Neural Networks and Evolutionary Algorithms for the prediction of thermodynamic properties for chemical engineering

Martin Mandischer[1] and Hannes Geyer[2] and Peter Ulbig[2]

[1] Department of Computer Science XI, University of Dortmund, Germany
[2] Institute for Thermodynamics, University of Dortmund, Germany

Abstract. In this paper[1] we report results for the prediction of thermodynamic properties based on neural networks, evolutionary algorithms and a combination of them. We compare backpropagation trained networks and evolution strategy trained networks with two physical models. Experimental data for the enthalpy of vaporization were taken from the literature in our investigation. The input information for both neural network and physical models consists of parameters describing the molecular structure of the molecules and the temperature. The results show the good ability of the neural networks to correlate and to predict the thermodynamic property. We also conclude that backpropagation training outperforms evolutionary training as well as simple hybrid training.

Keywords: Neural Networks, Evolution Strategies, Hybrid-Learning, Chemical Engineering

1 Introduction

In chemical engineering the simulation of chemical plants is an important task. Millions of chemical compounds are known yet and experimental data are often not available. For this reason there is a need for calculation methods which are able to predict thermodynamic properties. Usually models are developed, which have a physical background and where the model parameters have to be fitted with the aid of experimental data. This leads usually to nonlinear regression models with a multi-modal objective function where evolution strategies are successfully used. In contrast to models with physical background simple so-called incremental methods are widely used, too. Each functional group of a molecule gives a contribution to the thermodynamic property and the sum of all contributions have to be calculated. A new way for the calculation and prediction of thermodynamic properties is the use of neural networks. Descriptors, which can be derived from the molecular structure, have to be defined for the input layer. Then experimental data for a specific thermodynamic property can be used for

[1] **Acknowledgments:** The work presented is a result of the *Collaborative Research Center SFB 531* sponsored by the Deutsche Forschungsgemeinschaft (DFG)

training. Predictions of this thermodynamic property are then possible by using the molecular structure for a chemical compound, where no experimental data are available. In this investigation the enthalpy of vaporization was taken. In section 2 we give a brief overview of the models used and continue in section 3 with an experimental comparison of physical models, networks trained with backpropagation, networks trained with evolutionary algorithms and a combination of the latter two.

2 Models for the enthalpy of vaporization

2.1 Physical models

The physical background for the enthalpy of vaporization ΔH_v consists of electrostatic interactions forced by the atoms of the molecules. Equations can be derived from statistical thermodynamics in order to describe the interactions between molecules (first level) and between functional groups of these molecules (second level). Physical models, such as UNIVAP[2] (UNIversal enthalpies of VAPorization) which summarizes the interactions between functional groups of the molecules within a pure liquid were developed [8]. This model consists of sums of exponential terms, which include the interaction parameters and the temperature. The interactions are weighed by the surface fractions of functional groups of a molecule. The interaction parameters can be fitted to experimental enthalpy of vaporization data. This leads to a non-linear regression problem with a multi modal objective function. This function consists of the mean absolute error (MAE) over all experimental data points N between the calculated values (physical model) and the experimental data: $MAE = \frac{1}{N} \sum_{N} \left| \Delta H_v^{calc.} - \Delta H_v^{exp.} \right|$

Due to the complex structure of the physical model, especially the exponential terms, multi-modality occurs. An evolution strategy for solving this problem was developed [3,8,9]. Another theoretical approach is the so-called EBGVAP model (Enthalpy Based Group contribution model for enthalpies of VAPorization) which was used in our investigation, too.

2.2 Neural Networks

Neural networks are able to acquire an internal model of a process by learning from examples. After successful training the network will be a model for the process which led to the experimental data. Theoretical results show that feed-forward networks are capable of arbitrary exact function approximation, given an unlimited number of free parameters or infinite precision [4].

In our experiments we used simple feed-forward networks with non-linear sigmoid activation functions. As training algorithms we employed the standard Backpropagation algorithm [6] and various (μ,λ) evolution strategies [7, 1] as well as a combination of both.

[2] For the UNIVAP model it was difficult to reach the critical point, where the enthalpy of vaporization reaches null. A modified temperature dependence was used in this investigation to improve the performance at critical points (UNIVAP2).

3 Experiments and Results

Comparing different methods or models is at least two-fold. On the one hand a fair comparison should allow all models the same number of free parameters to adjust to the problem. On the other hand, one can say that it is sufficient if a model performs good on formerly unseen data regardless of the number of parameters it needed.

Some of our experiments were designed to find good neural models under the most similar conditions for the calculations as the physical models. Here the number of adjustable parameters was almost the same for all models. In other experiments we searched for good results independent of the number of free parameters (weights) used. One difficulty is to find the optimal structure of the neural network and the optimal structure of the temperature dependent equation of the physical model. Here we only investigated the structure of the network. Another important issue is to have the same input information for all methods, which can be derived from the structure of the molecules.

3.1 Generation and Description of the Data

Selection of data. The experimental data concerning the enthalpy of vaporization were taken from data handbooks. Data for three different classes of chemical compounds were used: normal alkanes, 1-alcohols, and branched alcohols. These data were chosen for the investigation of three different functional groups, the so-called main groups: CH_3, CH_2 and CH_nOH. The group CH_nOH contains the functional groups CH_3OH, CH_2OH and CHOH. The experimental data cover a temperature range from 92 K to 776 K. The number of carbon atoms in the n-alkanes goes from 2 (Ethane) to 19 (Nonadecane), for the 1-alcohols from 1 (Methanol) to 14 (Tetradecanol) and for the branched alcohols from 4 (2-Methyl-2-propanol) to 6 (2-Methyl-2-pentanol).

Selection of descriptors. There are several possibilities for the definition of descriptors as input variables for a neural network: number of atoms, number of single bonds, molar mass, dipole moment and topological parameters concerning the connectivity between atoms [2]. In our investigation the descriptors for the input layer are the surface fractions of the functional groups within a molecule and the temperature. Therefore a definition of functional groups is necessary. Here the definition of the UNIVAP model [8] shall be used.

Group interaction	parameters	data points (total)	data points (training)
CH_3CH_3	3	110	53 (48.18 %)
CH_3CH_2 / CH_2CH_2	9	138	75 (54.35 %)
CH_nOHCH_nOH	3	19	10 (52.36 %)
CH_3CH_nOH / CH_2CH_nOH	12	162	76 (46.91 %)
total:	27	429	214 (49.88 %)

Table 1. Number of experimental data for the different group interactions

Partitioning into subsets for cross validation. After generating the data set it was subdivided into 3 classes: training (50%), validation (25%) and test (25%) set. The training set was used to adapt the parameters for all our models. The validation set could be used during the adaption process to evaluate the algorithms performance on unknown data and stop the adaption process if the error on the validation set increases. Validation and test set therefore measure the generalization ability of our models. However, 50% of the data were used only for comparison, i. e. for a test of the prediction of the enthalpy of vaporization. The distribution of the data can be seen in Table 1.

Transformation. For the use with the neural network the data were normalized via separate linear transformations of main-groups, temperature and enthalpy to the interval [0.1 .. 0.9]. Network responses outside of this interval were mapped onto the boundaries and then re-transformed to the original scale.

3.2 Physical Model Experiments

The training set was used for the regression of the interaction parameters and the training of the neural network. First the parameters were computed successive, i. e. first the 3 parameters for the interaction CH_3/CH_3 were calculated and with these parameters the next 9 parameters (corresponding to Table 1) were calculated and so on. These sequential experiments for the physical models were done with the aid of an encapsulated evolution strategy without a correlated step-length control [3]: $\left[GG\ 3 + 8(GG\ 7 + 19)^{200}\right]^{40}$

Three different runs with each $1.2 \cdot 10^6$ function calls of the evolution strategies gave similar results. These results were optimized by the simplex-method with 30 different runs of 1000 iterations each. The best result for UNIVAP2 (seq) and EBGVAP (seq) can be found in Table 2. In contrast to this sequential regression of the model parameters a simultaneous regression (sim) of all 27 parameters was investigated by using a similar encapsulated ES as for the sequential experiments. The results of these runs were improved by a simplex-method, too and can be seen in Table 2.

3.3 Neural Networks Experiments (Backpropagation)

The learning rate η and the architecture of the network (number of hidden units and connections) have the biggest influence on the performance of the network [5]. To find good neural network solutions we did a primitive parameter study. We first varied the learning rate with a fixed architecture which had approximately the same number of free parameters (connections) as the UNIVAP methods. With the best learning rate found, we searched for a good number of hidden units. All runs were performed 10 times.

Variation of the learning rate. We fixed the architecture of the network at 4 input, 4 hidden and 1 output units (4-4-1) to have approximately the same number of free parameters (25) as the UNIVAP method. We started with a very low learning rate $\eta = 0.001$ and ended with a far too high rate $\eta = 10.0$. The

momentum term α was fixed to 0.2. A training run was stopped after it reached the error limit or exceeded a maximum number of 100,000 pattern presentations (epochs). The error is defined as: $tss = \frac{1}{2} \times \sum_{i=1}^{n} (\theta_i - o_i)^2$. With θ as target vector and o as output activation of the network.

Figure 1 and 2 show the curves for 10 different runs with the best learning rate which was used throughout all other experiments. The left-hand side figure gives the error on the training set and on the right-hand side we see the validation error. If an error curve reaches the base of the graph it satisfied a specified error limit ($tss \leq 5 \times 10^{-5}$) for the whole training set. Networks with very low learning rate never reached the specified error limit, due to the very slow learning progress. A too high rate, resulted in oscillating error curves.

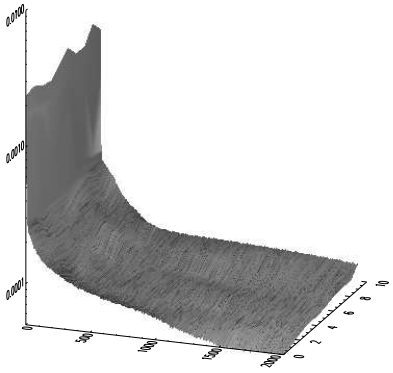

Fig. 1. Training error (η=0.8)

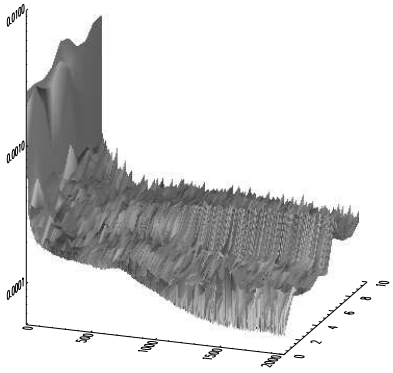

Fig. 2. Validation error (η=0.8)

Variation of the number of hidden units. After variation of η we used the best rate, as constant for the hidden unit search[3]. The number of hidden units were varied between 1 and 40. Networks with less then 3 units failed to learn the task. Up to 40 units the results on training as well as validation data were almost independent of the number units employed. We therefore used our initial 4-4-1 network. This is an additional advantage because it can now be directly compared to other methods which use the same number of free parameters.

3.4 Neural Networks Experiments (Evolution Strategy)

In this experiment we substituted the Backpropagation algorithm with an evolution strategy. Some authors [10] reported good results when training a network with an ES. Again we systematically searched for a good parametrisation of the (15,100)–ES. Parameters under consideration were the number of mutation step-sizes σ_i and the recombination scheme used on the object variables x_i

[3] This does not mean that both parameter are independent of each other. We consider this value to be a first estimate to start with.

(the network weights). Each parameter setting was run for 10,000 generations (1,000,000 pattern presentations) and repeated 10 times to have some statistical validity. All of the following variations of the bisexual recombination scheme were done with 1 and 25 σ: no recombination of x_i and σ_i, discrete recombination of x_i and discrete of σ_i discrete recombination of x_i and intermediate of σ_i intermediate recombination of x_i and discrete of σ_i intermediate recombination of x_i and intermediate of σ_i. For details on ES see [1, 7].

None of the parameter settings lead to good and reliable results. Only one out of all ES trained network performed comparable to Backpropagation. All other networks give rather poor results. The quality of the average result did improve when using backpropagation as local search procedure (an additional training of 250,000 epochs) after ES optimization but was not as good as Backpropagation alone. Figure 3 shows the best run, which we regard as a very rare event, with a (15,100)–ES.

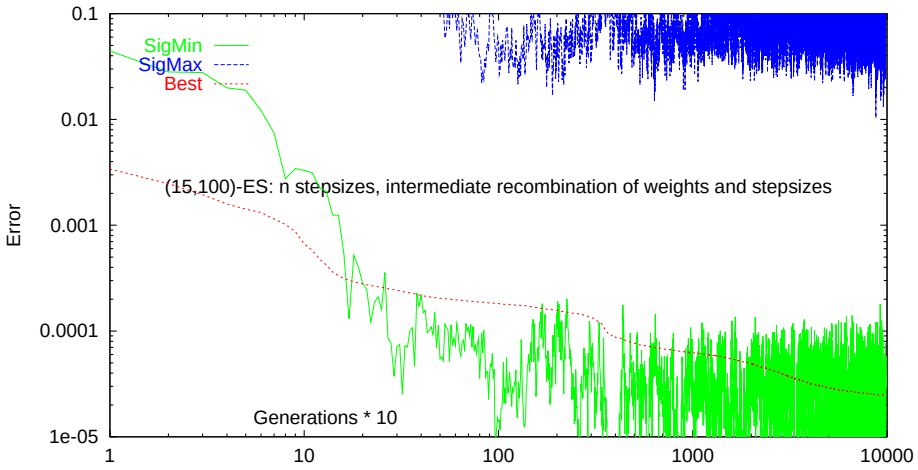

Fig. 3. (15,100)–ES (error during training)

3.5 Comparison

For a comparison, we took two network architectures with learning rates gained by the previous experiments. Architecture A has 4 hidden units and the nearly the same number of free parameters (25 weights) as the UNIVAP models (27). Architecture B performs alike and has 6 hidden units (37 weights). Table 2 gives on overview of all experiments.

1. Parameters for NN-A (4-4-1): η=0.8, epochs=250,000
2. Parameters for NN-B (4-6-1): η=0.8, epochs=250,000
3. Parameters for NN-ES (4-4-1): best (15,100)–ES $\#\sigma = n$, intermediate recombination of x_i and σ_i (100,000 generations)
4. Parameters for NN-ES (4-4-1): average (15,100)–ES $\#\sigma = n$, intermediate recombination of x_i and σ_i (100,000 generations)

	UNIVAP 2 (seq)	EBGVAP (seq)	UNIVAP 2 (sim)	EBGVAP (sim)
Train	0.914	0.635	1.128	3.784
Valid	1.190	0.867	1.353	4.436
Test	0.948	0.613	1.209	3.863
All	1.017	0.705	1.230	4.028
	NN-A	NN-B	NN-ES (best)	NN-ES (avrg)
Train	0.652	0.570	0.612	1.143
Valid	0.566	0.878	0.876	1.536
Test	0.686	0.703	0.747	1.357
All	0.635	0.717	0.745	1.345

Table 2. Mean absolut error per pattern for different data sets and models

As an additional test for generalisation ability, we used all data of an ethane molecule. In figures 4 and 5 we compare all models on the enthalpy prediction for ethane. It can be seen that the physical model and the neural network performs equally well on this task, except for the critical regions near $T \to T_{cr}$ and $\Delta H_v(T_{cr}) = 0$ J/mol, where the network outperforms all other models. Almost all networks trained with an ES and the UNIVAP model give only a poor linear approximation of the enthalpy curve.

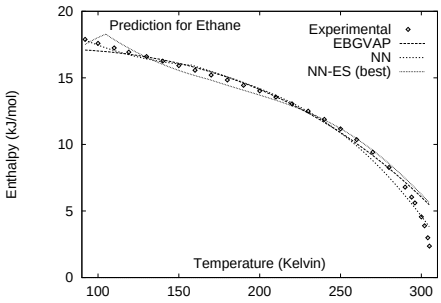

Fig. 4. Performance on ethane (good)

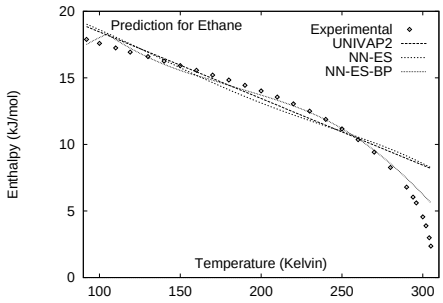

Fig. 5. Performance on ethane (poor)

4 Discussion

The most important result of this investigation is the good ability to correlate as well as to predict the enthalpy of vaporization with neural and physical methods. Neural networks with simple Backpropagation training are as good as physical models and especially at critical temperatures even slightly better, but their computational effort is much lower. The comparison of the results for UNIVAP2 and EBGVAP show the influence of the structure of the model itself. Further investigations could use evolutionary algorithm to optimize the structure of the models with regard to the temperature dependence. For the neural networks it can be stated that the use of surface fractions of functional groups as descriptors

for a neural network leads to good results for both correlation and prediction. The big advantage of this new procedure is, that the molecules can easily be divided into functional groups, which makes it easy to use in engineering applications and allows the direct comparison of neural networks and physical models, due to the same input information. The investigations concerning the architecture of the neural networks show, that a simple network structure is sufficient and a more complicated network does not give better results. In this context evolution strategies as training algorithm and combinations of ES with backpropagation failed to deliver useful models in almost all experiments.

From a thermodynamic point of view, it is interesting that a simple method like a neural network can give similar results in comparison with much more complicated physical motivated models. If a physical model gives results with a quality less than a neural model, the physical model should be improved. However, in chemical engineering there are many thermophysical properties, which are usually not described by physical methods, but by incremental methods. These methods, for example, for critical data, normal boiling points and so on, could be replaced by neural networks. However, these results are first steps in developping efficient network structures for our purpose and especially investigations with more functional groups will give a better comparison between physical models and neural networks.

References

1. T. Bäck. *Evolutionary Algorithms in Theory and Practice.* Oxford Univ. Press, New York, 1996.
2. L. M. Egolf and P. Jurs. Prediction of boiling points of organic heterocyclic compounds using regression and neural network techniques. In *J. Chem. Inf. Comput. Sci. 33*, pages 616–625. 1993.
3. H. Geyer, P. Ulbig, and S. Schulz. Encapsulated evolution strategies for the determination of group contribution parameters in order to predict thermodynamic properties. In *5th Int'l. Conf. on Parallel Problem Solving from Nature.* Amsterdam, 1998.
4. K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.
5. M. Mandischer. Evolving recurrent neural networks with non-binary encoding. In *Proc. Second IEEE Int'l Conf. Evolutionary Computation (ICEC '95), vol. 2*, pages 584–589, Perth, Australia, 1995. IEEE Press, Piscataway NJ.
6. D. E. Rummelhart and J. L. McClelland. *PDP: Explorations in the Microstructure of Cognition*, volume 1. MIT Press, Cambridge, MA, USA, 1986.
7. H.-P. Schwefel. *Evolution and Optimum Seeking.* Sixth-Generation Computer Technology. Wiley, New York, 1995.
8. P. Ulbig. *Gruppenbeitragsmodelle UNIVAP & EBGCM.* Dr.-Ing. Thesis, Univ. of Dortmund, Institute for Thermodynamics, 1996.
9. P. Ulbig, T. Friese, H. Geyer, C. Kracht, and S. Schulz. Prediction of thermodynamic properties for chemical engineering with the aid of Computational Intelligence. In *Progress in Connectionist-Based Information Systems.* Springer, 1997.
10. W. Wienholt. Minimizing the system error in feedforward neural networks with evolution strategy. In S. Gielen and B. Kappen, editors, *Proc. of the Int'l. Conf. on Artificial Neural Networks*, pages 490–493, London, 1993. Springer-Verlag.