# Dimensionally Homogeneous Neural Networks for System Identification

Dipl.-Ing. Steffen Brückner and Dr. Stephan Rudolph
Institute for Statics and Dynamics
of Aerospace Structures (ISD)
University of Stuttgart
Pfaffenwaldring 27
D-70550 Stuttgart
*[brueckner | rudolph]@isd.uni-stuttgart.de*
*http://www.isd.uni-stuttgart.de/pigroup/*

## Abstract

Dimensionally homogeneous neural networks [10, 11] have been proven to possess several important advantages over dimensional inhomogeneous neural networks. The principal idea of combining artificial intelligence techniques with the powerful concept of dimensional analysis opens new perspectives and possibilities in the identification of adaptive systems and in intelligent control. Some issues in the application of dimensional analysis and neural networks in system identification is investigated and discussed in this paper.

## 1 Introduction

Using the so-called Pi-transform (as shown later in eq. (6)), problems involving $n$ quantities in the physical domain can be reformulated involving $m$ ($m < n$) dimensionless groups of the original $n$ quantities. This transform is commonly used to simplify subsequent modelling and analysis efforts (see Bridgman [2]).

In the control of smart structures new control designs, such as nonlinear control and neural network control, have to be used due to the highly nonlinear nature of these control problems. Neural network controllers are often implemented as indirect controllers, meaning that they rely on an identified (neural network) model of the plant.

This neural network model identification technique using dimensional homogeneous neural networks is discussed in this paper.

## 1.1 Controlled systems

For most neural control problems the modelling and identification of the controlled system plays an important role. These dynamical systems are categorized in linear systems, which obey the rules of superposition, and non-linear systems, where such principles do not hold. Classical system and control theory are based on linear time-invariant systems. New approaches to system modelling include nonlinear time-varying systems [8].

## 1.2 State-space notation

The state-space notation provides a standard form of the equation of motion for dynamical systems using the state vector $\mathbf{x}$ as coordinates. The number of states equals the number of degrees-of-freedom of the system. Especially the matrix notation (eq. 1 and 2) is often used in modern dynamic system theory and control system design and analysis.

In state-space notation dynamical systems are written as first-order differential equations. Dynamic systems with linear equations of motion of second or higher order can be transformed into an equivalent set of linear first-order differential equations and therefore all linear systems can be written in state-space notation. State-space notations are explicit notations, systems that can only be written using implicit formulations cannot be transformed into state-space notation.

Linear time-invariant (LTI) continuous time systems can be described using a matrix state-space notation

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y} &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \end{aligned} \tag{1}$$

and discrete time LTI systems as

$$\begin{aligned} \mathbf{x}(t+\Delta t) &= \mathbf{A}_z\mathbf{x}(t) + \mathbf{B}_z\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}_z\mathbf{x}(t) + \mathbf{D}_z\mathbf{u}(t) \end{aligned} \tag{2}$$

Non-linear time-variant systems can also be written in state-space notation as

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{u}, t) \\ \mathbf{y} &= g(\mathbf{x}, \mathbf{u}, t) \end{aligned} \tag{3}$$

with the nonlinear functions $f$ and $g$ which usually include derivatives of the states $x_i$ and inputs $u_i$.

# 2 Dimensional Analysis

A function relating measured quantities is called a ’complete’ equation according to Bridgman [2] if it is of a form that it remains formally true without any change in the form of the function whenever the size of the fundamental units is changed in any way whatever. Every adequate and correct expression can be made ’complete’ by means of dimensional constants. Dimensional analysis only applies to to ’complete’ equations (which are called dimensionally homogeneous equations), where the numerical size of the fundamental units can be changed, but their physical statement remains unaffected.

Dimensional analysis helps to establish the form of relationships governing physical phenomena that are too complicated to be obtained by regular mathematical methods. Also, dimensional analysis helps to find the scaling-laws for a problem which helps to reduce the number of parameters involved in the problem and therefore reduces the experimental work needed. This also simplifies graphical and tabular representations.

## 2.1 Similarity

The dimensionless groups found by dimensional analysis allow the scaling of the process as long as the governing equation or at least the relevance list holds. For the scale-up or scale-down of processes, the following types of similarities have to be fulfilled [2]:

- geometric similarity: *ratios of lengths are equal*
  Two bodies are geometrically similar if the, with sufficient enlargement, the smaller can be brought to exact coincidence with the larger.
- kinematic similarity: *ratios of speeds are equal*
  Two moving systems are kinematically similar if homologous points experience the same motion in homologous times.
- dynamic similarity: *ratios of forces are equal*
  Two bodies are dynamically similar if their homologous points experience the homologous forces in homologous times.
- thermal similarity: *ratios of temperatures are equal*
  Two bodies are thermally similar if at their homologous points they have homologous temperature at homologous times.

If for two systems all dimensionless groups formed by the relevant physical variables are equal, then the two systems are considered *analogous*. All measurements and simulations of these two systems are closely correlated, in fact, the description of both systems in terms of the dimensionless variables is identical. This property allows the scaling of model experiments to the real-world implementations, e.g. wind channel testing of airplanes and large buildings using scale models.

In fluid dynamics some force ratios (dynamic similarity numbers) have been named honoring various researchers in the field of fluid dynamics. These numbers (e.g. REYNOLDS and FROUDE number) are very easily interpreted as the ratio of physically relevant forces.

## 2.2 The Pi-Theorem

**Pi-Theorem [3, 2]** *From the existence of a dimensionally homogeneous and complete equation $f$ of $n$ physical quantities $x_i$, the existence of an equation $F$ of only $m$ ($m = n - r$) dimensionless [1] quantities $\pi_j$ can be shown*

$$\begin{aligned} f(x_1, \ldots, x_n) &= 0 && (4) \\ F(\pi_1, \ldots, \pi_m) &= 0 && (5) \end{aligned}$$

*where $r = m - n$ is the rank of the dimensional matrix constructed by the $x_i$ and with dimensionless quantities $\pi_j$ of the form*

$$\pi_j = x_j \prod_{i=1}^{r} x_i^{-\alpha_{ji}} \quad (6)$$

*with $j = 1, \ldots, m \in \mathbb{N}$ and $\alpha_{ji} \in \mathbb{R}$ as constants.*

Having formulated a relevance list of a physical problem, Buckingham's Pi-theorem [3] shows a straightforward way to determine the dimensionless groups involved in the problem.

## 2.3 The structure of physical equations

Most physical phenomena can be written in an explicit formulation such as $p = f(x, y, z)$.

$$\begin{aligned} p_1 &= f(x_1, y_1, z_1) \\ p_2 &= f(x_2, y_2, z_2) && (7) \end{aligned}$$

[1]The symbol $\pi$ for the dimensionless variables is not to be confused with the also dimensionless ratio of circumference to diameter of a circle $\pi = 3.1415926\ldots$, which by itself is only one of many possible dimensionless groups

Now the unit used to measure $x$ is reduced by a factor of $1/a$, the measure of $x$ increases to $ax$. The units used to measure $y$ and $z$ are also reduced by a factor of $1/b$ and $1/c$ respectively. The above equations change to

$$\begin{aligned} p_1 &= f(ax_1, by_1, cz_1) \\ p_2 &= f(ax_2, by_2, cz_2) \end{aligned} \tag{8}$$

Stating that physical laws should not be changed by a change in the unit of measurement, it is expected that

$$\frac{f(x_1, y_1, z_1)}{f(x_2, y_2, z_2)} = \frac{f(ax_1, by_1, cz_1)}{f(ax_2, by_2, cz_2)} \tag{9}$$

is valid for all $a, b, c \in \mathbb{R}$.

It can be shown (see Bridgman [2]) that for every definitorial equation of a physical quantity eq. (9) is satisfied if the function $f$ is of the form

$$f(x, y, z) = Cx^{\alpha}y^{\beta}z^{\gamma} \tag{10}$$

where $C$ is a constant (*P*roduct-Theorem).

The dimensionless groups are quantities which are obtained by multiplication and division of physical quantities in such a way that their dimensions are effectively one.[2].

Using this definition of dimensionless groups, equation (10) can be written using dimensionless groups $\pi_i$, where the number of dimensionless groups for a given problem is always equal or less than the number of physical quantities involved. The equality only holds if there is the same number of physical quantities as basic dimension involved in the problem.

$$F(\pi_1, \pi_2) = C_1\pi_1^{\alpha_1}\pi_2^{\beta_1} \tag{11}$$

For non-definitory physical equations additive dimensionless terms might occur and the pure product form of equation (11) still holds for each summation term.

[2]Sometimes it is claimed that the dimension should be zero ("0"). Defining a physical variable to be a measured number times the dimension, this doesn't hold and the "dimension of a dimensionless number" has to be one ("1").

## 2.4 Relevance list

Dimensional analysis, although a very formal method, requires profound engineering skills in preparing the relevance list of the problem. This list contains all variables and constants relevant to the problem and can be found by examining the governing equations (if known) or by good engineering sense. One of the most interesting things about dimensional analysis is that you don't need to know any complete equations at the beginning, but you have to make very educated guesses about the variables involved [5].

The relevance list can be found by examining the analytical or differential equations governing the physical phenomenon. If the equations are unknown, the relevance list is generated by engineering insight and other domain knowledge. Only those variables and universal (dimensional) constants that are thought or known to form the governing equation of the problem under investigation are part of the relevance list.

## 2.5 Dimensional Matrix

Writing the exponents of the dimensions in a matrix with the physical quantities $x_i$ as rows and the basic dimensions as columns establishes the so-called dimensional matrix. The seven basic units of the SI system are often used as column titles, but theoretically correct, only the titles $[L]$ for length, $[T]$ for time, and $[M]$ for mass should be used. This leaves the specific choice of basic units open since all the quantities appearing in the laws of mechanics can be written in terms of length, time, and mass. Additionally, this shows the independence of the basic dimensions from the units of measurement. Sometimes it can be even more advantageous to use force $[F]$ and length $[L]$ instead of mass-time-length as base units.

Once the dimensional matrix is determined, the rank of the matrix is determined. The dimensional matrix (in fig. 1 left) has as many columns as the rank of the matrix is. If this is not the case columns (and therefore variables) have been omitted or are unnecessary (and can therefore be omitted).

The dimensional matrix is then transformed into an upper diagonal matrix (in fig. 1 right) using rank-preserving calculations. The dimensionless groups can then be directly determined from this special form of the dimensional matrix. The upper $r$ rows are the base rows and the lower $m$ rows are used for the determination of the dimensionless groups $\pi_j$ using the formula

$$\pi_j = x_j \prod_{i=1}^{r} x_i^{-\alpha_{ji}} \qquad ; j = r+1, \ldots, n \tag{12}$$

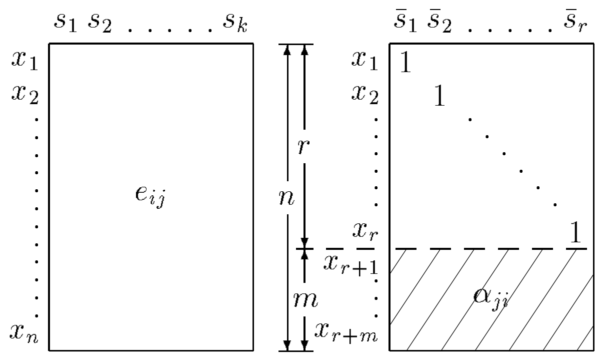



Figure 1: The dimensional matrix [11]

The set of dimensionless groups $\pi_j$ found with this scheme is not unique, other sets $\hat{\pi}_j$ can easily be created by

$$\hat{\pi}_j = \prod_{i=1}^{r} k_{ji} \pi_i^{\beta_{ji}} \tag{13}$$

where the $k_{ji} \in \mathbb{R}$ and the $\beta_{ji} \in \mathbb{R}$ are dimensionless constant factors. An infinite number of dimensionless groups can be found employing the $\Pi$-Theorem and equation (13) but only those groups that are 'minimal' in the sense that they can't be divided into smaller dimensionless groups are of physical relevance. This feature is called the property of structural independence [6, 5] and relies on the fact that the $\pi$-system found by equation (12) and (13) form a free abelian group.

# 3 Neural networks

Artificial Neural Networks (ANNs or NNs) were invented early in this century as a mathematical model of the human brain. Since then neural networks have been used in different applications, like pattern recognition, classification, clustering, function approximation, etc. Different neural network architectures and learning algorithms have been developed since [13].

Neural networks consist of computational elements called neurons which are arranged in layers and all of the neurons of one layer have input connections from all neurons of the preceding layer and output connections to all neurons in the subsequent layer. The weighted outputs of all neurons from the preceding layer are integrated using the integration function (usually a summation). The result of this operation, a scalar, is then transformed with the activation function (often a

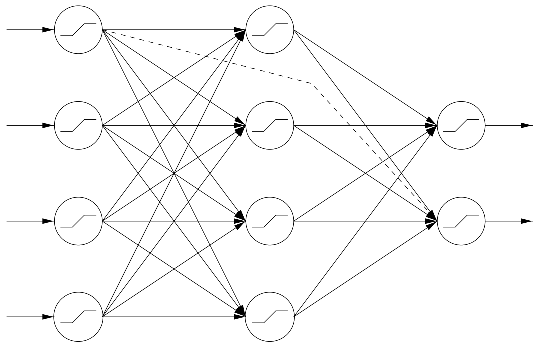


Figure 2: Feedforward neural network width three layers and one shortcut connection (dashed)

sigmoid, hyperbolic tangent or linear function) and then again transformed by the output function (usually the identity function). An input vector presented to the first layer is propagated to the last (output) layer using this scheme in each layer.

Usually the weights of the connection of the neural network can be adjusted using an appropriate learning algorithm, such as backpropagation [12, 9, 16].

## 3.1 Feedforward neural networks

Feedforward networks, also called multilayer perceptrons (MLPs) are simple neural networks as described above. They are often called backpropagation neural networks according to the most popular learning algorithm for this kind of neural network. For backpropagation [12] a pattern is presented to the network and the error of the output layer to a defined pattern output is calculated and propagated back to the input layer. Then the weights of the neural network are adjusted according to the error gradients. Another training algorithm for feedforward neural networks is the stochastic Threshold Accepting algorithm [4].

Feedforward neural networks provide a static mapping and using only linear activation functions and the summation as integration function, thy can be described in mathematical matrix notation. A special type of feedforward neural networks is the kind with shortcut connections, where the neurons of one layer are not only connected to all neurons of the succeeding layer but with all neurons in all subsequent layers.

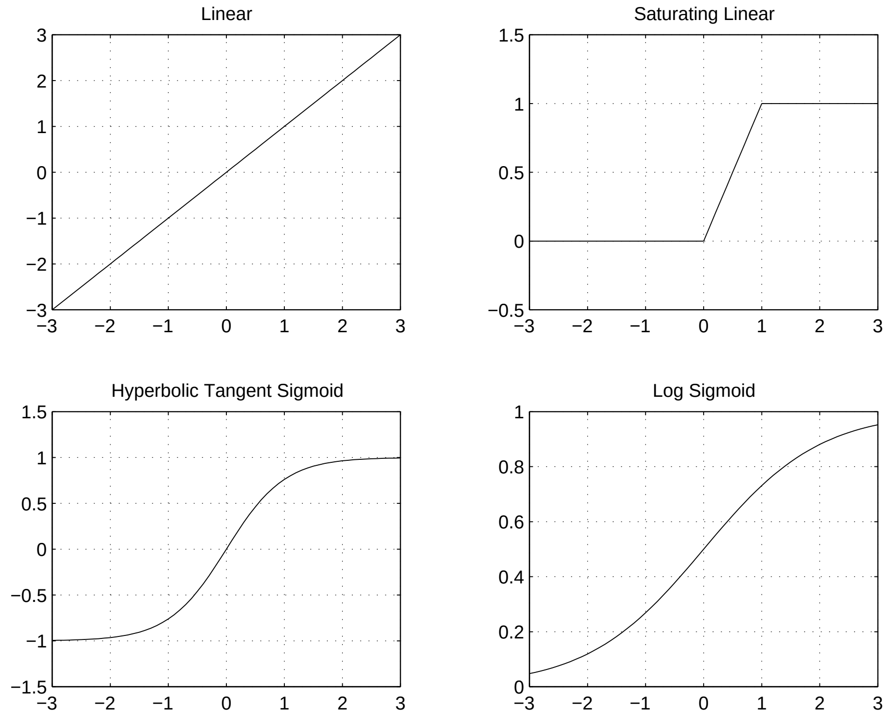



Figure 3: Activation function for neural networks

## 3.2 Generalized Radial Basis Function Neural Networks

Radial basis function neural networks (GRBF-NNs) [7] are an extension to radial basis function neural networks (RBF-NNs) which were mainly used for strict interpolation tasks. GRBF neural networks are equivalent to generalized splines. To approximate a function given in $N$ points $(\mathbf{x}, y)(\mathbf{x} \in \mathbb{R}^n, y \in \mathbb{R})$, one can find an approximation function of the following form

$$f(\mathbf{x}) = \sum_{i=1}^{K} c_i h(|\mathbf{x}_i - \mathbf{t}_i|) \tag{14}$$

where the $h(\cdot)$ are the radial basis functions at the $K$ centers $\mathbf{t}_i$ ($\mathbf{t}_i \in \mathbb{R}^n$). Approximating the function, the RBF centers $\mathbf{t}_i$ have to be chosen and the coefficients $c_i$ to be calculated.

GRBF neural networks are three-layer neural networks with linear activation functions in the input and the output layer and the radial basis function (usually a Gaussian) with different centers in the hidden layer. Since for GRBFs usually a

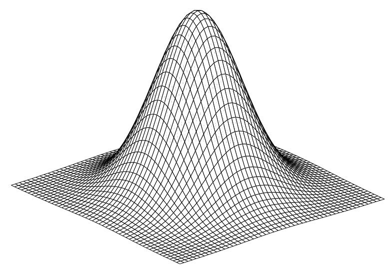
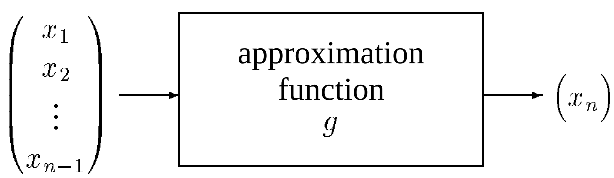


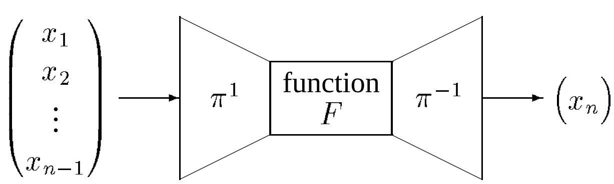
Figure 4: Twodimensional Gaussian activation function for GRBF neural network

direct learning algorithm is employed, the problem of local minima of backpropagation is avoided and the training itself is very fast [7]. These advantages are paid for with slow recalls when the networks are in use.

## 3.3 Dimensionally Homogeneous Neural Networks




Classical neural network




Similarity network

Figure 5: Neural networks for approximation [11]

Dimensionally homogeneous neural networks or for short *s*imilarity networks are restricted to dimensional homogeneous formulations. Due to this property, a significant reduction of the training space for the neural network weighted connections results. For these *s*imilarity networks, better approximation and learning properties for problems in the physical domain compared to classical neural network implementations have been observed [10, 11].

# 4 Applications of dimensional analysis

Dimensional Analysis has already been applied in engineering sciences like fluid dynamics, heat transfer and dynamics [14, 15]. The advantages are in the experimental sciences, where experimental effort can be saved, as well a in the theoretical sciences, where dimensional analysis gives a better insight into the scaling of problems and therefore into the nature of the problem itself.

## 4.1 Single degree-of-freedom system

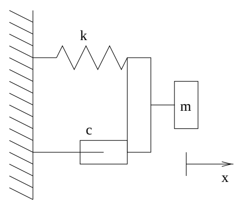



Figure 6: The mass-spring-dashpot system

The equation of motion for a mass-spring-dashpot system as shown in fig. 6 is found by first principles to be equal to

$$m\ddot{x} + c\dot{x} + kx = \hat{p} \cdot f_{ext} \tag{15}$$

and can be transformed into a dimensionless form (according to the procedures described around tables 1, 2, and 3)

$$\xi_{,\tau\tau} + 2D\xi_{,\tau} + \xi = f\left(\frac{\tau}{\omega_0}\right) \tag{16}$$

where $_{,\tau}$ means differentiation with respect to $\tau$.

The relevance list of this mass-spring-dashpot system is given by the variables occurring in the differential equation of motion (15).

| | | | |
|---|---|---|---|
| m | mass | $kg$ | $[M]^1$ |
| c | damping coefficient | $\frac{kg}{s}$ | $[M]^1\,[T]^{-1}$ |
| k | stiffness | $\frac{kg}{s^2}$ | $[M]^1\,[T]^{-2}$ |
| x | position | $m$ | $[L]^1$ |
| t | time | $s$ | $[T]^1$ |
| $\hat{p}$ | external force constant | $N$ | $[M]^1\,[L]^1\,[T]^{-2}$ |
| $F_{ext}$ | external force time dep. | - | - |

Table 1: The relevance list

Using this relevance list, the dimensional matrix is assembled and transformed into an upper diagonal matrix

| | $[M]$ | $[L]$ | $[M]$ |
|---|---|---|---|
| m | 1 | 0 | 0 |
| x | 0 | 1 | 0 |
| t | 0 | 0 | 1 |
| c | 1 | 0 | -1 |
| k | 1 | 0 | -2 |
| $\hat{p}$ | 1 | 1 | -2 |
| $f_{ext}$ | - | - | - |

$$\pi'_1 = \frac{c \cdot t}{m}$$
$$\pi'_2 = \frac{k \cdot t^2}{m}$$
$$\pi'_3 = \frac{\hat{p} \cdot t^2}{m \cdot x}$$
$$\pi'_4 = f_{ext}$$

Table 2: The dimensional matrix

The dimensionless groups in equation (16) can be derived from the dimensionless equations found in this dimensional analysis by applying equation (13)

| | | |
|---|---|---|
| $\pi_1$ | $D = \frac{\delta}{\omega_0} = \frac{1}{2}\sqrt{\frac{c^2}{m \cdot k}} = \sqrt{{\pi_1'}^2/\pi_2'}$ | attenuation constant |
| $\pi_2$ | $\tau = \omega_0 \cdot t = \sqrt{\frac{kt^2}{m}} = \sqrt{\pi_2'}$ | dimensionless time |
| $\pi_3$ | $\xi = \frac{k \cdot x}{\hat{p}} = \pi_2'/\pi_3'$ | dimensionless position |
| $\pi_4$ | $f_{ext} = \pi_4'$ | dimensionless ext. force |

Table 3: The dimensionless groups

To write the dimensionless form of the equation of motion the position coordinate $x$ is transformed with $\pi_3$ so that $x = \frac{\hat{p}\xi}{k}$ and the differentiation with respect to time $t$ is transformed into a differentiation with respect to dimensionless time $\tau$ using $dt = \frac{1}{\omega_0} d\tau$.

The modified dimensionless groups $\pi_1, \cdots, \pi_4$ might be more meaningful and better suited for interpretation by many readers as the original dimensionless groups $\pi_1', \ldots, \pi_4'$ originally found in the example dimensional analysis above, but mathematically speaking, both sets are equally valid.

## 4.2 Multiple degree-of-freedom system

The equation of motion for an undamped linear multi-degree-of-freedom (MDOF) system is given by the matrix equation

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \hat{\mathbf{p}}^T \mathbf{f} \tag{17}$$

where the mass matrix $\mathbf{M}$ and the stiffness matrix $\mathbf{K}$ are symmetric and the mass matrix is also positive definite. Equation (17) can be transformed in modal coordinates $\mathbf{q}$, which leads to uncoupled equations of motion in terms of modal generalized coordinates $q_i$. The mass and stiffness matrices are therefore transformed using the matrix of the eigenvectors of the undamped problem $\mathbf{\Lambda} = \text{eigv}\left(\mathbf{M}^{-1}\mathbf{K}\right)$ into

$$\mathbf{M}_m = \mathbf{\Lambda}^T \mathbf{M} \mathbf{\Lambda} = \mathbf{I} \tag{18}$$

$$\mathbf{K}_m = \mathbf{\Lambda}^T \mathbf{K} \mathbf{\Lambda} = \text{diag}(\omega_{0i}^2) = \boldsymbol{\omega} \tag{19}$$

$$\mathbf{q} = \mathbf{\Lambda}\mathbf{x} \tag{20}$$

where $\omega_{0i}$ are the eigenfrequencies of the undamped system. Using the equation of motion in modal coordinates and applying modal or Rayleigh damping ($\mathbf{C} =$ diag$(c_i)$), which allows for the concurrent diagonalization of all three matrices $\mathbf{M}, \mathbf{C}$, and $\mathbf{K}$, the resulting equation of motion reads

$$\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \boldsymbol{\omega}\mathbf{q} = \hat{\mathbf{p}}_m \cdot \mathbf{f} \tag{21}$$

The dimensionless groups corresponding to the terms of the single-degree-of-freedom equation of motion can be found as

$$\tau_i = \omega_{0i} \cdot t \tag{22}$$

$$D_{ii} = \frac{\delta_i}{\omega_{0i}} = \frac{1}{2}\frac{c_i}{\omega_{0i}} \tag{23}$$

$$\xi_i = \frac{\omega_{0i}^2 \cdot q_i}{\hat{p}_i} \tag{24}$$

The equation of motion can then be written in matrix notation as

$$\boldsymbol{\xi}_{,\boldsymbol{\tau\tau}} + 2\mathbf{D}\boldsymbol{\xi}_{,\boldsymbol{\tau}} + \boldsymbol{\xi} = \mathbf{f}\left(\frac{\tau_i}{\omega_{0i}}\right) \tag{25}$$

where the matrix of damping coefficients $\mathbf{D} =$ diag$(D_{ii})$ is given as a diagonal matrix. In the notation of equation (25) the index „ $_{,}\boldsymbol{\tau}$“ means component wise differentiation with respect to the vector of dimensionless times $\boldsymbol{\tau}$.

## 5 System Identification with neural networks

System identification is the task to identify the free parameters of a chosen model structure. In the context of this work the model parameters are derived from dimensional analysis and the constants and exponents that cannot be determined by dimensional analysis alone are then identified using a neural network approximation technique.

Let us suppose the function $g(x)$ of figure 5 is of monomial form as in equation (10), the dimensionless formulation in the $\pi$-domain is given by equation (11).

In an similarity network as shown in figure 5, this product form can be integrated using the logarithm as activation function of the input layer and the exponential in the hidden layer. By using this construction it is avoided to use the product as the integration function of the hidden layer and so the network can be trained using known algorithms like backpropagation, etc., and standard neural network packages which often don't use integration functions other than summation.

$$\begin{aligned} \pi_n &= \exp\left(\sum_{i=1}^{n-1} \alpha_i \ln(\beta_i \pi_i)\right) \\ &= C \prod_{i=1}^{n-1} \pi_i^{\alpha_1} \end{aligned} \tag{26}$$

Equation (26) shows that a feedforward neural network with the above architecture of activation functions implements a product function of the input arguments. The number of input arguments of the kernel of the similarity network is given by the number of dimensionless products which is less than the number of physical quantities involved in the problem.

## 5.1 Finite difference neural networks

In the time-domain systems can be identified approximating time-series data. This can be accomplished by using time-delayed input units in the neural network.

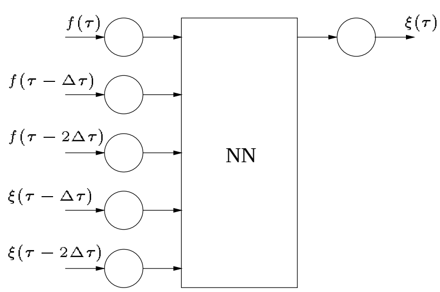



Figure 7: Feedforward neural network for system identification in the time-domain using a fixed sampling rate of $1/\Delta\tau$

## 5.2 State-space neural networks

The state-space notation of discrete time systems can be transformed into a feedforward neural network with time delay feedback lines. This type of feedforward

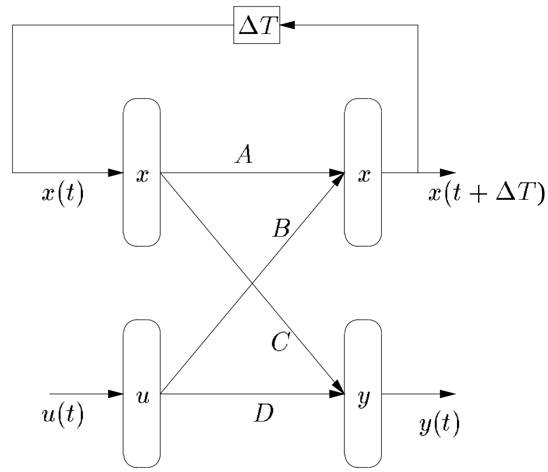



Figure 8: A discrete-time state-space neural network. $A, B, C$, and $D$ denote the weight matrices which are identical to the state-space matrices.

neural network can be treated as a MLP and be trained using standard algorithms like backpropagation, etc.

State-space neural networks with linear activation functions as shown in fig. 8 are the neural network representation of linear discrete-time state-space systems. This type of dynamic system is called a recurrent neural network with the special property, that it can be implemented as feedforward neural network with external time delay feedback lines. Special training algorithms like recurrent backpropagation and backpropagation-through-time (BPTT) have been developed for this kind of neural networks [9, 16].

# 6 Simulation Example

A single degree-of-freedom, single-input-single-output model of a mass-spring-dashpot system (fig. 6) has been simulated with a discrete time-step of $\Delta t = 0.01s$. The input signal was generated using discrete time steps of $\Delta t_{input} = 0.1s$) using the Matlab/Simulink software package [1].

For the identification in the time domain a neural feedforward network with five input neurons with linear activation function, five hidden neurons with hyperbolic tangent sigmoid activation function, and one output neuron with hyperbolic tangent sigmoid activation function was used. According to fig. 7 the input layer is a combination of the past two dimensionless input-output pairs

$(\xi(\tau - \Delta\tau), f(\tau/\omega_0))$ and the actual dimensionless input $f(\tau/\omega_0)$. The network was trained using standard backpropagation with a learning rate of $\eta = 0.01$ in $15.000$ epochs, reaching a summed square error of less than $5 \cdot 10^{-6}$.The identification was verified using the test data set shown in 9(b), the result is plotted in fig. 10. Both data sets contained 198 time steps with $-2.1227 < u(t) < 2.3867$ and $-5.45 \cdot 10^{-4} < y(t) < 6.94 \cdot 10^{-4}$.

This modelling principle for single input single output systems can also be used for multiple degree-of-freedom systems and easily expanded for multiple input multiple output systems. The neural network with hyperbolic tangent sigmoid activation function provides a static nonlinear mapping, dynamic effects are considered as inherent time delays in the choice of the input variables. Using a feed-forward neural network with linear activation functions with the above choice of input variables represents a linear difference equation.

The same system was identified using a radial basis function neural network with 20 neurons, using more neurons decreases the identification error but increases computation time and memory usage. The network was trained in 20 epochs to a sum-squared error of $2.15 \cdot 10^{-13}$. The system simulation with the RBF neural network using the not identified test data set is shown in figure 11.

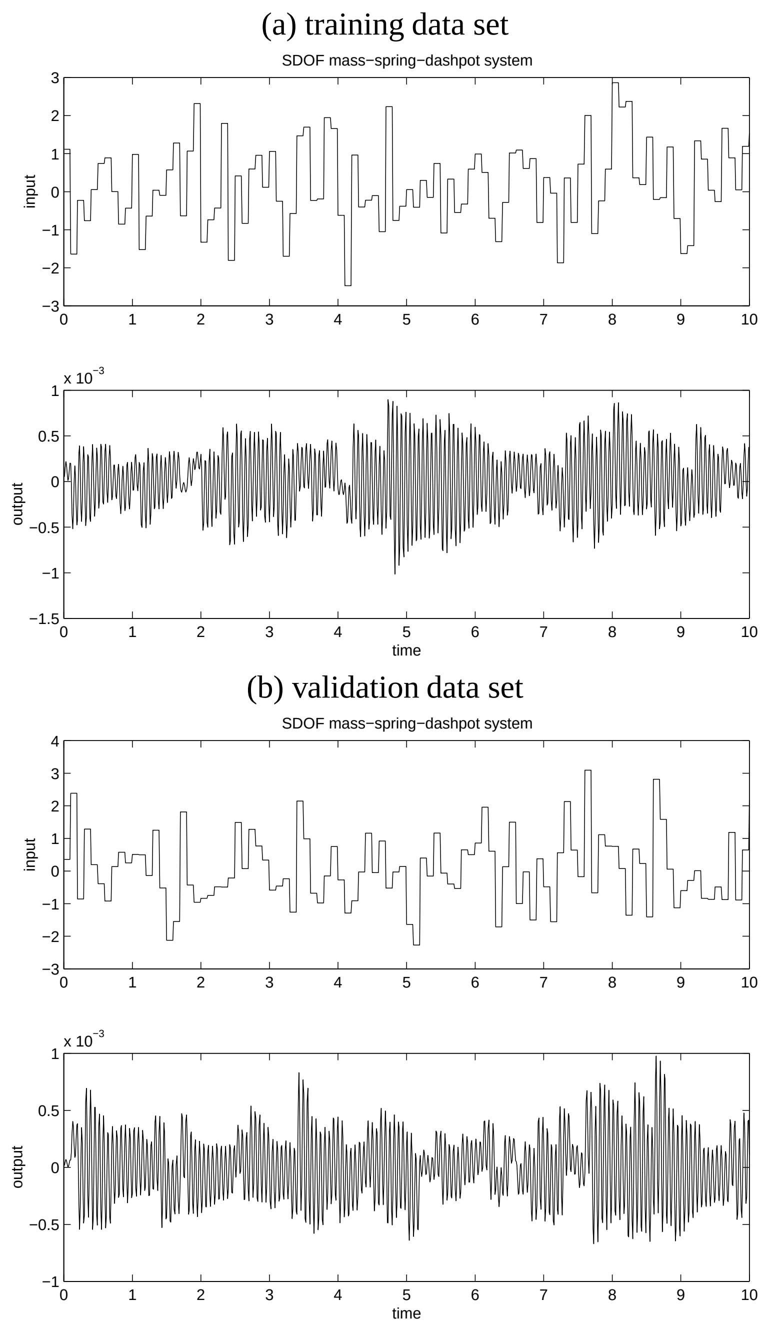



Figure 9: Data sets for single degree-of-freedom single-input-single-output mass-spring-dashpot system using a white noise input signal

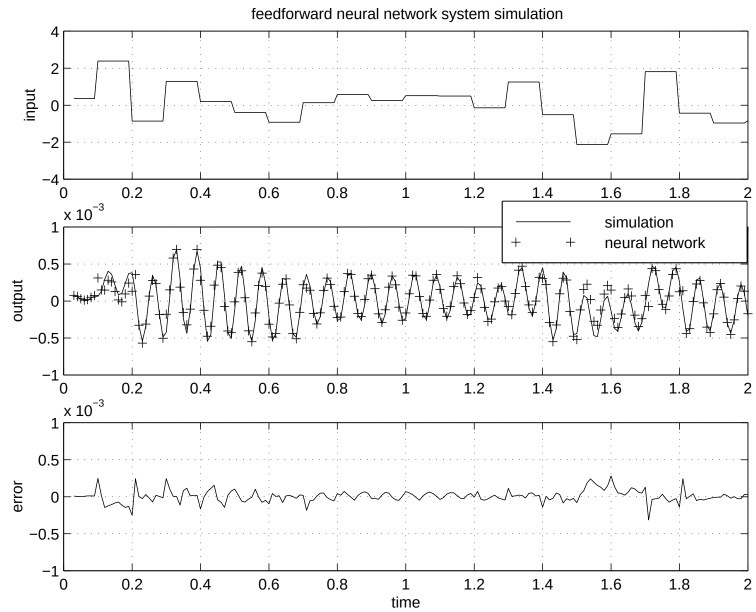



Figure 10: Neural Network simulating the mass-spring-dashpot single-input-single-output single degree-of-freedom system in the time domain

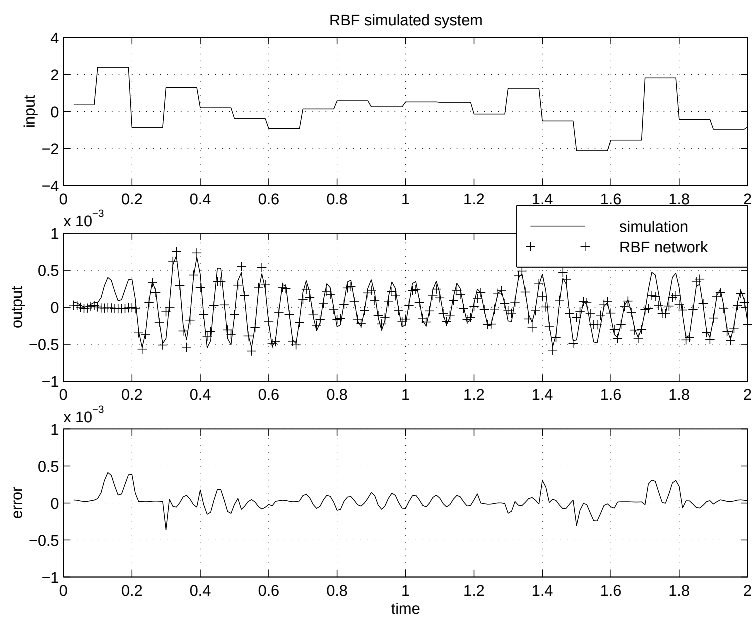



Figure 11: RBF neural network simulating the mass-spring-dashpot single-input-single-output singel degree-of-freedom system in the time domain

# 7 Conclusion

Several modelling techniques for dynamical systems are described and embedded into a neural network architecture. Using the principles of dimensional analysis, it has been shown that the combination of dimensional analysis and artificial intelligence has a high potential in dynamic system identification and represents a valuable prerequisite for successful neural control. The application of dimensional analysis and artificial intelligence on the equation of motion of linear single degree-of-freedom and multiple degree-of-freedom systems merges efficiently engineering domain knowledge with the learning potential of biological systems encoded in neural networks.

## 7.1 Acknowledgements

The support of this work by the Deutsche Forschungsgemeinschaft (DFG), under contract SFB 409 „Adaptive Structures in Aerospace Engineering“ Project A1, is acknowledged.

# References


[1] *Matlab Ver. 5*. The MathWorks, Cambridge, MA, 1997.

[2] P.W. Bridgman. *Dimensional Analysis.* Yale University Press, New Haven and London, 1931.

[3] E. Buckingham. The principle of similitude. *Nature*, 96:396–397, 1915.

[4] Scheuer Tobias Dueck, Gunter and Hans-Martin Wellmeier. Toleranzschwelle und sintflut: neue ideen zur optimierung. *Spektrum der Wissenschaft*, 1993.

[5] H. Görtler. *Dimensionsanalyse.* Springer, Berlin, 1975.

[6] Juri Pawlowski. *Die Ähnlichkeitstheorie in der physikalisch-technischen Forschung, Grundlagen und Anwendung.* Springer, Berlin, 1971.

[7] T. Poggio and F. Girosi. *A Theory of Networks for Approximation and Learning*. A.I. Memo 1140. Mass. Inst. of Techn., Cambridge, MA, July 1989.

[8] Andre Preumont. *Vibration Control of Active Structures*. Kluwer Academic Publishers, Dordrecht, 1997.

[9] Raul Rojas. *Theorie der neuronalen Netze.* Springer, Berlin, 1996.

[10] S. Rudolph. On topology and generalization in feed-forward neural networks. In B.H. Kröplin, editor, *Neuronale Netze in Ingenieuranwendungen, Internationals Workshop*, pages 17–26, Stuttgart, Germany, Fabruar 1996. Institut für Statik und Dynamik der Luft- und Raumfahrtkonstruktionen.

[11] S. Rudolph. On topology, size and generalization of non-linear feed-forward neural networks. *Neurocomputing*, 16(1):1–22, July 1997.

[12] D. Rumelhart and J. McClelland. *Parallel Distributed Processing*, volume I and II. MIT Press, Cambridge, MA, 1986.

[13] E. Sachenz-Sinencio and C. Lau, editors. *Artificial Neural Networks.* IEEE Press, New York, 1992.

[14] J. Stichlmair. *Kennzahlen und Ähnlichkeitsgesetze im Ingenieurwesen.* Altos-Verlag, Essen, 1990.

[15] Thomas Szirtes. *Applied dimensional analysis and modeling*. McGraw-Hill, 1997.

[16] Andreas Zell. *Simulation Neuronaler Netze.* Addison-Wesley, Bonn, 1994.