



HAL
open science

Use of a new trnsys type using neural network for non linear models creation

Laetitia Adelard, Franck Lucas, Philippe Lauret, Eric Fock

► **To cite this version:**

Laetitia Adelard, Franck Lucas, Philippe Lauret, Eric Fock. Use of a new trnsys type using neural network for non linear models creation. 15-18 September 2001, Sep 2001, Naples, Italy. pp.1-9. hal-01068560

HAL Id: hal-01068560

<https://hal.science/hal-01068560>

Submitted on 25 Sep 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Use of a new trnsys type using neural network for non linear models creation

LAETITIA ADELARD, *Assistant Professor, University of Reunion Island*

FRANCK LUCAS, *Researcher, University of Reunion Island, France*

PHILIPPE LAURET, *Assistant Professor, University of Reunion Island*

ERIC FOCK, *Researcher, University of Reunion Island, France*

Presenter: Dr. Laetitia ADELARD

Dr. Laetitia ADELARD's CV

Laetitia ADELARD is engineer from the National Engineering School of Tarbes, France. She obtained a PhD degree in Building physics in 1998. Subsequently (1994) she joined the Industrial Engineering Laboratory. She has since worked in one main subject area: Applications of the weather data generator RUNeole, (created while her thesis) in buildings physics. She participated to others study such as condensation pathologies in buildings in tropical humid climates. Actually, she is teaching at the University College of Technology of Saint Pierre, Réunion.

Use of a new trnsys type using neural network for non linear models creation

LAETITIA ADELARD, *Assistant Professor, University of Reunion Island*

FRANCK LUCAS, *Researcher, University of Reunion Island, France*

PHILIPPE LAURET, *Assistant Professor, University of Reunion Island*

ERIC FOCK, *Researcher, University of Reunion Island, France*

ABSTRACT

Model determination must sometimes be rapidly made because of missing data in a building simulation. This paper will deal about the creation of a new trnsys type using neural network method to create new non linear models. The new type 111 integrates both the models creation process, and the use of this model immediately in simulations. This new type will be used in Hvac building simulation in trnsys environment but it can also be used in a new weather data generator.

Illustration of such a type is made in HVAC simulation, for the determination of the cooling power, or for missing data in meteorological data files like long wave radiation estimation.

Keywords: Non linear stochastic models ; neural network ; Trnsys platform ; HVAC simulations.

Corresponding author :

Dr. Laetitia ADELARD

Laboratory of Industrial Engineering, University of Reunion Island

I.U.T. de Saint Pierre

40, Avenue de Soweto

97410 Saint Pierre, Réunion

FRANCE

Phone: +33 02 62 96 28 90; Fax: +33 02 62 96 28 99; E-mail: <adelard@univ-reunion.fr>

Use of a new trnsys type using neural network for non linear models creation

LAETITIA ADELARD, *Assistant Professor, University of Reunion Island*

FRANCK LUCAS, *Researcher, University of Reunion Island, France*

PHILIPPE LAURET, *Assistant Professor, University of Reunion Island*

ERIC FOCK, *Researcher, University of Reunion Island, France*

ABSTRACT

Model determination must sometimes be rapidly made because of missing data in a building simulation. This paper will deal about the creation of a new trnsys type using neural network method to create new non linear models. The new type 111 integrates both the models creation process, and the use of this model immediately in simulations. This new type will be used in Hvac building simulation in trnsys environment but it can also be used in a new weather data generator.

Illustration of such a type is made in HVAC simulation, for the determination of the cooling power, or for missing data in meteorological data files like long wave radiation estimation.

1. INTRODUCTION

The use of stochastic models is widespread, for example in climatic variables modelisation studies. Concerning linear models, we can quote (Hong T. and al., 1979), (Van Paassen and al., 1979). These models take into account the evolution of input variables or output variable in time. It can be written for example for an AutoRegressiveXternal model:

$$y(t) = f(u(t-1), u(t-2), \dots, u(t-n_a)) + h(y(t-1), y(t-2), \dots, y(t-n_b))$$

$y(t)$ stands for the output variable at the time t , $y(t-1)$, stands for the output variable at the time at the time $(t-1)$ (a lag space of one is taken into account for this variable). $U(t)$, $(t-1)$, stand for the external input variables that can be used, also with different lag spaces.

As it is hard to find simple linear relations in certain kind of applications, we decided to use neural network as a non-linear black box model. Neural networks are increasingly used in scientific studies, as illustrated by the works of Noorgaard, (Noorgaard M., 1996), and Kemmoku (Kemmoku and al. 1999) for examples.

2. NEURAL NETWORK DESCRIPTION :

Single stage neural network is an assembly of sigmoid functions associated to the input variables with pondered connections. Each neurone of the hidden layer has a network of connections (called synaptic connection) function of the input variables. Each connection has a different weight related to the importance of the connection. The weights θ_i are determined in a “learning” phase. This phase requires an experimental database comprising the input variables (the physical variables) of the model, and the expected output of the model. A single stage neural network includes a “hidden” and an “output” layer. Choosing a number of hidden neuron, the number of neuron in the output layer is fixed by the number of predicted outputs.

The learning phase includes the optimisation of the pondered connections by minimising the mean quadratic error $J(k, \theta)$. If we note $s(t)$ the experimental output with N elements, and $y(t, k)$, the output vector calculated at the k iteration:

$$J(k, \theta) = \frac{\sum_t^{1..N} (s(t) - y(t, k))^2}{2N} = \frac{\sum_t^{1..N} (\varepsilon(t, \theta))^2}{2N}$$

We proceed by iteration, and the weights estimated at iteration k can be obtained by the weights estimated at the iteration $k-1$ with the relation: $\theta(k) = \theta(k-1) + \mu f(k-1)$

For this, we use a Gauss Newton method to optimise the J criteria. This method takes advantage of the first order prediction error approximation :

$$\tilde{\varepsilon}(t, \theta) = \varepsilon(t, \theta)_{k-1} + (\varepsilon'(t, \theta)_{k-1})^T (\theta_k - \theta_{k-1})$$

Using the function $\psi(k, \theta) = \frac{d(y(k, \theta))}{d\theta}$, the gradient $G(k, \theta)$ and the Hessian $R(k, \theta)$ is obtained with the relations :

$$G(k, \theta) = \frac{1}{N} \sum_{t=1}^N \psi(k, \theta) \varepsilon(k, \theta) \text{ and } R(k, \theta) = \frac{1}{N} \sum_{t=1}^N \psi(k, \theta) \psi^T(k, \theta)$$

Multiples algorithms can then be used to determine these weights using first initial values, and adjusting them at each iteration of the learning phase. The optimisation algorithm for the search direction choice is based here on the Levenberg Marquardt method. Minimising the $J(k, \theta)$ criterion, this leads to the following relation :

$$f(k-1) = -[R(k, \theta) + \lambda \mathbf{I}]^{-1} G(k, \theta)$$

λ influences the step size. It's a function of the mean quadratic error $J(\theta)$. If λ increases, the step size is decreased, to find the best approximation of the minimum of $J(\theta)$. If in the opposite case, λ is close to zero, the step size increases to change significantly the search direction.

Therefore, because of its performances, such a tool should be interesting to integrate the TRNSYS platform (SEL, 1990), for modelisation and immediate use of non-linear models.

3. DESCRIPTION OF THE TYPE 111.

Type 111 has two aims. The first objective concerns the model determination. The second purpose is the immediate use of the determined non-linear model for simulations. Our type

allows several outputs using for the same entries. See figure 1 for a description of the network.

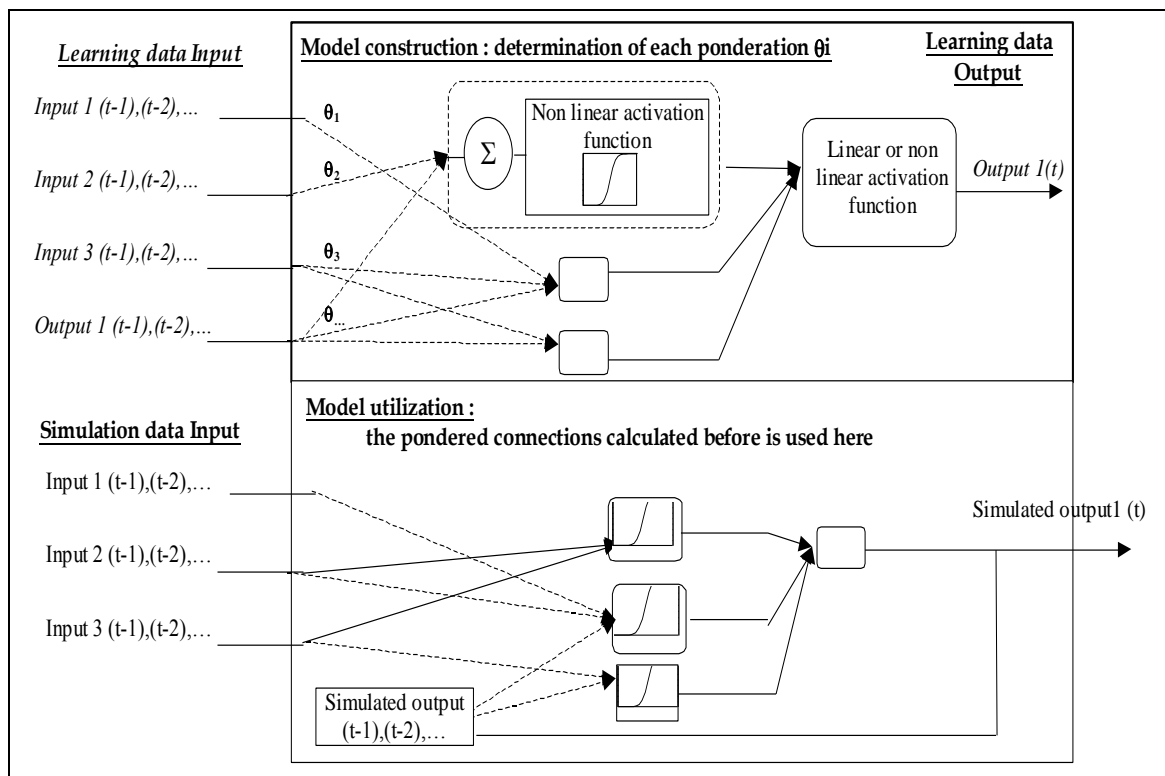


Figure 1: Description of the new type 111

Using the type 111 needs a succession of stages described below:

a) Choice of the network inputs, lag spaces and number of neuron:

The choice of the input variables and the lag space for all the variables used is left to the user. He will just have to indicate these numbers in the information window of the type (figure 2). The name and path of the learning data file must be indicated.

b) Parameters estimation:

The purpose of this stage is to determine the different connections weights thanks to a learning stage. The learning is based upon the minimisation of a quadratic criterion J , which is defined as the sum of the squares of the modelisation errors. The algorithm used here is made for a single hidden layer, which is the general configuration used by all users. Multiple hidden layers can be used for specific use. For example, it is useful to simulate simultaneously weekly and daily weather data (Kemmoku, 1999). This step should be studied later.

c) Implementation:

The optimal number of hidden neurons needed for the model is determined by various tests. Two criterions help evaluating the model. The first is the quadratic average error calculated between the measured output $s(k)$ and the calculated output $y(k)$, during the learning stage. The second is the average quadratic error calculated for a set of different data, allowing to check the network adaptability to new sets of data. It must be noted that too many hidden neurons lead to the diminution of the network adaptability to various sets of data.

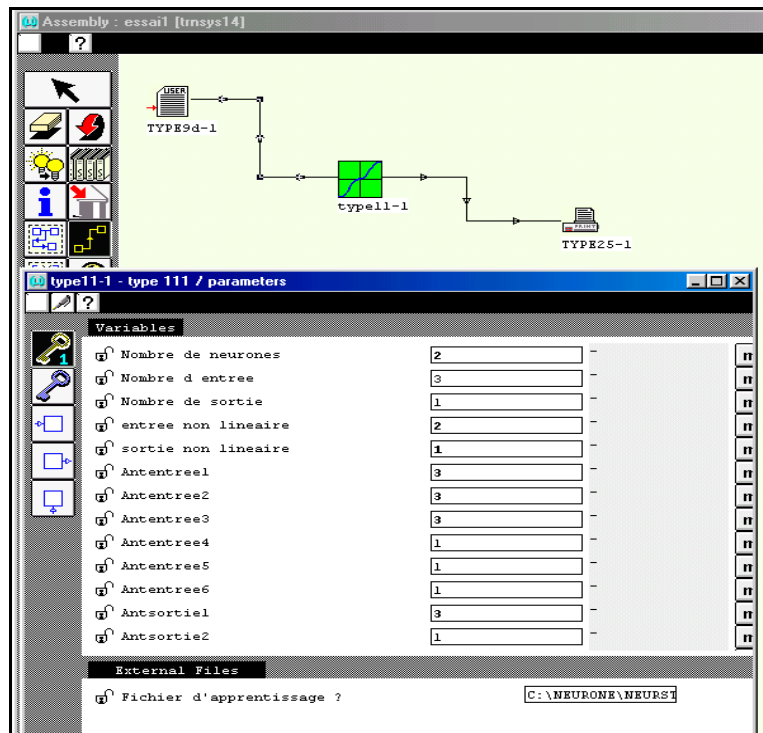


Figure 2 : Windows description for the trnsys type 111

4. RESULTS AND DISCUSSION

4.1 Modelling HVAC system

The simulation of thermal behaviour of building is a way to reduce the energy demand. Many model of HVAC have been developed these last years. We can distinguish two different types: the steady state models and the dynamic model. As the new thermal regulation in France uses dynamic models to evaluate building consumption in case of heating. There is a need of dynamic models in case of cooling. Beside detailed dynamic models, which are use by HVAC manufacturer for design purpose, many simplified models allow to determine thermal loads of buildings. These models are very helpful for building designers as long as they remain easy and fast to use. Several dynamic models have been developed in Reunion Island University, based on manufacturer's data. It appears that transient behaviour of the HVAC system is very difficult to consider. For example, the coefficient of performance over a long simulation period can vary from 1.2 to 3.5 depending on the part load factor. The on/off cycling of the system, which is difficult to evaluate, modifies its performance. A neural network using experimental data can simulate the consumption of the system taking into account its transient behaviour and the non-linear phenomena induced.

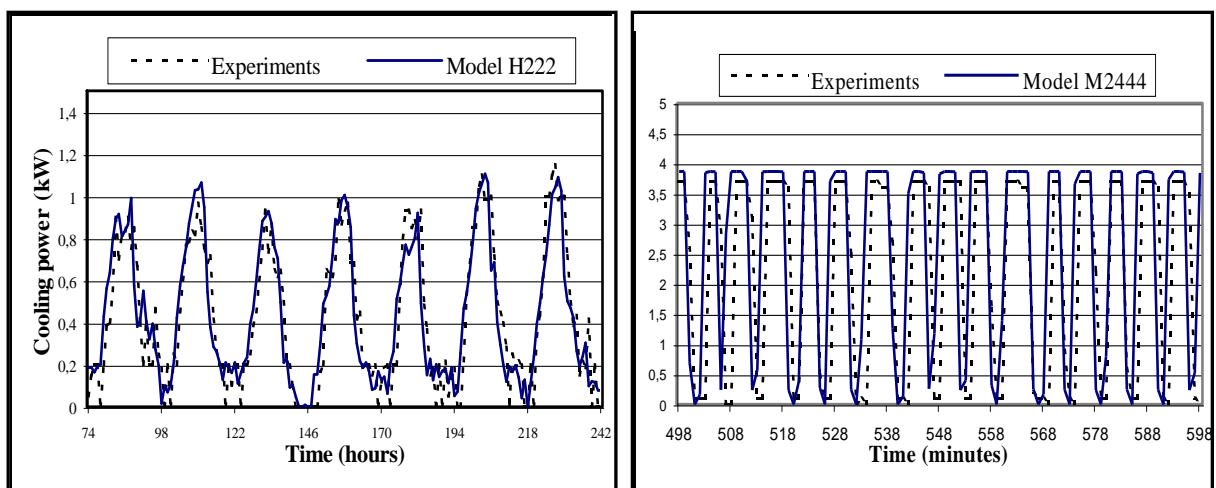
The model used is based on experiments carried out on a test cell, in real climatic conditions. The output values can be chosen by the user between the set of measured data (i.e.: the total, the sensible, the latent cooling power and the electrical power). The input data are the dry outside air bulb, the dry inside air bulb and the inside relative humidity. A previous study including the use of neural network based on manufacturer's data has been

developed with matlab software (Fock and al., 2000). Two time steps have been tested, a short time step file of one minute and an hourly time step. The hourly data are determined as the average value (on one hour) of the minute data. The short time step file includes 10000 measurements and the hourly file 400. Beside the learning file size, we also test the influence of the neural network parameters (i.e.: number of neurons, number of input and output antecedents). For both learning file we ran simulations with two and three neurons and a number of antecedent varying from two to four. The table 1 below presents the results obtained for each model.

Table 1: Results of models

Time step of learning file	Models	Number of neurons	Number of input antecedent (lag space)	Number of output antecedent (lag space)	Mean of residual	Standard deviation of residual	Average relative error
Minute	M244	2	4	4	-0.021	1.449	-0,014
	M233	2	3	3	-0.270	1.674	-0,178
	M222	2	2	2	0.180	1.043	0,119
	M322	3	2	2	-0.624	1.174	-0,411
Hour	H344	3	4	4	0.010	0.300	0.009
	H333	3	3	3	-1.034	0.853	-1.001
	H322	3	2	2	-0.012	0.207	-0.012
	H222	2	2	2	0.003	0.207	0.003
	H233	2	3	3	0.007	0.132	0.007
	H244	2	4	4	0.015	0.167	0.015

The best results were obtained for the model H222 with two neurons and two antecedents for the inputs and outputs. The more accurate short time model is the model M244 with two neurons and four antecedents.



Figures 3 and 4: Results of hourly and minute simulations with trnsys type 111.

It appears that, the neural network is very sensitive to learning file quality. The network better evaluates hourly data since the variations of the inputs are weak and smooth. One can also notice that best results are obtained with few neurons (2). The computing time decreases as the accuracy increases. The combination of number of neurons, number of antecedent will not leads to similar accuracy with the hourly time step and short time steps learning file. To give good results, neural network parameters must be adjusted to the purpose of the simulation. To define the best parameters, it is convenient to run simulations using simulation entries identical to learning data. Once the network is adjusted, simulations can be run with the real data.

5. CONCLUSION AND OUTLINES

To evaluate the accuracy of the models and to adjust the parameters, we used the type 111 alone to focus on its output. Then the model is included in building simulation project were the data under study is the inside temperature and the relative humidity of the zone. In this case we use the temperature and the relative humidity of the previous time step to feed the neural network. The output under study is the cooling power in the zone for the present time step. An assembly panel of the building project simulation is presented in figure 5 :

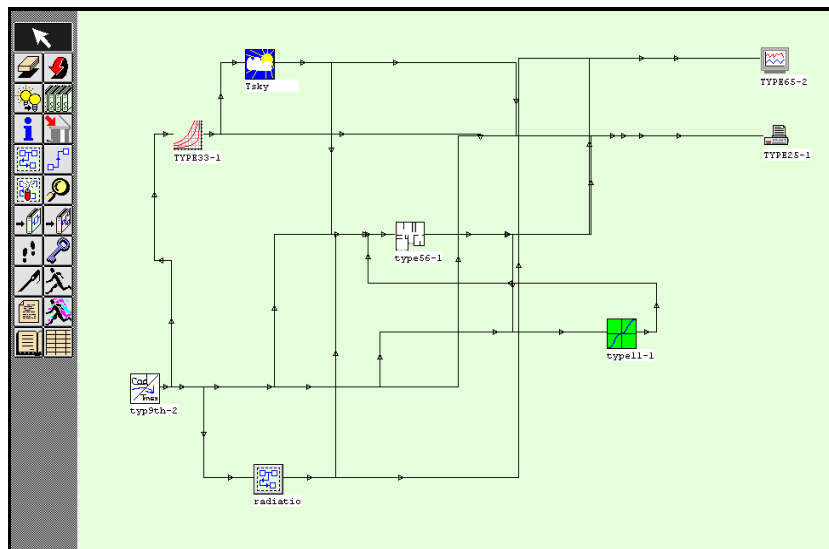


Figure 5 : TRNSYS assembly panel of a simulation project including Type 111

Results will be given in a further work. The algorithm used for the determinations of weights in the neural network can be optimised by including a “pruning” method, which leads to the automatic determination of the optimised number of hidden neurons. The use of genetic algorithm can ameliorate the time delay for the learning phase (Cegout, 1994). Some studies have been made for the optimisation. Its is absolutely necessary to test the robustness of the type with any sets of data or variables, so, the next step will be to test it in the aim of weather data modelisation (Adelard, 2000).

Acknowledgement : The authors wish to acknowledge M. Alain Cordier from the L.E.S.E.T.H., University of Toulouse for his collaboration and his comments on this work.

Références :

- A.H.C. Van Paassen and A.G. Dejong. 1979, The Synthetical Reference Outdoor Climate. *Energy and Buildings, Vol (2) Pages 151-161*
- TRNSYS: A Transient Simulation Program, 1990, *Solar Energy Laboratory, University of Wisconsin, Madison.*
- Cédric Gégout, Fabrice Rossi 1994. Initialisation des Réseaux de Neurones non récurrents à coefficients réels par algorithmes évolutionnistes, *Proceedings Congrès NeuroNimes*
- Hong T. , Y. Jiang, 1995. Stochastic Weather Model for Building HVAC Systems, *Building and environment, Vol. 30, N°4, p. 521-532, 1995*
- Noorgaard M., 1996. System identification and control with neural networks, *Thesis, Department of Automation , Technical University of Denmark.*
- Kemmoku Y., Orita S., Nakagawa S., Sakakibara T., June 1999. Daily insolation forecasting using a multi-stage neural network, *Solar Energy Vol. 66 pp. 193-200, N°3.*
- L. Adelard, H. Boyer, F. Garde, J.C. Gatina, 2000. A detailed weather data generator for building simulations. *Energy and Buildings, Vol. 31, pp 75-8,*
- Fock E., Garde F., Lauret P., Gatina J.C., 2000. Artificial neural networks for the prediction of cooling loads of HVAC systems : A case study under tropical climate, *World Renewable Energy Congress Proceedings, pp. 661-664.*