combined-slip tyre/suspension model in single-slip tests. Figure 6 shows accurate results for a straight line braking and acceleration test under zero steer. The modelled longitudinal acceleration is indistinguishable from the measured data throughout, despite the high $k_x$ excursion caused by wheel-spin around 10 s; lateral accelerations track correctly at zero. Finally, Figure 7 sees high accuracy in the single slip lateral case of a constant speed, random steer manoeuvre. Both of these tests explore the tyre in low to medium magnitude and also in over-slip saturation conditions.

Both IEKF and UKF identified the same successful model here, so the decision over which filter is 'better' depends only on computational efficiency and complexity of code. UKF has the advantage that Jacobians are not required, so it is simpler to code, though Matlab's Symbolic toolbox means this overhead in complexity is slight. For the simple model in Section 5 the IEKF was clearly faster, but here processing times are only slightly different and the order is reversed; each IEKF iteration takes 266 s whereas each UKF iteration takes 212 s.

The longer processing times are not simply due to the more complex model. As the number of parameters increases, the UKF has more sigma points to process at each time step. Conversely the IEKF has become slower due to the computation of Jacobians **F** and **H**; these become highly complex here due to the Pacejka combined slip tyre model. The 'best' Kalman filter in applications on other model structures and for on-line use will therefore depend on the relative complexity of the Jacobians. Provided the number of parameters to be identified is not excessive, the UKF may be quicker to run and is certainly simpler to implement.

In the specific context of on-line application, where the inputs would certainly be less aggressive than considered here and full parameter identification is no longer the objective, a subset of parameters could easily be adapted. For example, even without modification of the sampling time or model conplexity, the UKF will run faster than real-time on the mid-range desktop PC used here; if the number of parameters is reduced from 14 to 8, 100 s of data is processed in 84 s.

## 8. Conclusions

Three alternative filtering methods have been adapted for system identification and applied to the identification of vehicle handling models. Although effective at identifiying near-optimal parameters for a simplified model, the PF is very slow to converge and could not be used to carry out full vehicle and tyre identification.

Both UKF and EKF were found to be effective at identifying both simple and complex vehicle models. Although they use different methods for parameter error covariance estimation, both techniques have identical convergence characteristics and yield near-identical models. In applications such as this, where model Jacobians are complex and a relatively small number of model parameters is identified, the UKF is simpler to implement and slightly faster to run. Both Kalman filters can identify a simple linear handling model in under 5 min, while the featured four-dof full vehicle and tyre model takes around 10–15 h to converge fully, depending on the filter used.

A full set of the independent parameters of the nonlinear handling model were identified, including combined-slip tyre/suspension characteristics identified over their full range up to and beyond saturation. This was done using only commonly available CG acceleration and roll rate measurements. The selection of identified parameters and their resulting conditioning has also been considered. The choice of which parameters to identify and which to fix can be critical, but the convergence behaviour of the identified parameters in the running filter make it easy to determine unphysical and/or under-determined combinations.

The resulting full vehicle model has been tested on identification data from a test vehicle and also on independent validation test vehicle data. It was found to be very accurate in

matching the measurements used for identification and also in explaining variables that are not readily measured, such as lateral velocity and roll angle. High quality results were seen over a range of validation tests, considering combined and separated longitudinal and lateral slip, at both high and low magnitudes.

## Disclosure statement

No potential conflict of interest was reported by the authors.

## Funding

## References

[1] Guan X, Ba T, Zhang J. Vehicle handling dynamics modelling by a data-driven identification method. Veh Syst Dyn. 2015;53(11):1580–1598.
[2] Lot R, Dal Bianco N. Lap time optimisation of a racing go-kart. Veh Syst Dyn. 2016;54(2):210–230.
[3] Savaresi SM, Sergio B., Mauro M. Identification of semi-physical and black-box non-linear models: the case of MR-dampers for vehicles control. Automatica. 2005;41(1):113–127.
[4] Au FTK, Jiang RJ, Cheung YK. Parameter identification of vehicles moving on continuous bridges. J Sound Vib. 2004;269(1):91–111.
[5] Kalman R. A new approach to linear filtering and prediction problems. J Basic Eng. 1960;82(1):35–45.
[6] Best MC, Gordon TJ, Dixon PJ. An extended adaptive Kalman filter for real-time state estimation of vehicle handling dynamics. Veh Syst Dyn. 2000;34(1):57–75.
[7] Best MC, Newton AP, Tuplin S. The identifying extended Kalman filter: parametric system identification of a vehicle handling model. Proc Inst Mech Eng K, J Multi-Body Dyn. 2006;221:87–98.
[8] Best MC. Parametric identification of vehicle handling using an extended Kalman filter. Int J Veh Auton Syst. 2007;5(3/4):256–273.
[9] Best MC, Bogdanski K. Extending the Kalman filter for structured identification of linear and nonlinear systems. Int J Modelling, Identif Control. 2017;27(2):114–124.
[10] Julier SJ, Uhlmann JK. A new extension of the Kalman filter to nonlinear systems. Int Soc Optics Photonics (AeroSense '97). 1997;7(28):182–193.
[11] Wan EA, Nelson AT. Dual Kalman filtering methods for nonlinear prediction, smoothing and estimation. Adv Neural Inf Process Syst. 1997;9:793–799.
[12] Wan EA, van der Merwe R. The unscented Kalman filter for nonlinear estimation. Adaptive Systems for Signal Processing, Communications, and Control Symposium. AS_SPCC. The IEEE 2000. IEEE; 2000.
[13] Wielitzka M, Dagen M, Ortmaier T. Joint unscented Kalman filter for state and parameter estimation in vehicle dynamics. 2015 IEEE Conference on Control Applications (CCA). IEEE; 2015.
[14] Ma Z, Zhang Y, Yang J. Velocity and normal tyre force estimation for heavy trucks based on vehicle dynamic simulation considering the road slope angle. Veh Syst Dyn. 2016;54(2):137–167.
[15] Gustafsson F, Gunnarsson F, Bergman N, et al. Particle filters for positioning, navigation, and tracking. IEE Trans Signal Process. 2002;50(2):425–437.
[16] Maskell S, Gordon N. A tutorial on particle filters for on-line nonlinear/non-Gaussian Bayesian tracking. Traget tracking: algorithms and applications (Ref. No. 2001/174), IEE. IET; 2001.

[17] Gordon NJ, Salmond DJ, Smith AFM. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proc F-Radar Signal Process, IET. 1993;140(2):107–113.

[18] Wenzel TA, Burnham KJ, Blundell MV, et al. Dual extended Kalman filter for vehicle state and parameter estimation. Veh Syst Dyn. 2006;44(2):153–171.

[19] Cui Y, Kurfess TR. Vehicle parameter identification for vertical dynamics. J Dyn Syst Meas Control. 2015;137(2):021013.

[20] Russo M, Russo R, Volpe A. Car parameters identification by handling manoeuvres. Veh Syst Dyn. 2000;34(6):423–436.

[21] Best MC, Bogdanski K. Full vehicle and tyre identification using unscented and extended identifying Kalman filters. Proceedings for the 13th International Symposium on Advanced Vehicle Control (AVEC); Munich, Germany; September 2016.

[22] Wu Y,Hu D, Wu M, et al. Unscented Kalman filtering for additive noise case: augmented vs nonaugmented. IEEE Signal Process Lett. 2005;12(5):357–360.

[23] Best MC. Optimisation of high-speed crash avoidance in autonomous vehicles. Int J Veh Auton Syst. 2012;10(4):337–354.

[24] Milliken DL, Milliken WF. Race car vehicle dynamics. Warrendale, PA: SAE International; 1995.

[25] Pacejka HB. Tyre and vehicle dynamics. Woburn, MA: Butterworth Heinemann; 2002.

[26] Best MC, Newton AP. Vehicle tyre and handling model identification using an extended Kalman filter. 9th International Symposium on Advanced Vehicle Control (AVEC '08); Taipei, Taiwan; 2008.