Particle filtering based identification for autonomous nonlinear ODE models ⋆

Jerker Nordh * **Torbjörn Wigren** ** **Thomas B. Schön** **
Bo Bernhardsson *

* *Department of Automatic Control, Lund University, SE-221 00 Lund, Sweden (e-mail: {jerker.nordh, bo.bernhardsson}@control.lth.se).*
** *Department of Information Technology, Uppsala University, SE-751 05, Uppsala, Sweden (e-mail: {torbjorn.wigren, thomas.schon}@it.uu.se.).*

Abstract: This paper presents a new black-box algorithm for identification of a nonlinear autonomous system in stable periodic motion. The particle filtering based algorithm models the signal as the output of a continuous-time second order ordinary differential equation (ODE). The model is selected based on previous work which proves that a second order ODE is sufficient to model a wide class of nonlinear systems with periodic modes of motion, also systems that are described by higher order ODEs. Such systems are common in systems biology. The proposed algorithm is applied to data from the well-known Hodgkin-Huxley neuron model. This is a challenging problem since the Hodgkin-Huxley model is a fourth order model, but has a mode of oscillation in a second order subspace. The numerical experiments show that the proposed algorithm does indeed solve the problem.

Keywords: Autonomous system, identification, neural dynamics, nonlinear systems, oscillation, particle filtering, periodic system, phase plane.

1. INTRODUCTION

The identification of nonlinear autonomous systems is a fairly unexplored field. While a very large amount of work has been devoted to the analysis of autonomous systems, given an ordinary differential equation (ODE), the inverse problem has received much less attention. At the same time system identification based on particle filtering ideas is expanding rapidly. However, little attention has so far been given to identification of nonlinear autonomous systems. The present paper addresses this by presenting a new approach for identification of nonlinear autonomous ODE model based on recently developed particle filtering methods. Furthermore, the paper presents new results on neural modeling, by applying the new algorithm to data generated by the well-known Hodgkin-Huxley model. These constitutes the two main contributions of the paper.

As stated above, much work has been performed on the analysis of a given nonlinear autonomous system, see e.g. Khalil (1996). The classical analysis provided by e.g. Poincaré provide tools for prediction of the existence of periodic orbits of a second order ODE. Bifurcation analysis and similar tools have also been widely applied to the analysis of chaos, inherent in nonlinear autonomous ODEs (Khalil, 1996; Li et al., 2007). There are much less publications on and connections to the inverse problem, i.e. the identification of an autonomous nonlinear ODE from measured data alone. However, algorithms tailored for identification of second order ODEs from periodic data appeared in Wigren et al. (2003a), Manchester et al. (2011) and Wigren (2014). A result on identifiability that gives conditions for when a second order ODE is sufficient for modeling of periodic oscillations is also available, see Wigren and Söderström (2005) and Wigren (2015). That work proves that in case the phase plane of the data is such that the orbit does not intersect itself, then a second order ODE is always sufficient for identification. In other words, higher order models cannot be uniquely identifiable.

There is a vast literature on stable oscillations in biological and chemical systems, see e.g. Rapp (1987). An important example is given by neuron spiking (Doi et al., 2002; Izhikevich, 2003; Hodgkin and Huxley, 1952). This spiking is fundamental in that it is the way nerve cells communicate, for example in the human brain. The field of reaction kinetics provides further examples of dynamic systems that are relevant in the field of molecular systems biology, see e.g. Ashmore (1975). Such kinematic equations typically result in systems of ordinary differential equations with right hand sides where fractions of polynomials in the states appear. The many authors that have dealt with identification of the Hodgkin-Huxley model have typically built on complete models, including an input signal current, see e.g. Doi et al. (2002); Saggar et al. (2007); Tobenkin et al. (2010); Manchester et al. (2011); Lankarany et al. (2013). With the exception of Wigren (2015) there does no seem to have been much work considering the fact that identification may not be possible if only periodic data is available.

⋆ This work was supported by the project *Probabilistic modeling of dynamical systems* (Contract number: 621-2013-5524) funded by the Swedish Research Council. (*) are members of the LCCC Linnaeus Center and the ELLIIT Excellence Center at Lund University.

The particle filter was introduced more that two decades ago as a solution to the nonlinear state estimation problem, see e.g. Doucet and Johansen (2011) for an introduction. However, when it comes to the nonlinear system identification problem in general, it is only relatively recently that the particle filter has emerged as a really useful tool, see Kantas et al. (2014) for a recent survey. The algorithm presented here is an adaptation of the so-called PSAEM algorithm introduced by Lindsten (2013). It provides a solution to the nonlinear maximum likelihood problem by combining the stochastic approximation expectation maximization algorithm of Delyon et al. (1999) with the PGAS kernel of Lindsten et al. (2014). This improves upon the earlier work of Wigren et al. (2003b), since it is no longer necessary to rely on the sub-optimal extended Kalman filter and restrictive models of the model parameters.

2. FORMULATING THE MODEL AND THE PROBLEM

The periodic signal is modeled as the output of a second order differential equation, and can in continuous-time thus be represented as

$$x_t = \begin{pmatrix} p_t & v_t \end{pmatrix}^{\mathsf{T}}, \tag{1a}$$

$$\dot{x}_t = \begin{pmatrix} v_t \\ f(p_t, v_t) \end{pmatrix}. \tag{1b}$$

The discretized model is obtained by a Euler forward approximation and by introducing noise acting on the second state and on the measurement according to

$$p_{t+1} = p_t + hv_t, \tag{2a}$$

$$v_{t+1} = v_t + hf(p_t, v_t) + w_t, \qquad w_t \sim \mathcal{N}(0, Q^w), \tag{2b}$$

$$y_t = p_t + e_t, \qquad e_t \sim \mathcal{N}(0, R). \tag{2c}$$

The noise is only acting on one of the states, implying that one of the states can be marginalized resulting in the following non-Markovian model

$$\begin{aligned} p_{t+1} &= p_t + h(v_{t-1} + hf(p_{t-1}, v_{t-1}) + w_{t-1}) \\ &= 2p_t - p_{t-1} + h^2 f\left(p_{t-1}, \frac{p_t - p_{t-1}}{h}\right) + hw_{t-1}, \end{aligned} \tag{3a}$$

$$y_t = p_t + e_t, \tag{3b}$$

where the noise is still Gaussian according to

$$w_{t-1} \sim \mathcal{N}(0, Q^w), \qquad e_t \sim \mathcal{N}(0, R). \tag{3c}$$

As suggested by Wigren et al. (2003b), the function $f(p, v)$ is parametrized according to

$$f(p, v) = \sum_{i=0}^{m} \sum_{j=0}^{m} a_{ij} p^i v^j. \tag{3d}$$

Here, m is a design parameter deciding the degree of the model used for the approximation and the indices a_{ij} denote unknown parameters to be estimated together with the process noise covariance Q^w. Hence, the unknown parameters to be estimated are given by

$$\theta = \{Q^w \quad a_{00} \quad \ldots \quad a_{0m} \quad \ldots \quad a_{m0} \quad \ldots \quad a_{mm}\}. \tag{4}$$

It has been assumed that the measurement noise covariance R is known. The problem under consideration is that of computing the maximum likelihood (ML) estimate of the unknown parameters θ by solving

$$\widehat{\theta}_{\mathrm{ML}} = \operatorname*{argmax}_{\theta} \log p_\theta(y_{1:T}), \tag{5}$$

where $y_{1:T} = \{y_1, \ldots, y_T\}$ and $p_\theta(y_{1:T})$ denotes the likelihood function parameterized by θ.

3. PARTICLE FILTERING FOR AUTONOMOUS SYSTEM IDENTIFICATION

After the marginalization of the v state in model (2) the problem becomes non-Markovian, for an introduction to non-Markovian particle methods see e.g., Lindsten et al. (2014) and Lindsten and Schön (2013). The remainder of this section will go through the components required for the algorithm and note specific design choices made to apply the methods to the particular class of problems that are of interest in this paper.

3.1 Expectation maximization algorithms

The expectation Maximization (EM) algorithm (Dempster et al., 1977) is an iterative algorithm to compute ML estimates of unknown parameters (here θ) in probabilistic models involving latent variables (here, the state trajectory $x_{1:T}$). More specifically, the EM algorithm solves the ML problem (5) by iteratively computing the so-called *intermediate quantity*

$$\mathcal{Q}(\theta, \theta_k) = \int \log p_\theta(x_{1:T}, y_{1:T}) p_{\theta_k}(x_{1:T} \,|\, y_{1:T}) \mathrm{d}x_{1:T} \tag{6}$$

and then maximizing $\mathcal{Q}(\theta, \theta_k)$ w.r.t. θ. There is now a good understanding of how to make use of EM-type algorithms to identify dynamical systems. The linear state space model allows us to express everything in closed form (Shumway and Stoffer, 1982; Gibson and Ninness, 2005). However, when it comes to nonlinear models, like the ones considered here, approximate methods have to be used, see e.g. (Lindsten, 2013; Schön et al., 2011; Cappé et al., 2005).

The sequential Monte Carlo (SMC) methods (Doucet and Johansen, 2011) or the particle Markov chain Monte Carlo (PMCMC) methods introduced by Andrieu et al. (2010) can be exploited to approximate the joint smoothing density (JSD) arbitrarily well according to

$$\widehat{p}(x_{1:T} \,|\, y_{1:T}) = \sum_{i=1}^{N} w_T^i \delta_{x_{1:T}^i}(x_{1:T}). \tag{7}$$

Here, $x_{1:T}^i$ denotes the samples (also referred to as particles, motivating the name particle filter/smoother), w_T^i denotes the corresponding weights and δ_x denotes a point-mass distribution at x. Schön et al. (2011) used the SMC approximation (7) to approximate the intermediate quantity (6). However, there is room to make even more efficient use of the particles in performing ML identification, by making use of the *stochastic approximation* developments within EM according to Delyon et al. (1999). In the so-called stochastic approximation expectation maximization (SAEM) algorithm, the intermediate quantity (6) is replaced by the following stochastic approximation update

$$\widehat{\mathcal{Q}}_k(\theta) = (1 - \gamma_k)\widehat{\mathcal{Q}}_{k-1}(\theta) + \gamma_k \log p_\theta(x_{1:T}[k], y_{1:T}), \tag{8}$$

where γ_k denotes the step size, which is a design parameter that must fulfill $\sum_{k=1}^{\infty} \gamma_k = \infty$ and $\sum_{k=1}^{\infty} \gamma_k^2 < \infty$. Furthermore, $x_{1:T}[k]$ denotes a sample from the JSD $p_{\theta_k}(x_{1:T} \,|\, y_{1:T})$. The sequence θ_k generated by the SAEM algorithm outlined above will under fairly weak assumptions converge to a maximizer of $p_\theta(y_{1:T})$ (Delyon et al., 1999).

For the problem under consideration the recently developed PMCMC methods (Andrieu et al., 2010; Lindsten et al., 2014) are useful to approximately generate samples from the JSD. This was realized by Lindsten (2013), resulting in the so-called particle SAEM (PSAEM) algorithm, which is used in this work.

3.2 The PGAS kernel

The particle Gibbs with ancestor sampling (PGAS) kernel was introduced by Lindsten et al. (2014). It is a procedure very similar to the standard particle filter, save for the fact that conditioning on one so-called *reference trajectory* $x'_{1:T}$ is performed. Hence, $x'_{1:T}$ have to be retained throughout the sampling procedure. For a detailed derivation see Lindsten et al. (2014), where it is also shown that the PGAS kernel implicitly defined via Algorithm 1 is uniformly ergodic. Importantly, it also leaves the target density $p(x_{1:T} \mid y_{1:T})$ invariant for any finite number of particles $N > 1$ implying that the resulting state trajectory $x^\star_{1:T}$ can be used as a sample from the JSD. The notation used in Algorithm 1 is as follows, $\mathbf{x}_t = (x_t^1, \ldots, x_t^N)$ denotes all the particles at time t and $\mathbf{x}_{1:T} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$ the entire trajectories. The particles are propagated according to a proposal distribution $r_t(x_t \mid x_{t-1}, y_t)$. The resampling step and the propagation step of the standard particle filter has been collapsed into jointly sampling the particles $\{x_t^i\}_{i=1}^N$ and the ancestor indices $\{a_t^i\}_{i=1}^N$ independently from

$$M_t(a_t, x_t) = \frac{w_t^{a_t}}{\sum_{l=1}^N w_t^l} r_t(x_t \mid x_{1:t-1}^{a_t}, y_{1:t}). \tag{9}$$

Finally, W_t denotes the weight function,

$$W_t(x_{1:t}, y_{1:t}) = \frac{p(y_t \mid x_{1:t}) p(x_t \mid x_{1:t-1})}{r(x_t \mid x_{1:t-1}, y_{1:t})}. \tag{10}$$

Algorithm 1 PGAS kernel

1: **Initialization ($t = 1$):** Draw $x_1^i \sim r_1(x_1 | y_1)$ for $i = 1, \ldots, N-1$ and set $x_1^N = x'_1$. Compute $w_1^i = W_1(x_1^i)$ for $i = 1, \ldots, N$.
2: **for** $t = 2$ to T **do**
3: Draw $\{a_t^i, x_t^i\} \sim M_t(a_t, x_t)$ for $i = 1, \ldots, N-1$.
4: Set $x_t^N = x'_t$.
5: Draw a_t^N with $\mathbb{P}\left(a_t^N = i\right) \propto \frac{w_{t-1}^i p(x'_t \mid x_{1:t-1}^i)}{\sum_{l=1}^N w_{t-1}^l p(x'_t \mid x_{1:t-1}^l)}$
6: Set $x_{1:t}^i = \{x_{1:t-1}^{a_t^i}, x_t^i\}$ for $i = 1, \ldots, N$.
7: Compute $w_t^i = W_t(x_{1:t}^i, y_{1:t})$ for $i = 1, \ldots, N$.
8: **end for**
9: **Return** $\mathbf{x}_{1:T}, \mathbf{w}_T$.

3.3 Identifying autonomous systems using PSAEM

The PSAEM algorithm for ML identification of autonomous systems now simply amounts to making use of the PGAS kernel in Algorithm 1 to generate a particle system $\{x_{1:T}^i, w_T^i\}_{i=1}^N$ that is then used to approximate the intermediate quantity according to

$$\widehat{\mathcal{Q}}_k(\theta) = (1 - \gamma_k)\widehat{\mathcal{Q}}_{k-1}(\theta) + \gamma_k \sum_{i=1}^N w_T^i \log p_\theta(x_{1:T}^i, y_{1:T}). \tag{11}$$

Note that similarly to (8) only the reference trajectory $x_{1:T}[k]$ could have been used, but in making use of the entire particle system the variance of the resulting state estimates are reduced (Lindsten, 2013). The result is provided in Algorithm 2.

Algorithm 2 PSAEM for sys. id. of autonomous systems

1: **Initialization:** Set $\theta[0] = (Q_0^w \quad 0^\mathsf{T})$ and set $x_{1:T}[0]$ using an FFBSi particle smoother. Set $\widehat{\mathcal{Q}}_0 = 0$ and set $\mathbf{w}[0]$ to an empty vector.
2: Draw $x'_{1:T}$ using FFBSi.
3: **for** $k \geq 1$ **do**
4: Draw $\mathbf{x}_{1:T}[k], \mathbf{w}_T$ by running Algorithm 1 using $x'_{1:T}$ as reference.
5: Draw j with $\mathbb{P}(j = i) = w_T^i$.
6: Set $x'_{1:T} = x_{1:T}^j[k]$
7: Set $\mathbf{w}[k] = ((1 - \gamma_k)\mathbf{w}[k-1] \quad \gamma_k \mathbf{w}_T)$
8: Compute $\widehat{\mathcal{Q}}_k(\theta)$ according to (11).
9: Compute $\theta[k] = \operatorname{argmax} \widehat{\mathcal{Q}}_k(\theta)$.
10: **if** termination criterion is met **then**
11: **return** $\{\theta[k]\}$
12: **end if**
13: **end for**

Note that the initial reference trajectory $x_{1:T}[0]$ is obtained by running a so-called forward filter backward simulator (FFBSi) particle smoother, see Lindsten and Schön (2013) for details. To indicate that the trajectories were generated at iteration k, we use $\mathbf{x}_{1:T}[k]$ and analogously for the weights. The 9$^{\text{th}}$ row of Algorithm 2 will for the model (3) under consideration amount to a weighted least squares problem, which is solved in Algorithm 3. For the work presented in this article the run Algorithm 2 for a fixed number of iterations, which gives the termination criterion.

Algorithm 3 Maximizing $\mathcal{Q}$

1: For each trajectory in $\mathbf{x}_{1:T}[k]$ calculate the velocity at each time $v_t^{(i)} = (p_{t+1}^{(i)} - p_t^{(i)})/h$
2: For each time step and for each trajectory in $\mathbf{x}_{1:T}[k]$, evaluate $f(p_t^{(i)}, v_t^{(i)})$.
3: Find $a_{00} \ldots a_{mm}$ via the related weighted least squares (WLS) problem.
4: Find Q^w by estimating the covariance of the residuals of the WLS-problem.
5: Set $\theta[k] = \{Q^w \quad a_{00} \quad \ldots \quad a_{mm}\}$.

3.4 Choosing the proposal distribution

The proposal density $r(x_t | x_{1:t-1}, y_{1:t})$ constitutes an important design choice of the particle filter that will significantly affect its performance. The commonly used bootstrap particle filter amounts to making use of the dynamics to propose new particles, i.e. $r(x_t | x_{1:t-1}, y_{1:t}) = p(x_t \mid x_{1:t-1})$. However, we will make use of the measurement model and the information present in the current measurement to propose new particles, i.e. $r(x_t | x_{1:t-1}, y_{1:t}) = p(y_t \mid x_t)$. This is enabled by the marginalization of the deterministic state in the model, since the dimension of the state-space then matches that of the measurement. As a possible further improvement, the proposal could also include the predicted state using the model. Recently, Kronander and Schön (2014) showed that the combined use of both the dynamics and the measurements results in competitive algorithms. In such a scenario the estimated uncertainty in the model would initially be large and thus not affect the proposal distribution significantly, but for each iteration of the PSAEM algorithm the model will be

more and more accurate in predicting the future states, and its influence on the proposal distribution would increase accordingly.

4. NUMERICAL ILLUSTRATIONS

The performance of the proposed algorithm is illustrated using two examples, namely the Van det Pol oscillator in Section 4.1 and the Hodgkin-Huxley neuron model in Section 4.2 – 4.3. There is no prior knowledge of the model, hence it is assumed that all the parameters a_{ij} in (3d) are zero. The initial covariance Q_0^w is set to a large value. The Python source code for the following examples can be downloaded from Nordh (2015), the implementation was carried out using the pyParticleEst software framework (Nordh, 2013).

4.1 Validation on the Van der Pol oscillator

To demonstrate the validity of the proposed solution it is first applied to the classic Van der Pol oscillator (Khalil, 1996), which belongs to the model class defined by (3). The oscillator is described by

$$\dot{x}_t = \begin{pmatrix} v_t \\ -p_t + 2(1 - p_t^2)v_t \end{pmatrix}, \tag{12}$$

where $x_t = (p_t \quad v_t)^\mathsf{T}$. Performing 50 iterations of Algorithm 2 gives the parameter convergence shown in Fig. 1. Here $N = 15$ particles were used for the PGAS sampler, and the model order was chosen as $m = 2$. For the SAEM step, the sequence $\gamma_{1:K}$ is chosen as

$$\gamma_k = \begin{cases} 1 & \text{if } k \leq 5, \\ (k-5)^{-0.9} & \text{if } k > 5. \end{cases} \tag{13}$$

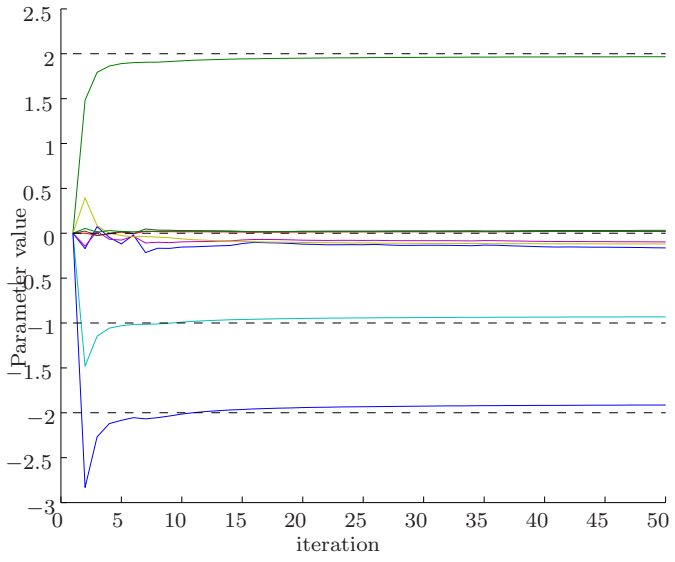

Fig. 1. Parameter convergence for the Van der Pol example (12). The dashed lines are the coefficients obtained from the discretization of the model, the other lines represent the different parameters in the identified model. Note that while the parameters do not converge to the 'true' values, they provide a very accurate model for predicting the signal as shown in Fig. 3.

It can be seen that the parameters converge to values close to those obtained from the Euler forward discretization. To analyze the behavior further the phase-plane for the system is shown in Fig. 2 and the time-domain realization in Fig. 3. Here it can be seen that the identified model captures the behavior of the true continuous-time system significantly better than the model obtained from the discretization.

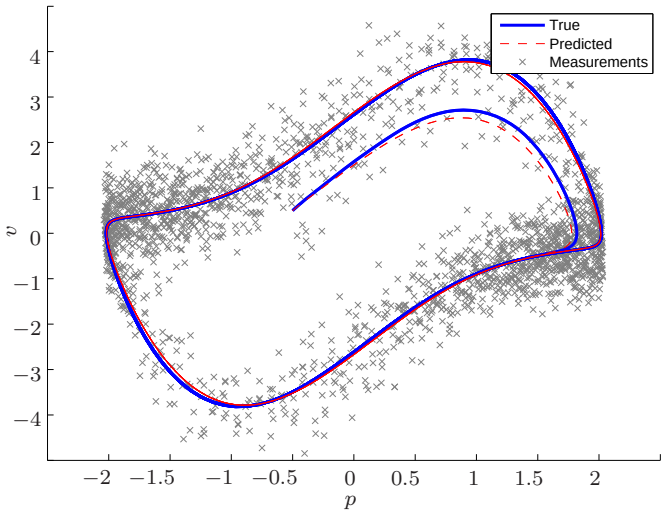

Fig. 2. Phase-plane plot for the Van der Pol example (12).

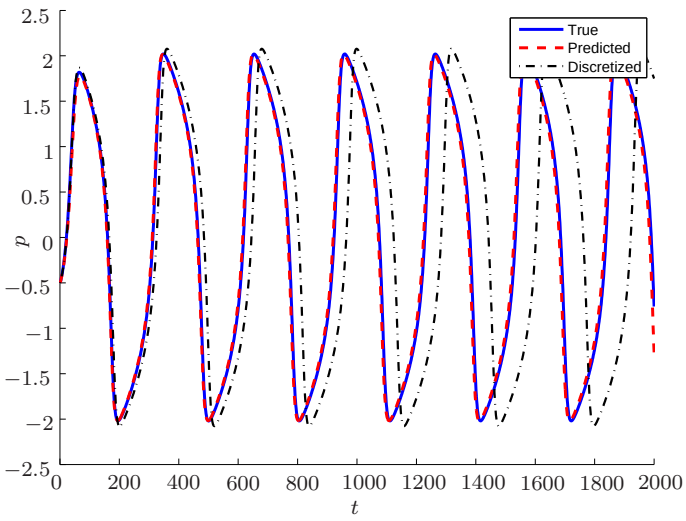

Fig. 3. Predicted output using the true initial state and the estimated parameters. The plot labeled "discretized" is obtained by discretization of the true continuous-time model using Euler forward and using that to predict the future output. It can been seen that the discretized model diverges over time from the true signal, clearly the identified parameter values give a better estimate than using the values from the discretization.

4.2 The Hodgkin-Huxley neuron model

The well-known Hodgkin-Huxley model uses a nonlinear ODE to describe the dynamics of the action potentials in a neuron. In this paper the model will be used for two purposes. First, simulated spiking neuron data is used to characterize the performance of the proposed algorithm when identifying a nonlinear autonomous system. It should be noted that the data does not correspond to a system

that is in the model set. The ability to handle nonlinear under-modeling is therefore also assessed. Secondly, the new algorithm and the identification results contribute to an enhanced understanding of spiking neurons by providing better performance compared to previous algorithms.

The Hodgkin-Huxley model formulation of Siciliano (2012) is used, where the following ODE is given

$$\frac{dv}{dt} = \frac{1}{C_m}\left[I - g_{na}m^3h(v-E_{na}) \times g_K n^4(v-E_K) - g_l(v-E_l)\right], \tag{14a}$$

$$\frac{dn}{dt} = \alpha_n(v)(1-n) - \beta_n(v)n, \tag{14b}$$

$$\frac{dm}{dt} = \alpha_m(v)(1-m) - \beta_m(v)m, \tag{14c}$$

$$\frac{dh}{dt} = \alpha_h(v)(1-h) - \beta_h(v)h. \tag{14d}$$

Here, v denotes the potential, while n, m and h relate to each type of gate of the model and their probabilities of being open, see Siciliano (2012) for details. The applied current is denoted by I. The six rate variables are described by the following nonlinear functions of v

$$\alpha_n(v) = \frac{0.01(v+50)}{1-e^{-(v+50)/10}}, \tag{15a}$$

$$\beta_n(v) = 0.125e^{-(v+60)/80}, \tag{15b}$$

$$\alpha_m(v) = \frac{0.1(v+35)}{1-e^{-(v+35)/10}}, \tag{15c}$$

$$\beta_m(v) = 4.0e^{-0.0556(v+60)}, \tag{15d}$$

$$\alpha_h(v) = 0.07e^{-0.05(v+60)}, \tag{15e}$$

$$\beta_h(v) = \frac{1}{1+e^{-0.1(v+30)}}. \tag{15f}$$

The corresponding numerical values are given by $C_m = 0.01\ \mu F/cm^2$, $g_{Na} = 1.2\ mS/cm^2$, $E_{Na} = 55.17\ mV$, $g_K = 0.36\ mS/cm^2$, $E_K = -72.14\ mV$, $g_l = 0.003\ mS/cm^2$, and $E_l = -49.42\ mV$.

4.3 Identifying the spiking mode of the Hodgkin-Huxley model

Using a simulated dataset of length $T = 10\,000$, a model of the form (3) with $m = 3$ is selected. Algorithm 2 was run for 200 iterations, employing $N = 20$ particles in the PGAS kernel. For the SAEM step, the sequence $\gamma_{1:K}$ is chosen as

$$\gamma_k = \begin{cases} 1 & \text{if } k \leq 40, \\ (k-40)^{-0.5} & \text{if } k > 40. \end{cases} \tag{16}$$

The predicted phase-plane of the identified model is shown in Fig. 4 along with the true phase-plane and the measurements.

Fig. 5 shows the same dataset in the time-domain, it shows the true signal (without added noise) and the output predicted by the identified model when initialized with the correct initial state. It can be seen that the model captures the behaviour of the signal, but introduces a small error in the frequency of the signal, causing the predicted and true signal to diverge over time.

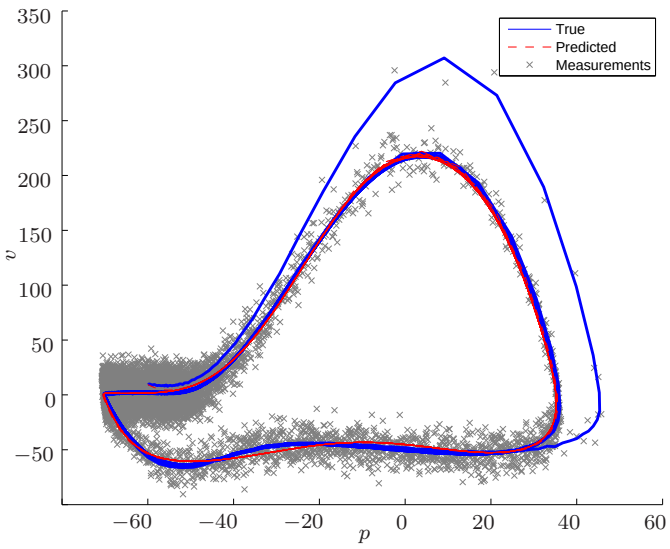

Fig. 4. Phase-plane for the Hodgkin-Huxley dataset. It can be seen that the predicted model fails to capture the initial transient, but accurately captures the limit cycle.

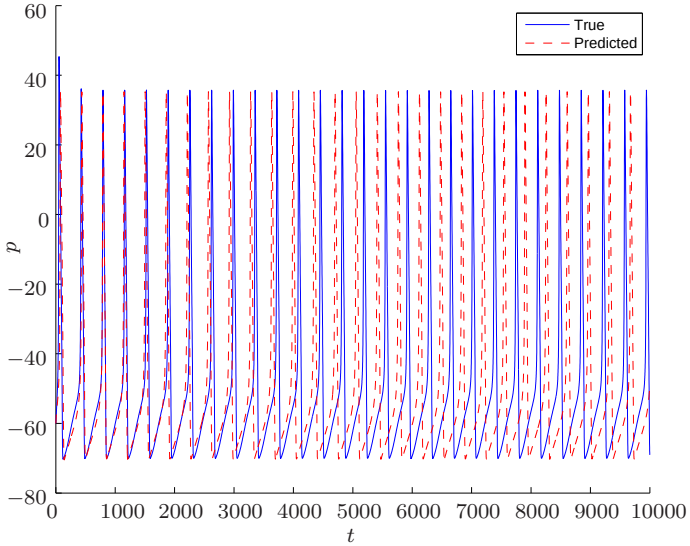

Fig. 5. Predicted output using the true initial state and the estimated parameters. The overall behaviour of the signal is captured, but there is a small frequency error which leads to the predicted signal slowly diverging from the true signal.

5. CONCLUSIONS

The new identification method successfully identified models for both the Van der Pol example and for the more complex Hodgkin-Huxley model. Even though the Hodgkin-Huxley model in general cannot be reduced to a second order differential equation it is possible to identify a good second order model for its limit cycle as shown in Fig. 4.

The use of a Bayesian nonparametric model in place of (3d) constitutes an interesting continuation of this work. A first step in this direction would be to employ the Gaussian process construction by Frigola et al. (2013, 2014).

REFERENCES

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society, Series B*, 72(2), 1–33.

Ashmore, P.G. (1975). *Kinetics of Oscillatory Reactions*. The Chemical Society.

Cappé, O., Moulines, E., and Rydén, T. (2005). *Inference in Hidden Markov Models*. Springer Series in Statistics. Springer, New York, USA.

Delyon, B., Lavielle, M., and Moulines, E. (1999). Convergence of a stochastic approximation version of the EM algorithm. *The Annals of Statistics*, 27(1), 94–128.

Dempster, A.P., Laird, N.M., and Rubin, D.B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, 39(1), 1–38.

Doi, S., Onada, Y., and Kumagai, S. (2002). Parameter estimation of various Hodgkin-Huxley-type neuronal models using a gradient-descent learning method. In *Proceedings of the 41st SICE Annual Conference*, 1685–1688. Osaka, Japan.

Doucet, A. and Johansen, A.M. (2011). A tutorial on particle filtering and smoothing: Fifteen years later. In D. Crisan and B. Rozovsky (eds.), *Nonlinear Filtering Handbook*. Oxford University Press.

Frigola, R., Lindsten, F., Schön, T.B., and Rasmussen, C.E. (2013). Bayesian inference and learning in Gaussian process state-space models with particle MCMC. In *Advances in Neural Information Processing Systems (NIPS) 26*, 3156–3164.

Frigola, R., Lindsten, F., Schön, T.B., and Rasmussen, C.E. (2014). Identification of Gaussian process state-space models with particle stochastic approximation EM. In *Proceedings of the 18th World Congress of the International Federation of Automatic Control (IFAC)*. Cape Town, South Africa.

Gibson, S. and Ninness, B. (2005). Robust maximum-likelihood estimation of multivariable dynamic systems. *Automatica*, 41(10), 1667–1682.

Hodgkin, A.L. and Huxley, A.F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117(4), 500–544.

Izhikevich, E.M. (2003). Simple models of spiking neurons. *IEEE Transactions on Neural Networks*, 14(6), 1569–1572.

Kantas, N., Doucet, A., Singh, S.S., Maciejowski, J., and Chopin, N. (2014). On particle methods for parameter estimation in state-space models. ArXiv:1412.8695, submitted to Statistical Science.

Khalil, H.K. (1996). *Nonlinear Systems*. Prentice Hall, Upper Saddle River.

Kronander, J. and Schön, T.B. (2014). Robust auxiliary particle filters using multiple importance sampling. In *Proceedings of the IEEE Statistical Signal Processing Workshop (SSP)*. Gold Coast, Australia.

Lankarany, M., Zhu, W.P., and Swamy, N.S. (2013). Parameter estimation of Hodgkin-Huxley neuronal model using dual extended Kalman filter. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, 2493–2496. Beijing, China.

Li, H.Y., Che, Y.Q., Gao, H.Q., Dong, F., and Wang, J. (2007). Bifurcation analysis of the Hodgkin-Huxley model exposed to external DC electric field. In *Proceedings of the 22nd International Symposium on Intelligent Control (ISIC)*, 271–276. Singapore.

Lindsten, F. (2013). An efficient stochastic approximation EM algorithm using conditional particle filters. In *Proceedings of the 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Vancouver, Canada.

Lindsten, F., Jordan, M.I., and Schön, T.B. (2014). Particle Gibbs with ancestor sampling. *Journal of Machine Learning Research (JMLR)*, 15, 2145–2184.

Lindsten, F. and Schön, T.B. (2013). Backward simulation methods for Monte Carlo statistical inference. *Foundations and Trends in Machine Learning*, 6(1), 1–143.

Manchester, I., Tobenkin, M.M., and Wang, J. (2011). Identification of nonlinear systems with stable oscillations. In *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, 5792–5797. Orlando, FL, USA.

Nordh, J. (2013). pyParticleEst software framework. URL `http://www.control.lth.se/Staff/JerkerNordh/pyparticleest.html`.

Nordh, J. (2015). Example source code for "Particle filtering based identification for autonomous nonlinear ode models". URL `http://www.control.lth.se/Staff/JerkerNordh/ode-id.html`.

Rapp, P. (1987). Why are so many biological systems periodic? *Progress in Neurobiology*, 29(3), 261–273.

Saggar, M., Mericli, T., Andoni, S., and Miikulainen, R. (2007). System identification for the Hodgkin-Huxley model using artificial neural networks. In *Proceedings of the International Joint Conference on Neural Networks*. Orlando, FL, USA.

Schön, T.B., Wills, A., and Ninness, B. (2011). System identification of nonlinear state-space models. *Automatica*, 47(1), 39–49.

Shumway, R.H. and Stoffer, D.S. (1982). An approach to time series smoothing and forecasting using the EM algorithm. *Journal of Time Series Analysis*, 3(4), 253–264.

Siciliano, R. (2012). The Hodgkin–Huxley model – its extensions, analysis and numerics. Technical report, Department of Mathematics and Statistics, McGill University, Montreal, Canada.

Tobenkin, M.M., Manchester, I., Wang, J., Megretski, A., and Tedrake, R. (2010). Convex optimization in identification of stable non-linear state space models. In *Proceedings of the 49th IEEE Conference on Decision and Control (CDC)*, 7232–7237. Atlanta, USA.

Wigren, T. (2014). Software for recursive identification of second order autonomous systems. Technical report, Department of Information Technologt, Uppsala University, Uppsala, Sweden. URL `http://www.it.uu.se/research/publications/reports/2014-014/SWAutonomous.zip`.

Wigren, T. (2015). Model order and identifiability of non-linear biological systems in stable oscillation. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. Accepted for publication.

Wigren, T., Abd-Elrady, E., and Söderström, T. (2003a). Harmonic signal analysis with Kalman filters using periodic orbits of nonlinear ODEs. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 669–672. Hong Kong, PRC.

Wigren, T. and Söderström, T. (2005). A second order ODE is sufficient for modeling of many periodic systems. *International Journal of Control*, 77(13), 982–986.

Wigren, T., Abd-Elrady, E., and Söderström, T. (2003b). Least squares harmonic signal analysis using periodic orbits of ODEs. In *Proceeding of 13th IFAC symposium on system identification (SYSID)*, 1584–1589. Rotterdam, The Netherlands.