11th World Congress on Computational Mechanics (WCCM XI)
5th European Conference on Computational Mechanics (ECCM V)
6th European Conference on Computational Fluid Dynamics (ECFD VI)
E. Oñate, J. Oliver and A. Huerta (Eds)

TOWARDS ON-LINE STATE TRACKING WITH DATA-DRIVEN PROCESS MODELS

SUSANNE FISCHER* AND NORBERT LINK*

* Institute of Applied Research
Karlsruhe University of Applied Sciences
Moltkestr. 30, 76133 Karlsruhe, Germany
e-mail: susanne.fischer@hs-karlsruhe.de, web page: http://www.hs-karlsruhe.de

Key words: Non-linear Dimensionality Reduction, Inverse Problems, Symbolic Regression, Identification.

Abstract. On-line optimization of manufacturing processes to increase the production efficiency is of growing interest. In order to reduce cost and improve the quality of the result the most efficient process path (optimal process parameters) has to be found. This requires on-line analysis and tracking of the evolution of the process state, based on few observable variables.
On-line state tracking with data-driven process models intends to optimize processes by learning process models to derive the state variable values (i. e. material and geometrical properties) during the manufacturing process. Process simulation is used in an observer model to perform state tracking. Common simulation methods like the finite element method or phase field models are computationally too expensive for on-line state tracking. Therefore, dimensionality reduction methods are analysed for the identification of a light-weight dynamical process model to perform state tracking in real-time. Methods such as symbolic regression are examined to identify a dynamical model based on the determined state features. The feasibility of extracting state features and identifying a reduced representation of the state is shown through numerical simulations of deep drawing process models.

1 INTRODUCTION

For many manufacturing processes it is technically or economically not feasible to measure state variables while the process is running. State variables characterize the state of the workpiece and are required for the process control. However, there is the capability to observe several measurements on-line that interrelate with the state. Methods for extracting the state variables from these observables using artificial neural networks have been presented by Senn [1]. One of the goals of state tracking is to control the process chain so that the process parameters are set cost-efficiently to obtain the optimal final product. To implement dynamic state tracking we use an observer similar to the Luenberger observer [2].

The main components of the observer are the process model, the measurement model and the correction model. All these components run in parallel to the real process (figure 1). The process model predicts the state variables. Therefore, we use machine learning algorithms to identify the dynamic process model based on simulation and/or experimental data. The measurement model maps these predicted state variables to observables so they can be compared to the real observables. The differences between the real and predicted observables take effect as observer gain through the correction model so that the observer can adjust the parameters and/or structure of the process model to minimize the difference between the observables.

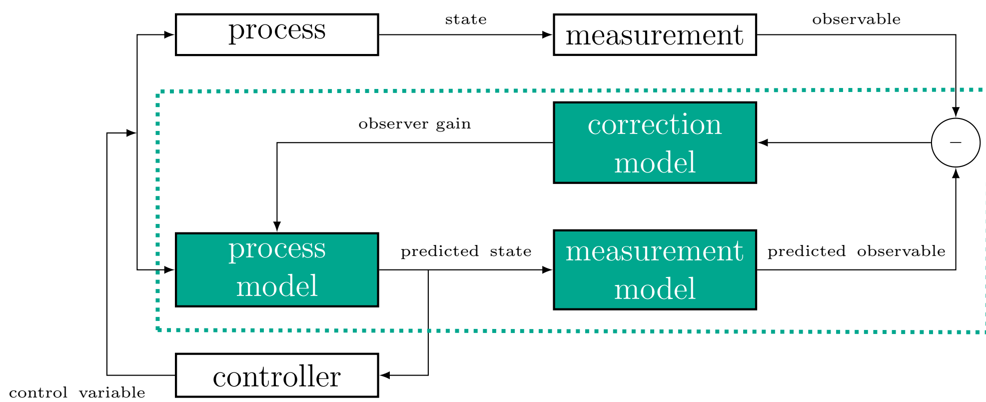

Figure 1: Observer in parallel to the process.

In this paper we take a look at dimensionality reduction methods to find a process model that is less complex than finite element or phase field models to make on-line state tracking feasible (section 2). A method to identify the process model is outlined in section 3. In section 4 we look at a deep drawing process to show first results of data reduction, feature extraction and dynamic process modelling.

2 Dimensionality reduction

In our case, data reduction has two goals: (1) to compare predicted and real observables and (2) to find the dynamic process model. Both are more complex and more time-consuming without dimensionality reduction. The data has a non-linear structure and many correlations. For the purpose of modelling the process and describing the state in a lower dimensional space the main requirements for the dimensionality reduction method are:

- feature extraction: few representative quantities of high-dimensional data,

- reconstruction of the original data from features and
- non-linearity.

Feature extraction attempts to find a subspace of the original data space which still describes the original data. One important property of the feature space is the orthogonality of the features; additionally, for our purposes the features have to be ordered such that the first feature has the highest information gain. The possibility to reproduce the original data from the extracted features is important for analysis of the reconstruction error. Thus, the main issue is minimizing the reconstruction error — the root mean squared error (RMSE):

$$E_{\mathrm{RMSE}} = \sqrt{\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}|X_{ij} - \hat{X}_{ij}|^2} \tag{1}$$

where m is the number of samples (numerical experiments with different values for a specific parameter) and n is the number of (state) variables. X indicates the real values and $\hat{X}$ the predicted ones. We apply the principal component analysis as a linear method and investigate non-linear principal component analysis [3] as a means of feature extraction.

2.1 Principal component analysis

Principle component analysis (PCA), also known as proper orthogonal decomposition (POD), is a widely used linear dimensionality reduction method that deals with the two issues of feature extraction and reconstruction of the original data. PCA starts from the premise that the highest variance in the data corresponds to the highest information gain and hence small variances constitute negligible information or even noise. Thus, PCA declares the highest variance to be the first feature, then searches in the orthogonal subspace for the next highest variance and so on. Hence, PCA extracts orthogonal, ordered features. Singular value decomposition is applied to run the PCA (see listing 1).

```
X = inputData;
m = numberOfSamples;
% compute covariance matrix
c = 1/m * (X'X);
% get eigenvectors and eigenvalues by singular value decomposition
[U,S,~]=svd(c);
% f vectors (features) of U such that a desired variance is covered
Ureduce = U(:,1:f);
% project the data to components
Z = X * Ureduce;
```

Listing 1: PCA using singular value decomposition (svd).

2.2 Non-linear principal component analysis

We consider non-linear principal component analysis (NLPCA) [3][4], which is based on an Autoencoder — a bottleneck neural network where the input has the same meaning as the output. There are three hidden layers: the mapping layer that maps the input data X to the features Z (figure 2: $X \to Z$), a bottleneck layer whose activation represents the feature space, and the demapping layer ($Z \to X$) that reconstruct the input data. Scholz [3] enhanced the NLPCA of Kramer [4] by using a hierarchical error function. Kramer does not take advantage of the property of linear PCA that features are sorted. Thus, it can only be used for dimensionality reduction and not for feature extraction, whereas Scholz' enhancement extracts non-linear features with obviously different variances; the extracted features are sorted as with linear PCA. The hierarchical error function E_h first calculates

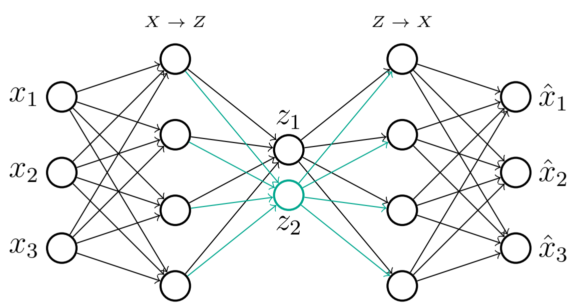

Figure 2: Architecture of NLPCA with two features.

the mean squared error E_1 between output X and predicted output $\hat{X}$ of the network using only the first bottleneck node (black lines in figure 2), then adds the error $E_{1,2}$ of the network taking the first and second node of the bottleneck layer into account (black and green lines in figure 2) and so on. This hierarchical error $E_h = E_1 + E_{1,2} + \ldots + E_{1,\ldots,k}$, where k is the number of bottleneck nodes and features, then has to be minimized. As a result we have the features in a hierarchical order.

3 PROCESS MODEL IDENTIFICATION

The reduced feature space representation is the starting point for the identification of the reduced dynamic model. Artificial neural networks are often used to predict parameters in manufacturing processes [5][6][7]. We instead propose symbolic regression with the goal to create a knowledge-based model. Symbolic regression [8][9] is a genetic algorithm that can estimate not only the parameters but also the model structure. It is possible to integrate expert knowledge about the model when learning a description of the feature space representation.

Symbolic regression deals with discrete mathematical formulas encoded in trees (figure 3). First, experts can make decisions about operands and general structure that will be used to find a solution that describes the feature space. Then, random symbolic equations are generated. Promising equations will be selected for further processing, like mutation, which can change a random node (e. g. from "+" to "·"), or crossover. Crossover selects

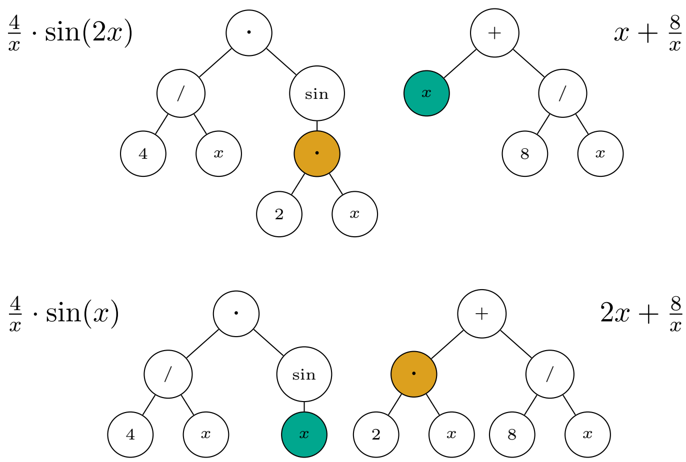

Figure 3: Tree architecture of symbolic regression equation and crossover.

two random nodes (green and yellow node in figure 3) of different trees and swaps them. The resulting new equations are again examined as to how they fit the data. This repeats until a satisfactory result is achieved, i. e. a result where the regression error is minimized and the mathematical formula has an acceptable length. The resulting reduced model of the process forms the core of the state observer (figure 1).

4 RESULTS

4.1 Deep drawing process data

The introduced dimensionality reduction methods and symbolic regression are applied to two simulation models of cup deep drawing of a metal sheet. Figure 4 shows the two finite element deep drawing models with their von Mises stress distribution when the workpiece is already formed by the punch. The finite element 2-dimensional model (figure 4 left) is axisymmetric and with isotropic plastic and elastic material behaviour. In contrast to this, the 3-dimensional model (figure 4 right) with anisotropic plastic and isotropic elastic material behaviour describes a more complex process [10]. For both models, statistical samples are generated by numerical experiments with different blank holder forces (time-independent). The state variables (time-dependent) describe the von Mises stress distribution in the workpiece during the deep drawing process.

The number of samples m, the number of state variables n and the number of time steps o have to be arranged into a 2-dimensional matrix for further processing. Regarding the dynamic process we aim for a time-dependent reduced state representation. Hence,

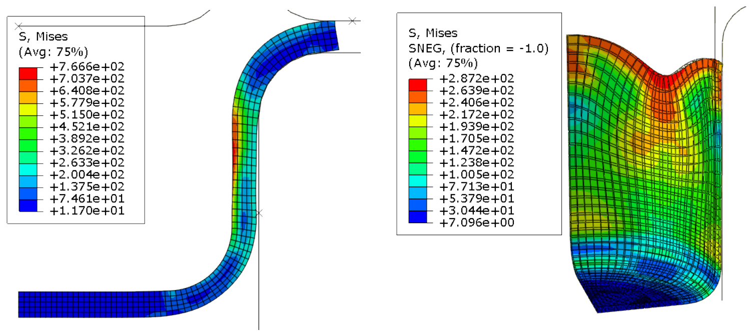

Figure 4: Workpiece during deep drawing. Left: elementary sample process. Right: complex sample process.

the time steps are unrolled as samples so that the resulting 2-dimensional data matrix is of size $(m \cdot o) \times n$:

$$\begin{matrix} x_{11}(t_1) & x_{12}(t_1) & \dots & x_{1n}(t_1) \\ x_{11}(t_2) & x_{12}(t_2) & \dots & x_{1n}(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_{11}(t_o) & x_{12}(t_o) & \dots & x_{1n}(t_o) \\ x_{21}(t_1) & x_{22}(t_1) & \dots & x_{2n}(t_1) \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1}(t_o) & x_{m2}(t_o) & \dots & x_{mn}(t_o) \end{matrix} \tag{2}$$

As a consequence, a state variable x of a discrete time step t_i $(0 \leq i \leq T)$ is described as $x_{sv}(t_i)$, where s is a specific sample and v is the index of the state variable.

4.2 Dimensionality reduction

Using the data structure of state variables, samples and time of section 4.1, the dimensionality reduction methods in section 2 have been applied. The first three features resulting from PCA and NLPCA for the samples with the highest and lowest blank holder forces are shown as a function of time in figures 5 and 6. So far there are three obvious insights to be gained from these diagrams. First, the effect of the blank holder force is most significant near the end of the deep drawing. Second, the projection of the features of the 3-dimensional model shows a peak (at time 0.86 s) when the sheet slides out of the clamp between die and blank holder. And third, the first feature shows similarities to a stress-strain curve — especially for the 3-dimensional model the distinct yield strength is obvious.

The RMSE as reconstruction error is used to verify the quality of the extracted features as a measure of how well the original data can be reproduced from the features. Table 1 shows the results of applying PCA and NLPCA to the 2-dimensional and 3-dimensional models. The more features are extracted the more accurately the original data can be

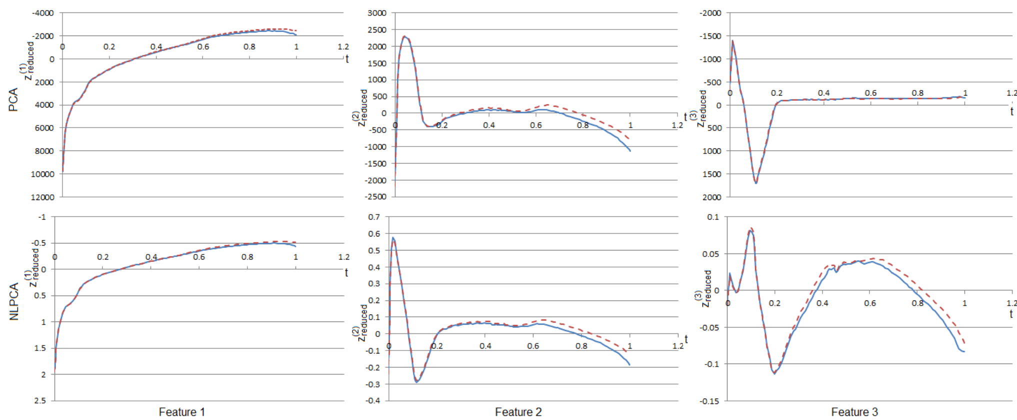

Figure 5: First three features of the 2-dimensional model as functions of time; time unit in seconds. Comparison of PCA (top row) and NLPCA (bottom row). The blue line is the experiment with the lowest blank holder force (70kN) and the red dotted line is the experiment with the highest blank holder force (100kN).

Table 1: Reconstruction error (RMSE) for 2-dimensional and 3-dimensional model with PCA and NLPCA in MPa.

# of features	2-d model		3-d model	
	PCA	NLPCA	PCA	NLPCA
1	52.5753	44.1448	33.6534	25.4876
3	29.7434	28.2493	11.5975	14.6937
5	19.2415	25.2808	7.8628	12.7749

reproduced. The features resulting from NLPCA, in comparison to the features resulting from PCA, are a more accurate representation of the original data when using three or less features considering the smaller RMSE. It is hypothesized that the first feature represents the majority of the non-linearity in the data.

4.3 Process model identification

Model identification is a method to describe time-series behaviour. Thus, it can be assumed that the highest information gain (first feature) regarding the state variables (von Mises stress) of a deep drawing process is in the deformation of the metal sheet, because it causes the highest changes in the stress distribution over time. The stress-strain curve of the 2-dimensional model (section 4.1) is described as

$$\sigma(\epsilon) = \sigma_0 + \Theta\epsilon^m, \tag{3}$$

where σ_0 is the starting flow stress, Θ is a strength coefficient, ϵ is the real strain and $m = 0.45$ is the strain hardening exponent. Based on the description $feature_1 = f(t)$ a symbolic regression has been applied. Assuming that the result could be similar to a

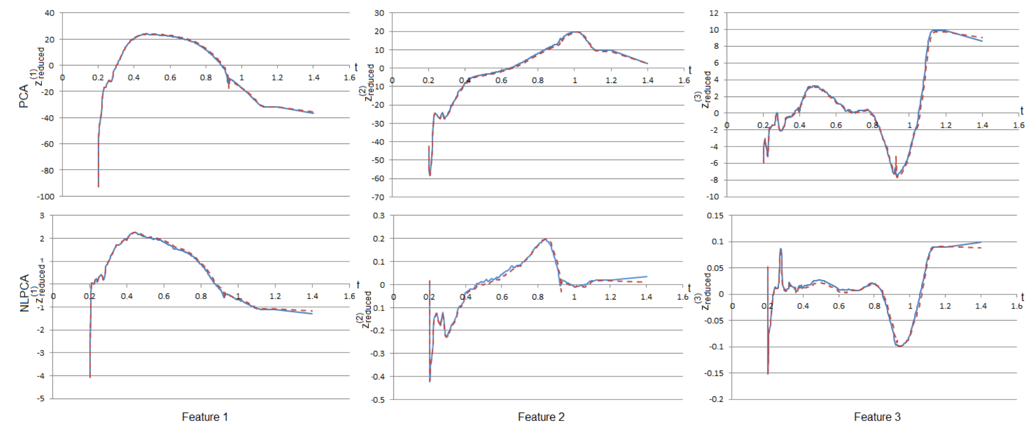

Figure 6: First three features of the 3-dimensional model as functions of time; time unit in seconds. Comparison of PCA (top row) and NLPCA (bottom row). The blue line is the experiment with the lowest blank holder force (16kN) and the red dotted line is the experiment with the highest blank holder force (24kN).

stress-strain curve, operators like $+$, $-$, $\cdot$, $/$, $\hat{}$, as well as constant numbers are used to find an estimated function. With this knowledge and applied to the first feature of the NLPCA the regression finds $f(t) = -3.49 + 4.08t^{0.123}$. Under the same conditions, the symbolic regression finds for the 3-dimensional model: $f(t) = 77t + 58.1t^2 - 4.06^t - 9.86\sqrt{t} - 127\sqrt{t^3}$.

5 CONCLUSION AND FUTURE WORK

In this paper state tracking based on a Luenberger observer and using symbolic regression to identify the process model is suggested. In this context dimensionality reduction methods are applied to reduce data (state variables) of simulation outputs. The result of the dimensionality reduction is a lower-dimensional representation of the state in a feature space. It is shown that a few features can represent the state variables with a negligible error. Based on this lower-dimensional representation the symbolic regression is applied. It is assumed that this knowledge-based modelling method will led to interpretable results to predict material characteristics. The feasibility of this idea is tested on deep drawing models.

Whereas the first feature appears to be the influence of strain on the stress (time-variant), the features two and three are still open for discussion. One theory is that they represent the changes in the stress distribution in certain regions of the workpiece. Furthermore, it is still to be clarified what impact the algebraic sign of each feature has — could it states the mechanical strain or compression? Future work will also attempt to learn differential equations.

ACKNOWLEDGEMENT

The funding of this project within the Research Training Group 1483 "Process Chains in Manufacturing" by the German Research Foundation (DFG) is gratefully acknowledged. We are also very grateful to Prof. Dr. Alexander Wanner for invaluable discussions and support, and the ITM at the KIT for providing the finite element deep drawing models.

REFERENCES

[1] M. Senn and N. Link, "Hidden state observation for adaptive process controls," in *Proceedings of the Second International Conference on Adaptive and Self-adaptive Systems and Applications (ADAPTIVE 2010)*, pp. 52–57, IARIA, 2010.

[2] D. G. Luenberger, "Observing the state of a linear system," *IEEE transactions on military electronics*, vol. 8, no. 2, pp. 74–80, 1964.

[3] M. Scholz and R. Vigrio, "Nonlinear pca: a new hierarchical approach," in *ESANN* (M. Verleysen, ed.), pp. 439–444, 2002.

[4] M. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE Journal*, vol. 37, pp. 233–243, 1991.

[5] J. Zhao and F. Wang, "Parameter identification by neural network for intelligent deep drawing of axisymmetric workpieces," *Journal of Materials Processing Technology*, vol. 166, no. 3, pp. 387–391, 2005.

[6] S. K. Singh and D. R. Kumar, "Application of a neural network to predict thickness strains and finite element simulation of hydro-mechanical deep drawing," *The International Journal of Advanced Manufacturing Technology*, vol. 25, no. 1–2, pp. 101–107, 2005.

[7] A. Brahme, M. Winning, and D. Raabe, "Prediction of cold rolling texture of steels using an artificial neural network," *Computational Materials Science*, vol. 46, no. 4, pp. 800–804, 2009.

[8] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

[9] M. Schmidt and H. Lipson, "Comparison of tree and graph encodings as a function of problem complexity," in *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation (GECCO '07)*, 2007.

[10] N. Kishor and D. Kummar, "Optimization of initial blank shape to minimize earing in deep drawing using finite element method," *Journal of Materials Processing Technology*, vol. 130–131, pp. 20–30, 2002.