

Process Identification through Modular Neural Networks and Rule Extraction

B.J. van der Zwaag^a C.H. Slump^a L. Spaanenburg^b

^a University of Twente, P.O.Box 217, 7500 AE Enschede (NL)

^b Groningen University, P.O.Box 800, 9700 AV Groningen (NL)

Abstract

Monolithic neural networks may be trained from measured data to establish knowledge about the process. Unfortunately, this knowledge is not guaranteed to be found and – if at all – hard to extract. Modular neural networks are better suited for this purpose. Domain-ordered by topology, rule extraction is performed module by module. This has all the benefits of a divide-and-conquer method and opens the way to structured design. This paper discusses a next step in this direction by illustrating the potential of base functions to design the neural model.

[Full paper published as: Berend Jan van der Zwaag, Kees Slump, and Lambert Spaanenburg. Process identification through modular neural networks and rule extraction. In *Proceedings FLINS-2002*, Ghent, Belgium, 16-18 Sept. 2002.]

1 Introduction

Process Identification is a major part of Control Theory, where the (partly) unknown process must be monitored and modeled before it can be controlled. This has been pursued through numerous mathematical schemes, fuzzy logic, and neural networks.

This paper sets out to explore the possibilities to provide in-line assistance to the process modeler. Neural networks are used to capture the process knowledge in a fusion of existing knowledge and measured data. Such networks are modular to guarantee learnability and to allow for knowledge re-use. Using a large number of small nets instead of a single one suppresses the combinatorial explosion of the search space from which the rule extraction on monolithic neural networks suffers. This can be further supported by the introduction of base functions: domain-specific pieces of common truth that help to construct the model by reading from a heterogeneous network with frozen parts.

2 Modular Neural Networks

Modular networks typically consist of a network of interconnected neural networks (also called modules or rule blocks) that each solve a sub-problem [1]. The major benefit of modular networks follows from the inherent *separation of concerns*. The disreputed failure to learn in monolithic neural nets is caused by the presence of conflicting features, where the feed-forward arrangement tends to compromise instead of to select. Separating such conflicts and solving them in separate modules before assembling the overall network solves such problems. This can be based on a natural partition of the problem space reflecting the domain knowledge.

3 Knowledge Extraction and Domain Development

Neural network behavior described in sets of rules can provide insight into how the network comes to an answer. However, there are problem domains where solutions cannot easily or comprehensively be described in sets of rules or decision trees, due to, e.g., high dimensionality of the input space or very large sets of independent features. A remedy is to identify basic functions with which users in a given domain are already familiar, and to describe trained networks, or parts thereof, in terms of those base functions. This will provide a comprehensible description of the neural net's function and may also provide an insight into its inner "reasoning". Table 1 gives some potential base functions.

The proper choice of base functions allows to introduce existing knowledge rather than to start from blind learning. In a 3-tier concept, the micro layer uses existing molecular formulae as base functions in a neural setting; the application model is a regular neural network that reflects the itinerary through the phase diagram; and the middle layer glues these two together and is based on conventional physical principles. This model has been suggested for the case of a metal scrap furnace, where the exhaust fumes are measured in controlling the process to optimize the industrial product while minimizing the environmental damage. The inability to measure directly on the process makes it necessary to bring all available information into one model and to enhance it from experience.

The methodology will be most attractive for the characterization of production processes, where the nature and location of the process make a direct measurement of process variables difficult, if not impossible. In such cases, sensors like the camera or the audiometer can be used to measure non-electrical process parameters and carry them into the input domain where a neural net can operate.

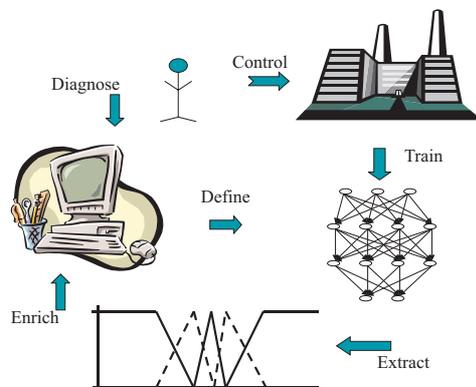


Figure 1: The knowledge-in-the-loop architecture is based on the iterative improvement of the process model through the extraction of knowledge within the (post-)training neural network.

Table 1: Some problem domains where neural networks have been successfully applied. Possible domain-specific base functions are presented for each of these domains.

application domain	potential base functions
signal processing	basic operational filters
digital image processing	differential operators
general classification problems	feature map regions (cp. Kohonen SOFM)
decision theory	if-then rules (i.e., classical rule extraction)
control theory	basic control operators

References

- [1] L. Spaanenburg, W.J. Jansen, and J.A.G. Nijhuis. Over multiple rule-blocks to modular nets. In *Proceedings Euromicro '97*, pp. 698-705, 1997.