Engineering Applications of Artificial Intelligence 38 (2015) 24–33

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

An efficient method for the numerical integration of measured variable dependent ordinary differential equations

C. Rodrigues de Azevedo [a], J. Peres [b], M. von Stosch [a,*]

[a] REQUIMTE, Departamento de Química, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, 2829-516 Caparica, Portugal
[b] LEPABE, Departamento de Engenharia Química, Faculdade de Engenharia, Universidade do Porto, Rua Dr. Roberto Frias s/n, 4200-465 Porto, Portugal

ARTICLE INFO

Article history:
Received 31 May 2014
Received in revised form
7 October 2014
Accepted 14 October 2014
Available online 12 November 2014

Keywords:
Dynamic neural networks
Hybrid semi-parametric modeling
Numerical integration
Ordinary differential equations
Differential equations

ABSTRACT

The Ordinary Differential Equations (ODEs) of dynamic models that are used in process monitoring, control or optimization, are not only functions of time and states, but also of measured variables. So far two possibilities for the numerical integration of such ODEs were given: (i) a fixed step size integration schema could be applied, matching the step size to the time instances of the measurements; or (ii) using an adaptive step size method while interpolating the measurements. While fixed step size methods are computationally expensive, the repetitive interpolation of measurements for the application of adaptive step size methods is prone to errors and time prohibitive, especially for great numbers of measured variables.

In this paper, an adaptive step size numerical integration method is proposed and evaluated for dynamic neural network/hybrid semi-parametric models. The method evaluates the ODEs only at time instances at which online measurements are available and adapts the step size according to those time instances. The numerical solution of the ODEs is provided at time instances which are specified by the user, i.e. at time instances of offline measured states. The rationale behind the proposed method is carefully analyzed, and it is demonstrated that its application along with a hybrid model/dynamic neural network model can result into a significant reduction of number of function evaluations, in the studied cases about 50%, while adhering user specified error tolerances for the numerical integration. In addition, the mutual interference between step-size adaption, parameter identification, coping of the neural network and model performance is studied, a fact that other studies have paid little to no attention.

1. Introduction

In the area of chemical and biochemical engineering the list of quantities that can be measured increased significantly in the last decade (Schuegerl, 2001). This development gave rise to the application of data-driven techniques for process modeling. In particular Neural Networks (NNs) have found wide application, since they can easily be applied for fast nonlinear process model development. Most (bio)chemical processes are dynamic, wherefore the standard, rather static, concept of NNs has been extended in many different ways in order to yield a dynamic model (Sinha et al., 2000). One peculiar way is modeling the time varying functions of Ordinary Differential Equations (ODEs) by NNs, resulting in dynamic neural networks (DNN) (Petre et al., 2010). This approach preserves some structural resemblance to the modeling of dynamic systems by first-principles, i.e. a set of ODEs. To an even bigger extend hybrid semi-parametric modeling exploits the available first-principles knowledge, represented in from of parametric models, by combining it with nonparametric models that are identified from data (Thompson and Kramer, 1994, von Stosch et al., 2014). In general, the model structure of DNNs or dynamic hybrid semi-parametric models can be expressed as a set of Ordinary Differential Equations (ODEs), i.e.,

$$\frac{dx}{dt} = f(x, z, t, w), \quad x(t_o) = x_o, \tag{1}$$

where x is a vector of state variables, z is a vector of measured variables, t is the time, w represents all model parameters and f is a vector of rate functions describing how the states change along with time. In case of the DNN models, these functions, f, are modeled by NNs. In case of hybrid semi-parametric models, these functions are combinations of NNs with fundamental knowledge, as e.g. presented in Oliveira (2004).

Depending on the arguments of f, the model structure can either be a one-step ahead predictor, namely if measured variables, z, are

* Corresponding author. Tel.: +351 212948313; fax: +351 212948550.
E-mail address: m.stosch@fct.unl.pt (M. von Stosch).

http://dx.doi.org/10.1016/j.engappai.2014.10.014
0952-1976/

incorporated or a multi-step ahead predictor if $f = f(x, t, w)$. In any case, Eq. (1) needs to be integrated to obtain the state estimations x, which is usually done numerically. Whereas in the case of multi-step ahead predictors traditional numerical integration schemas can readily be applied (e.g. the ode45 of the MATLAB toolbox), this is not the case for one-step ahead prediction structures, since the value of the z might not be measured/available at the time instances at which f is evaluated by the traditional schemas. Thus, for their integration either a fixed-step size schema, such as the well known Euler or Heun is adopted, fitting the step size to the sampling frequency of z, or some approach is adapted to represent z as a function of time, e.g. spline or polynomial approaches, and then standard adaptive integration schema is applied, as for multi-step ahead predictors. Both approaches are computationally inefficient. Fixed step size methods allocate the function evaluations on a rigid grid, completely disregarding f's curvature. Interpolation becomes computational expensive when repetitively carried out as e.g. for network structure discrimination and network training. Moreover for other types of measured variables, such as those obtained from spectroscopic devices (e.g. Near InfraRed (NIR)), the application of interpolation techniques might not even be appropriate, since the error introduced by the interpolation technique can be great.

Another issue at stake is to what extend the training of the nonparametric model, i.e. parameter identification, will compensate for the numerical error introduced due to an inappropriate step-size. While it can generally be expected that the overall error will decrease due to the compensation, the function that is sought to be modeled is distorted.

In what follows, a numerical integration method is proposed that provides the solution within user specified tolerances at the desired time instances by varying the step size, but that only evaluates the function $f(x(t_z), z(t_z), t_z, w)$ at those time instances t_z at which measured variables z are available. The proposed methodology is (1) analyzed using a simulation case study and (2) evaluated in comparison to fixed step size methods on an experimental case study. In addition the mutual interference between step size, parameter identification, learning of the neural network and model performance is assessed.

2. Methodology

Two conditions arise for the numerical integration:

2.1. Condition 1 (C1)

For parameter identification or for model performance assessment, Eq. (1) needs to be solved at those time instances at which state measurements were made. Most numerical integration methods are capable of this. The methods usually either integrate from time instance to time instance or they integrate over the whole time interval and interpolate thereafter (Shampine, 1987), the post-interpolation usually being computationally less expensive.

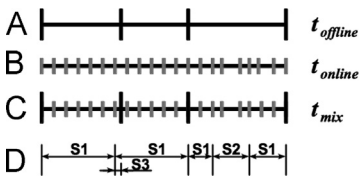

Fig. 1. (A) Timeline at which the ODE solution is desired e.g. at the time-instances of offline measured state variables; (B) timeline of online measured variables; (C) timeline comprising timelines A and B; and (D) classification of the timeline C into sections S1–S3.

2.2. Condition 2 (C2)

For the numerical integration of Eq. (1), the measured variables, z, needs to be provided at those time instances t_z at which the function $f(x(t_z), z(t_z), t_z, w)$ is evaluated. Even though z is measured with high frequency (implied through its choice as input) it might not be available at t_z. To obtain $z(t_z)$, Hermite cubic interpolations or smoothing spline interpolation can be applied for some types of measured variables, but during the network structure discrimination, the network training or process optimization the repetitive interpolation becomes computationally expensive. For other types of measured variables, such as those obtained from spectroscopic devices (e.g. Near InfraRed (NIR)), the application of interpolation techniques might not even be appropriate, because (i) the error introduced by the interpolation technique can be large; and (ii) the number of measured variables is large, wherefore interpolations come with a significant time and computational burden. In order to avoid (repetitive) interpolations, fixed step size methodologies find application in which the integration step-size is determined by the sampling frequency of the measured variables z (Schubert et al., 1994; van Can et al., 1996; von Stosch et al., 2012). However, it is well known that fixed step size methodologies are, likewise, not computationally efficient.

The sampling rate of online measured variables, z, is in most (bio) chemical processes both time constant and frequent. In contrast, the proportions of the divisions made for each step in most standard numerical integration schema, such as Runge–Kutta Fehlberg (4–5) or Dormand–Price (4–5) (Gladwell et al., 2003, Ashino and Vaillancourt, 2009) are irregular. Therefore their direct application along with the frequently sampled online measurements is hindered. In the following the intermeshing use of four integration methods, all of which having different but regular step proportions, is proposed. Since their application comes with different requirements on two consecutive steps, the time interval of the integration is subdivided into different sections, as explained in the following. For each of these sections a different integration schema is applied. A schema for the control of the integration error is proposed, in which the adaptation of the step size is carried out with respect to the given sampling time instances of the measurements.

2.3. Time instance analysis and classification of equal property sections

The proposed numerical integration method has to obey to two timelines, (1) the timeline at which the solution of the ODE is desired, t_{Xmes}, i.e. the time instances at which infrequent offline measurements have been obtained and at which the solution of the ODE is compared to the measurements (e.g. for parameter identification see Section 3.1), see Fig. 1A; and (2) the timeline of the frequent online measurements, t_z, whose entries determine the minimal possible step sizes that can be chosen, see Fig. 1B. The sampling frequency of $z(t_z)$ is typically constant throughout the experiment, such that the increments between the time instances contained in t_z do all have the same size. However, online measurements might not exist at the time instances of the offline measurements at which the solution is desired, i.e. the time instance elements of t_{Xmes} do not have to be contained in t_z, see Fig. 1. In order to obey to these given conditions, it is proposed to divide the timeline into distinct sections such that for the numerical integration of each section a different methodology can be applied. In particular, the time vector t_{mix} (which contains all the sampling times of t_{Xmes} & t_z) is divided into the following categories, which are represented in Fig. 1:

S1) Sections with (at least) two equal sequential steps; (equal in the context of numerics meaning up to eight digits behind the point, 10^{-8});