

Nevronske mreže za napovedovanje sile pri vrtanju

Drilling-Force Forecasting Using Neural Networks

Ivana Polanecka¹ - Marjan Korošec² - Janez Kopač²

(¹ Tehnična univerza Košice, Slovaška; ²Fakulteta za strojništvo, Ljubljana)

Nevronske mreže so orodje za pomoč pri napovedovanju, ki je uporabno na najrazličnejših področjih. Na podlagi podatkov iz nadzora postopka lahko napovedo lastnosti izdelka že med samim izdelovalnim postopkom. Narava časovne dinamike postopkov je pri tem lahko zelo pestra - od zelo dinamične do navidezno ustaljene. Naš cilj je izdelati bazo podatkov za obdelovalni stroj. Na podlagi zbranih podatkov bo v prihodnje mogoče napovedovati rezalne sile in rezalni navor za nove materiale, ki jih bomo obdelovali z odrezovanjem. Z napovedanimi silami pa bo nenazadnje mogoče tudi napovedati obrabo orodij.

© 2007 Strojniški vestnik. Vse pravice pridržane.

(Ključne besede: obdelovalni stroji, baze podatkov, sile vrtanja, napovedi, nevronske mreže)

Neural networks are a forecasting tool that can be applied in many fields. Process sensing and data acquisition, for example, can be used to improve both the machinability and product properties during the manufacturing process. The time dynamics of these processes may be anywhere from highly dynamic to quasi-stationary. Our goal was to create a machinability database. The collected data will provide a basis for forecasting the cutting forces and cutting torque for new materials in the future. The force forecasts will also allow tool-wear monitoring and prediction.

© 2007 Journal of Mechanical Engineering. All rights reserved.

(Keywords: machinability, databases, drilling forces, forecasting, neural networks)

1 NEVRONSKE MREŽE

1.1 Zgradba

Nevronske mreže so ena od možnosti strojnega učenja. Delo z nevronskimi mrežami omogočajo moderni programi, to so Matlab, NeuroSolution, SNS itn. Temelj uporabnosti nevronskih mrež je njihova učljivost. Nevronske mreže je moč naučiti oz. trenirati tako, da opravljajo določeno funkcijo, pri kateri se uspešnost učenja preverja z dodatnim testiranjem.

Nevronske mreže spadajo med metode umetne inteligence. Za reševanje problemov v proizvodnem strojništvu uporabljamo poleg nevronskih mrež tudi mehko logiko [11], evolucijsko računanje, genetske algoritme in genetsko programiranje [12], "multi agent tehnologijo" [13] ter sorodne tehnologije in metode.

Da bi načelo učenja kar najbolj ustrezalo človeškemu mišljenju, so umetne nevronske mreže (UNM) zasnovane po vzorcu nevronskih mrež v možganih. Le-te so fiziološki temelj za človekovo učenje in na splošno za mišljenje. Kakor v

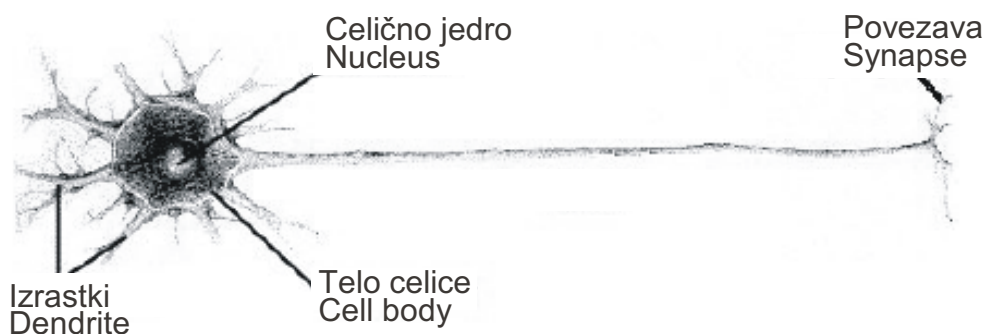
1 NEURAL NETWORKS

1.1 Structure

Neural networks are a subtype of machine learning. Computer programs such as Matlab, NeuroSolution, SNS, etc. allow data analysis with neural networks. The foundation for the usability of neural networks is their learning ability. Neural networks can be trained to execute a certain function and the success of the training can be verified by subsequent testing.

Neural networks (NN) are methods of artificial intelligence. For industrial problems solutions we used by NN also fuzzy logic [11], GA (genetic algorithm) and genetic programming [12], multi agent technology [13] and some more additional methods.

In order to make the artificial neural network (ANN) learning principle similar to human thinking ANNs have been designed on the basis of the neural networks found inside the brain. These networks are the physiological foundation for human learning and for thinking in general. As in the brain, data is



Sl. 1. Zgradba nevrona

Fig. 1. Neuron structure

možganih, se tudi v umetnih nevronskih mrežah podatki prenašajo po nevronih. Nevroni imajo v obeh primerih enako zgradbo. Najpomembnejši sestavni deli nevrona so telo celice, izrastki in vlakna (sl. 1).

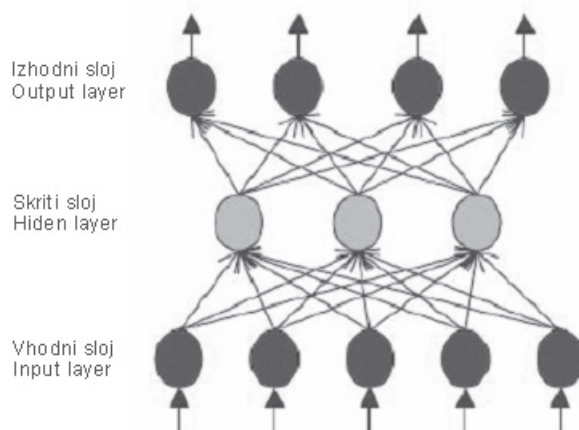
Nevronska mreža sestoji iz skupka nevronov in skupka uteži. Obnašanje mreže je močno odvisno od vplivov med tema skupkoma. UNM imajo vedno tri različne ravni nevronov, pri čemer sta prva in zadnja raven vidni, vmesna raven pa je skrita. Prva raven je vhodni vmesnik, zadnja pa izhodni vmesnik. Srednja raven je skrita in pogosto ni znano, kakšni postopki se odvijajo v njej. Srednjo raven zato praviloma obravnavamo kot črno škatlo. Skrita raven je po eni strani temelj za učinkovitost nevronskih mrež, po drugi strani pa prav zaradi njega ni mogoče sklepati o logičnih ozadjih napovedi (sl. 2) [1].

Ta model ustreza t.i. skritemu modelu Markova, ki dela z naključnimi stanji. Naloga pri tem je na podlagi zaporedja vidnih znakov

transferred in artificial neural networks through neurons. The structure of the neurons is the same in both cases. The essential parts of a neuron are the soma, the dendrites and the axon (Fig.1).

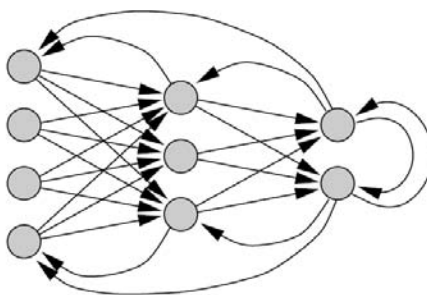
A neural network consists of a block of neurons and a block of weights. The performance of a network depends heavily on the interactions between these two blocks. ANNs usually have three different layers of neurons, the first and the last layer being visible and the intermediate layer being hidden. The first layer is the input interface and the last layer is the output interface. The intermediate layer is hidden, and often it is not clear what kind of processes are taking place inside it. The intermediate layer is therefore normally treated as a black box. The hidden layer, on the one hand, represents the foundation for the efficiency of neural networks, but on the other hand, also makes it impossible to make any conclusions about the logical background of a network's forecasts (Fig. 2) [1].

This model corresponds to the so-called Hidden Markov model that works with random states. The



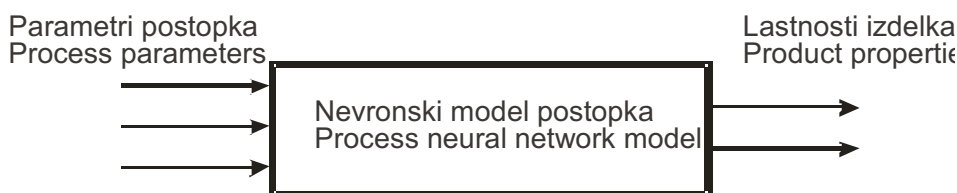
Sl. 2. Zgradba nevronske mreže

Fig. 2. Neural network structure



Sl. 3. Mreža s povratnim učenjem

Fig. 3. Network with backpropagation in learning



Sl. 4. Blokovna shema nevronskega modela postopka

Fig. 4. Block diagram of process neural network model

ugotavljati zaporedje skritih stanj. Za reševanje tega problema se uporablja t.i. vnaprejšnji algoritem.

Pri napovedovanju gre torej za naprej usmerjene mreže. Pri naprej usmerjenih mrežah se podatki prenašajo od nevronov na nižjih ravneh v nevronske na višjih ravneh, pri katerih so povezave usmerjene naprej.

Pri nadzorovanem povratnem učenju ("nazaj" – sl. 3) mreža grede v vstopne nevrone vstopni parametri, v izstopne nevrone pa želeni izstopi. Povratna mreža se tako uči povezav med vhodi in izhodi [2]. Povratne mreže imajo poleg vstopnih in izstopnih nevronov tudi poljubno število skritih nevronov, ki so prav tako razdeljeni v ravni. Vsaka raven, tako vstopna kot izstopna raven, sta v celoti povezani z naslednjo višjo ravni, medtem ko drugih povezav praviloma ni. Tako število skritih ravni kakor njihovo velikost (število nevronov) je treba določiti za vsak problem posebej. Pri večini uporab zadostujeta ena ali dve skriti ravni, le redko je potrebna še tretja raven. Slika 3 prikazuje triravensko povratno mrežo s štirimi vstopnimi, dvema izstopnima in tremi skritimi nevrone.

1.2 Potrebni podatki in delo s podatki

Nevronska mreža lahko opiše obnašanje sistemov tudi v tistih primerih, pri katerih znani

challenge is to determine the sequence of hidden states from the sequence of observable parameters. This problem is solved by using a so-called *forward*-algorithm.

Forecasting, therefore, involves *feedforward* networks. In *feedforward* networks, data is transferred from lower-layer neurons to higher-layer networks and the connections are directed forwards.

In supervised backpropagation networks (Fig. 3) the input neurons are fed with the input parameters and the output neurons are fed with the desired outputs. This is how a backpropagation network is trained to recognise the relations between inputs and outputs [2]. In addition to the input and output neurons, backpropagation networks also include an arbitrary number of hidden neurons, which are arranged into layers as well. Each layer – including the input and output layers – is entirely connected to its successor layer; there are usually no other connections. The number of hidden layers and their size (i.e., the number of neurons) has to be determined for each problem individually. Most applications can be adequately covered with one or two hidden layers, a third level is rarely used. Fig. 3 presents a three-layer backpropagation network with four input, two output and three hidden neurons.

1.2 Necessary data and data handling

A neural network is capable of describing the performance of a system even if the existing

vstopni podatki ne pokrivajo celega delovnega območja. Napovedi pa bodo zanesljive le pod pogojem, da uporabljene vstopne informacije izpolnjujejo določene najmanjše zahteve na obravnavanem oz. želenem delovnem območju.

Glede na vrsto naloge se je treba odločiti, ali razpoložljivi nabor podatkov zadostuje ali ne. Vsak podatkovni zapis je sestavljen iz n parametrov postopka in m lastnosti izdelka (sl. 4). Nevronska mreža običajno potrebuje več sto podatkovnih zapisov oz. odvisno od zahtevnosti naloge.

1.3 Učenje in testiranje

Učenje in testiranje sta najpomembnejša sestavna dela nevronske mreže, saj šele učenje in testiranje določita lastnosti nevronske mreže. Prvi korak učenja je vnos vprašanj z znanimi odgovori. Sledi primerjava odgovora mreže z znanimi pravilnimi odgovori. Nato se prilagajajo uteži povezav med posameznimi nevroni, dokler mreža ne da pravega odgovora. Postopek se ponavlja, dokler ni UNM ustrezno naučena. Pričakovani rezultat je nevronska mreža, ki lahko odgovarja tudi na vprašanja z neznanimi odgovori. Za preizkušanje pravilnosti sledijo testi z drugimi vprašanji, na katera so odgovori že znani. Če daje nevronska mreža pričakovane odgovore, je učenje končano, v nasprotnem primeru je potrebna nova učna doba. Postopek je tako iterativne narave in se izvaja tako dolgo, da so izračunani rezultati oz. napovedi zadovoljivi/-e.

1.4 Prezasičenost mreže

Če konec postopkov učenja in testiranja ni povsem jasno definiran, se pojavi problem

input data does not cover the whole operating range. However, the forecasts are only reliable if the input information fulfils certain minimum requirements on the operating or needed range.

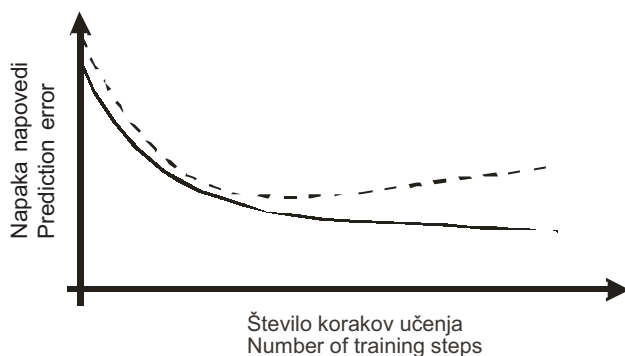
Depending on the setting of the task at hand, one has to decide if the available database is sufficient or not. Each data record consists of n process parameters and m product properties (Fig. 4). A neural network usually requires a couple of hundred data records, depending on the complexity of the task to solve.

1.3 Training and testing

Training and testing are the most important parts of a neural network, since it is the training and testing that determines the properties of a neural network. In the first step of the training, questions with known answers are entered. The network's response is then compared to the known correct answers. Next, the weights of the connections between individual neurons are altered until the network provides the correct answer. This procedure is repeated until the ANN is properly trained. The expected result is a neural network that is capable of responding to questions with yet unknown answers. The correctness is then verified with other questions, for which the answers are already known. The training terminates if the neural network is able to provide the expected answers, otherwise a new training sequence follows. This process is executed for as long as it is necessary in order to obtain satisfactory results.

1.4 Overfitting of ANN

In the case when the end of the training and testing sequences is not clearly defined, we are



Sl. 5. Problem prezasičenosti mreže (— testni podatki, — podatki učenja)
Fig. 5. The overfitting problem (— test data, — training data)

prezasičenosti mreže. Nevronska mreža je tedaj preveč specializirana za uporabljene učne dobe. Drugačne učne dobe zato dajejo druge rezultate. V fazi učenja se zmožnost razporeditve podatkov z vsakim naslednjim korakom sprva povečuje. V nekem trenutku pride do zasičenja, v tem trenutku se začne zmožnost razporeditve podatkov zmanjševati. Zasičena UNM je preveč prilagojena sestavi učnih podatkov in ni več usmerjena na dejansko obliko zgradbe, ki se jih mora naučiti. Pogosto ni preprosto prepoznati, kdaj je prišlo do zasičenja, mogoče pa je vnaprej napovedati, da bo kmalu prišlo do njega (sl. 5).

Iz tega izhaja, da število učnih korakov ne sme biti preveliko, saj nevronska mreža sicer zaradi problema prezasičenosti izgubi zmožnost posploševanja. Mreža zbira vse več podrobnosti o učnih podatkih, ki niso bistveni za rezultat testiranja. Nevronski mreži se tako sicer res zmanjša pogostnost napak pri posebnih podatkih, s katerimi pretežno zalagamo mrežo, izgubi pa zmožnost posploševanja, ki jo potrebuje za učinkovito delo, tj. zmožnost napovedati kar največ različnih primerov. Rezultat prenasičenja so zelo specializirane mreže, ki lahko izpolnjujejo svojo nalogo za določene uporabe, vendar so največkrat preveč specializirane.

Problem se je mogoče izogniti z izvajanjem testov z neodvisnimi testnimi dobami med učenjem. Pri tem dobimo neodvisne reference, ki jih lahko uporabimo za razpored.

1.5 Napovedi z nevronskimi mrežami

Napovedi je mogoče razložiti tudi kot napoved časovnih vrst z več ali manj šuma, pri čemer se časovna vrsta ravna po neki posredni zakonitosti. Matematični in statistični postopki, ki se uporabljajo za preprosto napovedovanje, v bistvu obravnavajo dve posebni vrsti problemov. Problem je bodisi mogoče opisati z velikim številom linearnih spremenljivk, katerih medsebojne povezave so preproste (linearne), ali pa ga je mogoče opisati z malo spremenljivkami, katerih medsebojne povezave so bolj zahtevne narave ([3] in [4]).

2 PREIZKUS

Cilj preizkusa je bila določitev značilk rezalnega postopka vrtanja z meritvami sil in navorov.

confronted with the network overfitting problem. This means that the neural network is too specialised for the training sequences that were used. Other training sequences then yield different results. In the learning phase, the ability to classify the data at first increases with every subsequent step. However, at a certain moment overfitting occurs and the network's data-classification ability starts to decrease. An ANN with overfitting is too adapted to the structure of training data and is no longer oriented on the actual form of structures it was supposed to train on. Recognising overfitting is often not an easy task, but it is possible to anticipate that it is going to occur soon (Fig. 5).

We can conclude that the number of training steps should not be too large, or else the neural network will lose its generalisation ability due to the overfitting problem. The network collects more and more details about training data, which are not relevant for the testing result. This may increase the frequency of errors for the special data, which is mostly fed to the network, but the network also loses its generalisation ability, which is necessary for efficient work, i.e., the ability to forecast many different cases. The result of overfitting is very specialised networks that may be able to fulfil their tasks for certain applications, but are mostly overly specialised.

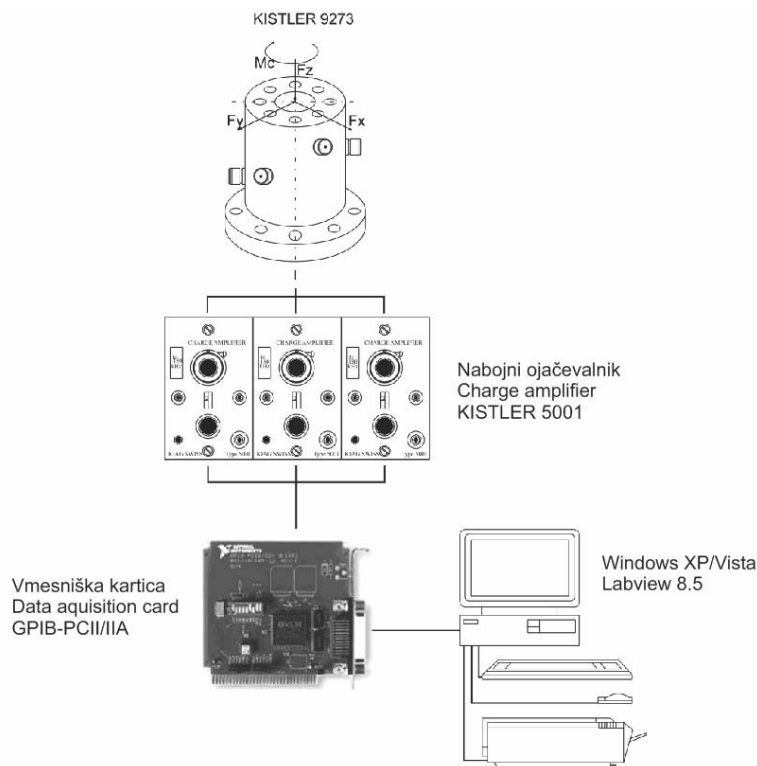
The problem can be avoided by executing tests with independent test sequences during the training. This yields independent references that can be used for the configuration.

1.5 Using neural networks for forecasting

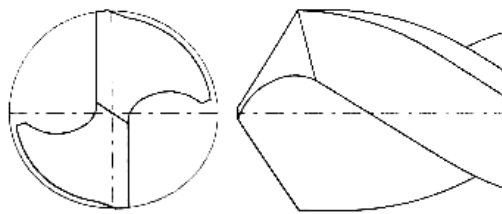
A prediction can also be interpreted as a forecast of a time series containing more or less noise, whereas the time series follows some implicit law. Mathematical and statistical procedures used for simple forecasting basically deal with two special types of problems. The problem can either be described by a large number of linear variables having simple (linear) interdependencies, or it can be described by a smaller number of variables with more complex interdependencies ([3] and [4]).

2 THE EXPERIMENT

The experiment was set up to measure the drilling-force characteristics. We have measured the feed



Sl. 6. Shema merilne verige za merjenje podajalnih sil in rezalnega navora
Fig. 6. Scheme of the measurement system for measuring the feed force and the cutting torque



Sl. 7. Rezalno orodje – vijačni sveder
Fig. 7. Cutting tool – drill

Merili smo podajalno silo (F_f) in rezalni navor (M_c) pri širšem naboru vrtilne frekvence in podajanja.

Merilna veriga (sl. 6) je bila sestavljena iz vrtilnega stroja DALMASTROJ SPLIT, dinamometra KISTLER 9273, digitalizatorja HP 54501A in računalnika s programskim paketom LABVIEW. Uporabljeni material je bilo jeklo 4742 UTOPN 1.2312 (40CrMnMoS8-6) [5]. Premer orodja je bil 8 mm (sl. 7). Rezultati meritev so zbrani v preglednici 1.

3 IZVEDBA NEVRONSKE MREŽE

Za ustvarjanje in učenje nevronske mreže smo uporabili Neural Network Toolbox (NNT), ki

force (F_f) and the cutting torque (M_c) over a range of different values of cutting speed and feed rate.

The measurement chain (Fig. 6) consisted of a DALMASTROJ SPLIT drill machine, a KISTLER 9273 dynamometer, a HP 54501A oscilloscope and a personal computer with LABVIEW software. The test material used was 4742 UTOPN WNr 1.2312(40CrMnMoS8-6) steel [5]. The tool diameter was 8 mm (Fig. 7). The measurement results are collected in Table 1.

3 NEURAL NETWORK REALISATION

The Neural Network Toolbox (NNT) was used to create and train the neural network. The

Preglednica 1. *Rezultati meritev – povprečna podajalna sila in povprečni rezalni navor*
 Table 1. *Experimental results – averaged feed force and averaged cutting torque*

f [mm/vrt]	v_c [m/min]	n [min ⁻¹]	Mc [Nm]	F_f [N]
0,050	6,28	250	1,735	930,621
0,080	12,56	500	2,776	1488,994
0,100	18,84	750	3,470	1861,243
0,120	20,72	825	4,267	1969,824
0,125	25,12	1000	4,445	2051,900
0,150	31,4	1250	5,334	2462,280
0,175	37,68	1500	6,223	2872,660
0,200	43,96	1750	6,266	3362,057
0,225	50,24	2000	7,049	3782,314
0,250	56,52	2250	7,833	4202,571
0,275	62,8	2500	8,616	4622,828
0,300	69,08	2750	9,399	5043,086
0,325	75,36	3000	10,182	5463,343
0,350	81,64	3250	10,966	5883,600
0,375	87,92	3500	11,749	6303,857
0,400	94,20	3750	12,532	6724,114
0,425	100,48	4000	13,315	7144,371
0,450	106,76	4250	14,099	7564,628
0,475	113,04	4500	14,882	7984,885
0,500	119,32	4750	15,665	8405,143

razširja možnosti uporabe paketa MATLAB. Orodna knjižnica programskega paketa MATLAB Neural Network Toolbox vsebuje vse običajne metode nevronske mreže, hkrati pa daje tudi veliko različnih učnih algoritmov. Kakor druge orodne knjižnice programskega paketa MATLAB tudi Neural Network Toolbox prevzame zelo veliko spretnega dela, uporabnik pa se lahko osredotoči le na bistvo problema. Na voljo so orodja za ustvarjanje, upodobitev, izvedbo in posnemanje različnih vrst mrež. Delali smo v ukazni vrstici z vnaprej definiranimi ukazi in v grafičnem uporabniškem vmesniku GUI. Rezultate iz grafičnega uporabniškega vmesnika MATLAB je mogoče izvoziti (okno *Workspace*), prav tako je podatke mogoče uvoziti v grafični uporabniški vmesnik [6].

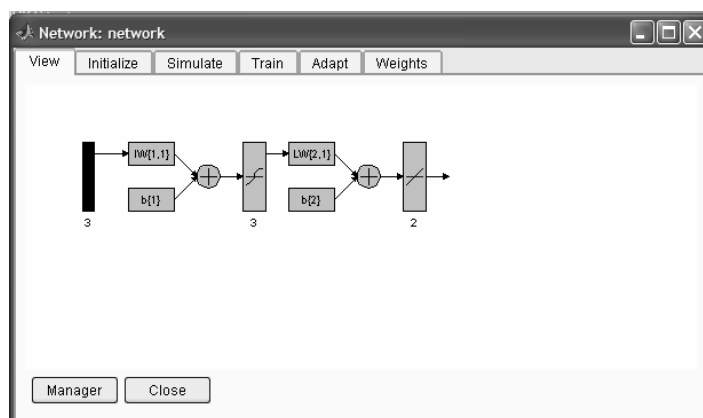
Mrežo ustvarimo v oknu Create New Network. V tem oknu definiramo sestavo mreže in vse funkcije, ki jih uporablja mreža. Vstopne podatke predstavljajo meritve dinamičnih in statičnih komponent sil in navorov. Vstopni podatki: n [min⁻¹], f [mm/vrt], v_c [m/min], izstopni podatki pa: M_c [Nm], F_f [N] ([7] in [8]).

Ustvarjena je bila nevronska mreža vnaprejšnjega tipa 3-3-2 in vrsta povratne mreže. Ta vrsta mreže uporablja pri prilagajanju utežnih

Neural Network Toolbox includes all the current neural network methods and makes available a number of different training algorithms. Like the other MATLAB toolboxes, the Neural Network Toolbox takes over the majority of the routine work, so the user can concentrate on the essential problems. NNT expands the possibilities of use for the MATLAB package. Tools for creating, visualising, realising and simulating different types of networks are available. In this work we used a command line with predefined commands, as well as a graphical user interface (GUI). The results can be exported from the MATLAB graphical user interface (*Workspace* window), and the data can also be imported into the graphical user interface [6].

The network is created in the Create New Network window. The network structure and all the functions used by the network are defined in this window. The measurements of the dynamic and static forces were used as input data. The input data are n [min⁻¹], f [mm/rev], and v_c [m/min], and the output data are M_c [Nm] and F_f [N] ([7] and [8]).

The backpropagation 3-3-2 type neural network was created with the subtype feed-forward backpropagation type of network. This network type uses the reverse-error propagation method for



Sl. 8. Grafični prikaz nevronske mreže

Fig. 8. Graphical overview of used neural network

koeficientov metodo povratnega učenja. Območje vhodnih podatkov je moč določiti v polju INPUT RANGES, kjer izberemo vstopno datoteko.

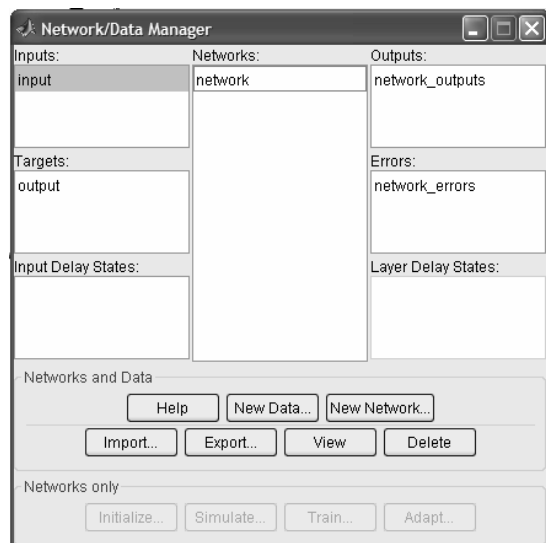
GUI nntool v paketu MATLAB 7 ustvari na podlagi vnesenih nastavitev skupinsko shemo nevronske mreže (sl. 8). V izbranem primeru lahko prepoznamo tri vstopne nevrone, tri skrite nevrone in dva izstopna nevrona. To zadostuje, saj mora mreža napovedovati tako podajalno kakor rezalno silo ([9] in [10]).

Vstopna raven (IN: vstopni nevroni) so izmerjene vrednosti, ki so bile med delom prepoznane kot najprimernejše. Srednja raven (SN: skriti nevroni) je t.i. skrita raven, katere

altering the weight of the coefficients. The input data range is set in the INPUT RANGES field, where the input file option has to be selected.

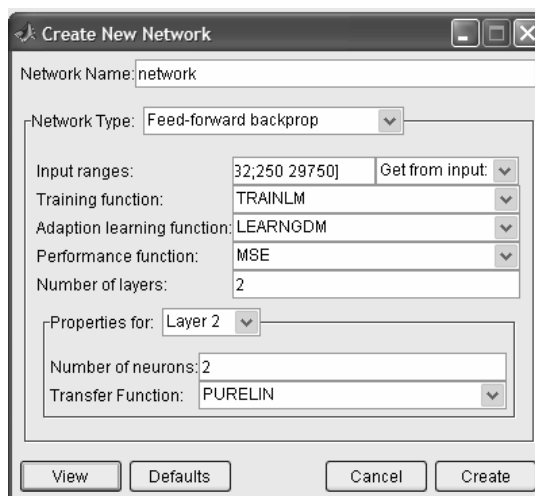
The MATLAB 7 GUI nntool uses the entered settings to create the neural network's block diagram (Fig. 8). In the current problem it is possible to input three neurons, three hidden neurons and two output neurons. This will suffice, as the network is supposed to forecast the feed force as well as the cutting torque ([9] and [10]).

The input layer (IN: input neurons) are the measured values, which were found to be the most suitable in the course of the project. The intermediate layer (HN: hidden neurons) is the so-



Sl. 9. Obravnava po uvozu vstopno-izstopnih podatkov

Fig. 9. Network/Data Manager after importing the input/output data



Sl. 10. Okno za ustvarjanje nevronske mreže

Fig. 10. Interface window for neural network creation/preparation

velikost je bila med delom optimizirana na tri nevrone. Zadnji nevron (IN: izstopni nevron) pa je izstopna veličina.

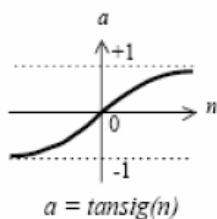
Za učenje mreže so bile izbrane naslednje funkcije [8]:

- *TRAINING FUNCTION* (uporabljena je bila TRAINLM)
- *ADAPTATION LEARNING FUNCTION* (kot učna funkcija za prilagajanje uteži pri povratnem učenju je bila nastavljena LEARNGDM)
- *PERFORMANCE* (Izbrana je bila MSE (Mean Square Error). Ta funkcija uporablja srednjo vrednost kvadratične napake)

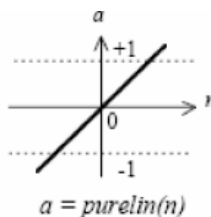
Vstopna raven v Matlabu ne spada med ravni ki sodelujejo pri preračunavanju, zato ji ni treba določiti nobenih funkcij.

- *TRANSFER FUNCTION*. Izbrana mreža ima skrito raven s tremi nevroni, sprožilna funkcija je TANSIG. V izhodnem nivoju sta dva nevrona, sprožilna funkcija pa je PURELIN.

Tansigm pomeni sigmasto funkcijo (1), kjer je oblika funkcije:



a)



b)

Sl. 11. Oblika uporabljenih funkcij
Fig. 11. Structure of the used functions

$$a = \frac{2}{1 + e^{-2n}} - 1 \quad (1).$$

Purelin je linearna funkcija (sl. 11) z naslednjimi lastnostmi:

Med učenjem imamo pregled nad naslednjimi parametri: učna funkcija, število dob (definirano število in število že končanih dob), srednja vrednost kvadratične napake (dejanska in zahtevana vrednost), gradient (dejanska vrednost in vrednost, pri kateri se postopek učenja konča). Nastavitev parametrov mreže za učenje je prikazana na sliki 12, potek učenja pa na sliki 13.

Nevronske mreže je mogoče v celoti ustvarjati in učiti v Matlabu. Naučene mreže je nato zaradi enostavnosti in preglednosti mogoče predstaviti in uporabiti v Simulinku (sl. 14).

called hidden layer that was optimised to include three neurons during the project. The last neuron (ON: output neuron) is the output parameter.

In this case the following functions used by the network and sets for the training method were selected [8]:

- *TRAINING FUNCTION* (we used TRAINLM)
- *ADAPTATION LEARNING FUNCTION* (we used the LEARNGDM learning function for backpropagation/bias)
- *PERFORMANCE* (for the predictor we used the Mean Square Error)

The input layer is not involved in Matlab's calculations, so no functions have to be assigned to it.

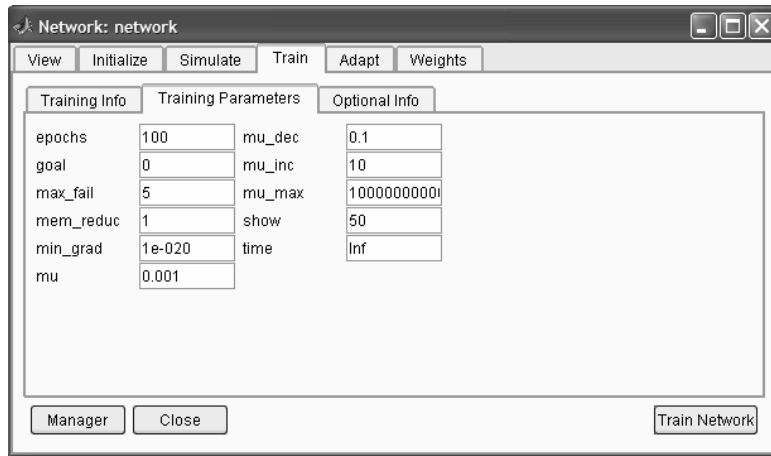
- *TRANSFER FUNCTION*. The selected network has a hidden layer with three neurons; the activation function is TANSIG. The output layer has two neurons and the PURELIN activation function.

Tansig is the sigmoid function (1), where the function's shape is defined as:

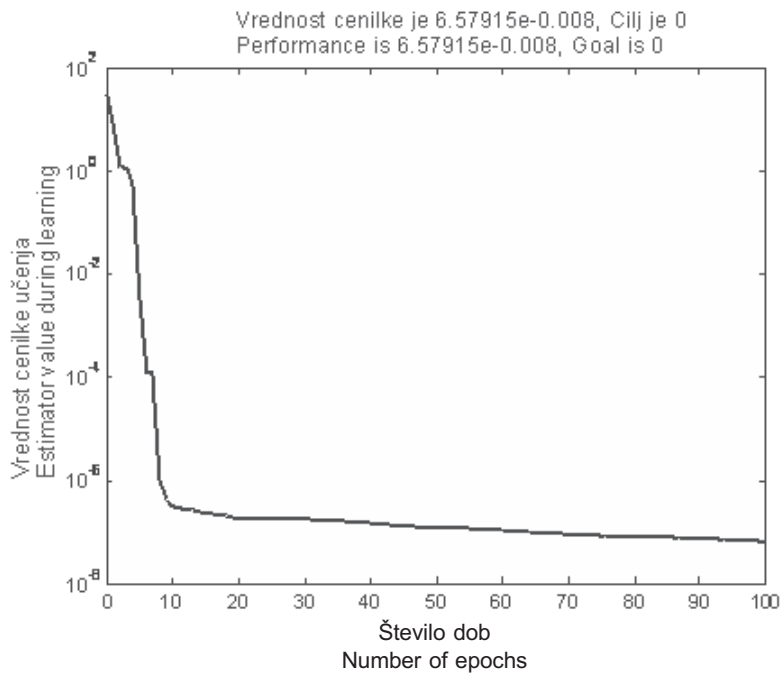
Purelin is a linear function (Fig. 11) with the following characteristics:

The following parameters are the output during the training: the training function, the number of epochs (predefined and elapsed), the mean square error (actual and specified value), and the gradient (actual value and value that terminates the training procedure). The network parameter settings for the training are shown in Fig. 12 and the training process performance is shown in Fig. 13.

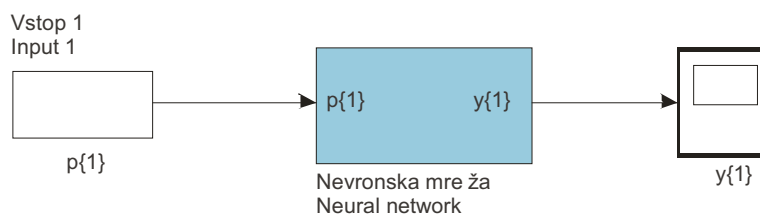
The neural networks can be created and fully trained in Matlab. The trained networks can also be embedded and simulated in Simulink because of their user friendliness and transparency (Fig. 14).



Sl. 12. Nastavitev parametrov nevronske mreže
Fig. 12. Setting the neural network training parameters



Sl. 13. Potek postopka učenja
Fig. 13. Training process performance



Sl. 14. Nevronska mreža kot podsistem v Simulinku
Fig. 14. Neural network as a subsystem in Simulink

Preglednica 2. *Napovedane testne vrednosti*

Table 2. *Predicted test values*

Želene vrednosti Vanted values		Napovedane vrednosti Predicted values	
M_c [Nm]	F_f [N]	M_c [Nm]	F_f [N]
1,735	930,621	1,712	931,213
17,225	9241,97	17,431	9238,673
36,033	19333,8	37,827	19322,82
47,002	25219,11	45,977	25221,23
62,671	33626,69	63,971	33618,17

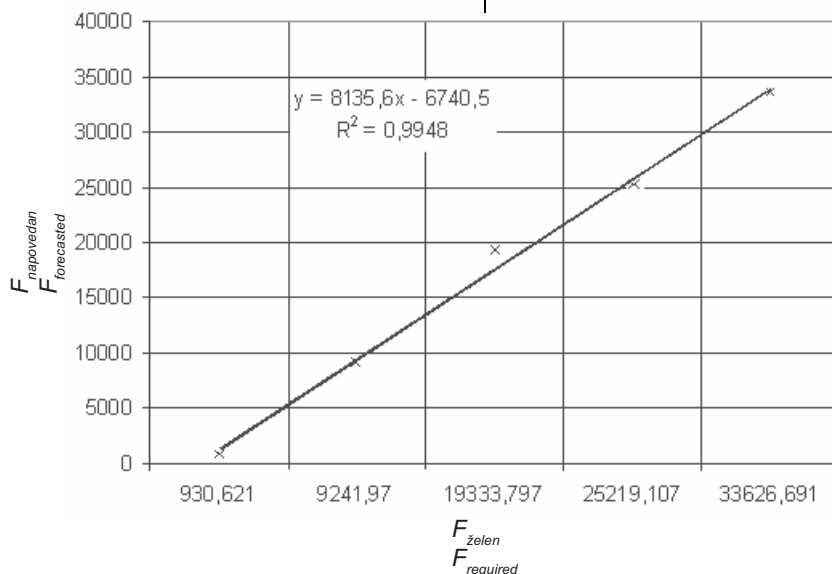

Sl. 15. *Odnos med želeno in napovedano podajalno silo (x – meritve, – linearna odvisnost F_f)*

Fig. 15. *The correlation between the required and the forecasted feed force (x – measurement, – linear approximation F_f)*

V preglednici 2 so testne vrednosti, s katerimi nevronska mreža ni imela opravka (niso bili vgrajeni v postopek učenja mreže). Odvisnost med napovedanimi in zelenimi vrednostmi je prikazana na slikah 15 in 16.

Odvisnost med obema vrednostma bi bilo mogoče še izboljšati, če bi uporabili več učnih podatkov, a je kljub temu treba poudariti, da so dobljeni rezultati v okviru naših pričakovanj.

4 SKLEP

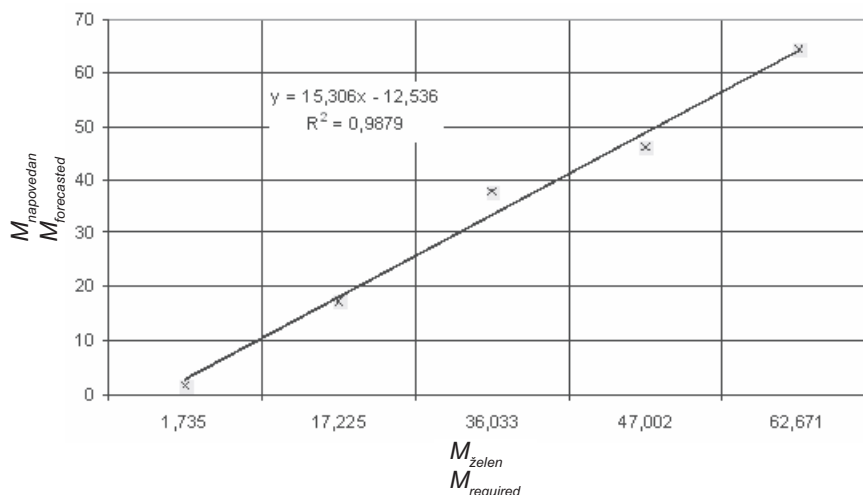
Opisana tehnika napovedovanja z nevronskimi mrežami tako ni omejena samo na nekatere tehnologije. Njene uporabe najdemo na najrazličnejših področjih (vremenskih napovedih, napovedih obnašanja postopkov itn.). Število vstopnih in izstopnih parametrov pri nevronskih mrežah je

Table 2 shows the test values that have not been seen by the neural network. The correlation between the forecasted and the required values is shown in Figures 15 and 16.

The correlation between both values could be further improved by using more training data. However, the obtained results are still within our expectations.

4 CONCLUSION

The described forecasting technique is not limited to any technology, its applications can be found in various fields (weather forecasting, the prediction of process behaviour, etc.). The number of the neural network's input and output parameters is arbitrary. In addition to that, the neural networks



Sl. 16. Odvisnost med želenim in napovedanim rezalnim navorom

(x – meritve, – linearna odvisnost M_f)

Fig. 16. The correlation between the required and forecasted cutting torque

(x – measurement, – linear approximation M_f)

poljubno. Nevronske mreže so razen tega poljubno razširljive, uporabe pa so lahko tudi močno nelinearne. Algoritmi v orodnih knjižnicah paketa MATLAB omogočajo hitro in učinkovito reševanje nalog.

Iz prispevka je razvidno, da je moderne metode modeliranja postopkov moč uporabiti tudi na odrezovalnih postopkih. In sicer za napovedovanje rezalnih sil/navorov pri določenih parametrih odrezovanja.

Sile oz. navori pri odrezovanju pa predstavljajo značilno cenilko obnašanja samega postopka. Posredno se kažejo na vibracijah in obrabi orodja. Tako je na podlagi te zamisli mogoče napovedovati tudi obrabo rezalnih orodij.

are optionally expandable and the applications can be highly non-linear. The algorithms included in MATLAB Toolboxes enable quick and efficient problem solving.

This paper proves that modern methods for process modelling can also be used for metal-cutting processes, in particular for the prediction of cutting forces/torques corresponding to the cutting parameters (feed, speed, depth, etc.)

While forces and torques during the cutting processes represent the dominant estimator for the cutting process' status, and they both have an influence on the cutting-tool wear, vibrations, etc., based on this idea it is also possible to predict the amount of cutting-tool wear.

5 LITERATURA

5 REFERENCES

- [1] Zakharian, S. (1998) Neuronale Netze für Ingenieure: Arbeits- und Übungsbuch für regierungstechnischen Anwendungen. *Vieweg-Verlag*.
- [2] <http://www.ece.ubc.ca/~elec592/PDFfiles/Backpropagation.pdf>
- [3] Armarego, E.J.A., Smith, A.J.R., Gong, Z.J. (1990) Four plane facet point drills – Basic design and cutting model predictions. *STC C 39/1/1990*, p. 41
- [4] Korošec, M., Balič, J., Kopač, J.: (2005) Neural network based manufacturability evaluation of free form machining. *Int. j. mach. tools manuf.* [Print ed.], January 2005, vol. 45, iss. 1, str. 13-20.
- [5] Krajnik, P., Korošec, M., Pušavec, F., Kopač, J.: Multiple layer perceptron simulation and advanced metamodeling of centreless grinding process. *AMPT 2006*, Las Vegas, NV, July 30-August 3, 2006.
- [6] Matlab 7 Help
- [7] Scherer, A. (1997) Neuronale Netze. Grundlagen und Anwendungen, *Vieweg-Verlag*.

- [8] Tič, R. (1994) Možnost kontrole vrtnja s pomočjo nevronske mreže. Diplomski naloga visokošolskega študija, *Univerza v Ljubljani, Fakulteta za strojništvo*.
- [9] Matlab select 2/03 http://www.mathworks.com/company/newsletters/matlab_select/files/matlab_select_02-03.pdf
- [10] <http://www.boerde.de>
- [11] Tsourveloudis, N., Ioannidis, S., Valavanis, K. (2006) Fuzzy surplus based distributed control of manufacturing systems, *Advances in Production Engineering & Management*, Vol. 1, No. 1, December 2006.
- [12] Balič J. (2006) Intelligent CAD/CAM systems for CNC programming - An overview, *Advances in Production Engineering & Management*, Vol. 1, No. 1, December 2006.
- [13] Nestler A., Dang T. N. (2008) The application of multi-agent systems for process management and NC data transfer aided by a DNC system, *Advances in Production Engineering & Management*, Vol. 3, No. 1, January 2008, on-line November 2007: www.apem-journal.org.

Naslova avtorjev: Ivana Polanecka

Tehnična univerza Košice
Fakulteta za strojništvo
Letná 9
042 00 Košice, Slovaška
ivana.polanecka@tuke.sk

dr. Marjan Korošec
prof.dr. Janez Kopač
Univerza v Ljubljani
Fakulteta za strojništvo
Aškerčeva 6
1000 Ljubljana
marjan.korosec@fs.uni-lj.si
janez.kopac@fs.uni-lj.si

Authors' Adresses: Ivana Polanecka

Technical University Košice
Faculty of Mechanical Eng.
Letná 9
042 00 Košice, Slovakia
ivana.polanecka@tuke.sk

Dr. Marjan Korošec
Prof. Dr. Janez Kopač
University of Ljubljana
Faculty of Mechanical Eng.
Aškerčeva 6
SI-1000 Ljubljana, Slovenia
marjan.korosec@fs.uni-lj.si
janez.kopac@fs.uni-lj.si

Prejeto: 25.7.2007
Received:

Sprejeto: 28.9.2007
Accepted:

Odperto za diskusijo: 1 leto
Open for discussion: 1 year