

Универзитет у Београду – Машински факултет

Дипломске академске студије

Модул: *Дизајн у машинству*

Предмет: *Методе одлучивања*

ПА-5 Параметри пројектовања базирани на оптимизацији токова материјала

Предметни наставник: *Проф. др Зоран Миљковић*

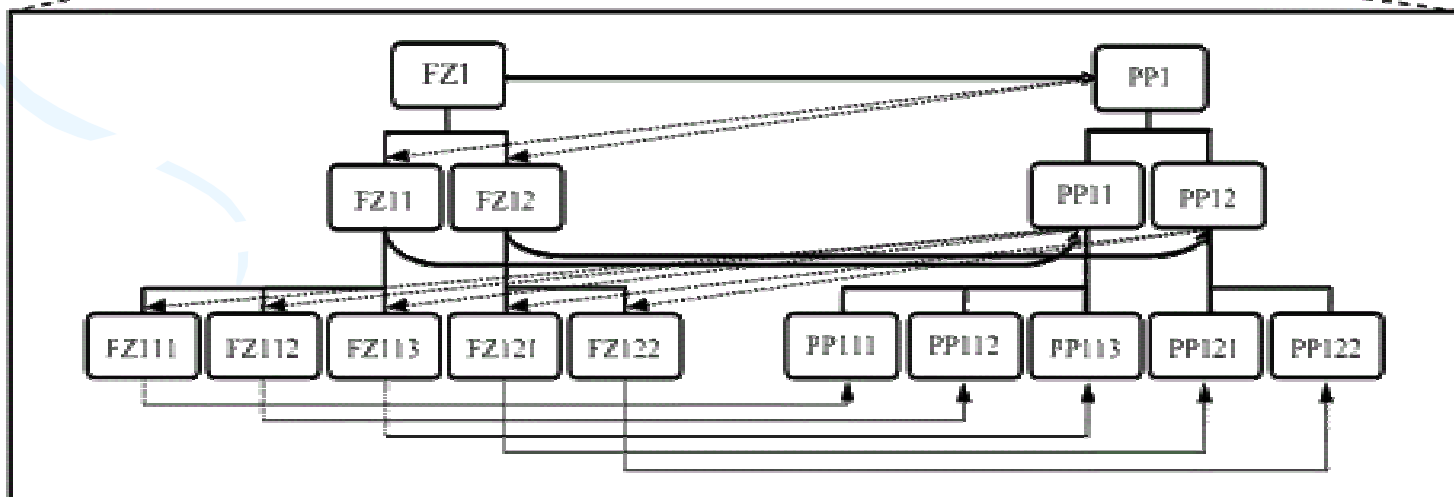
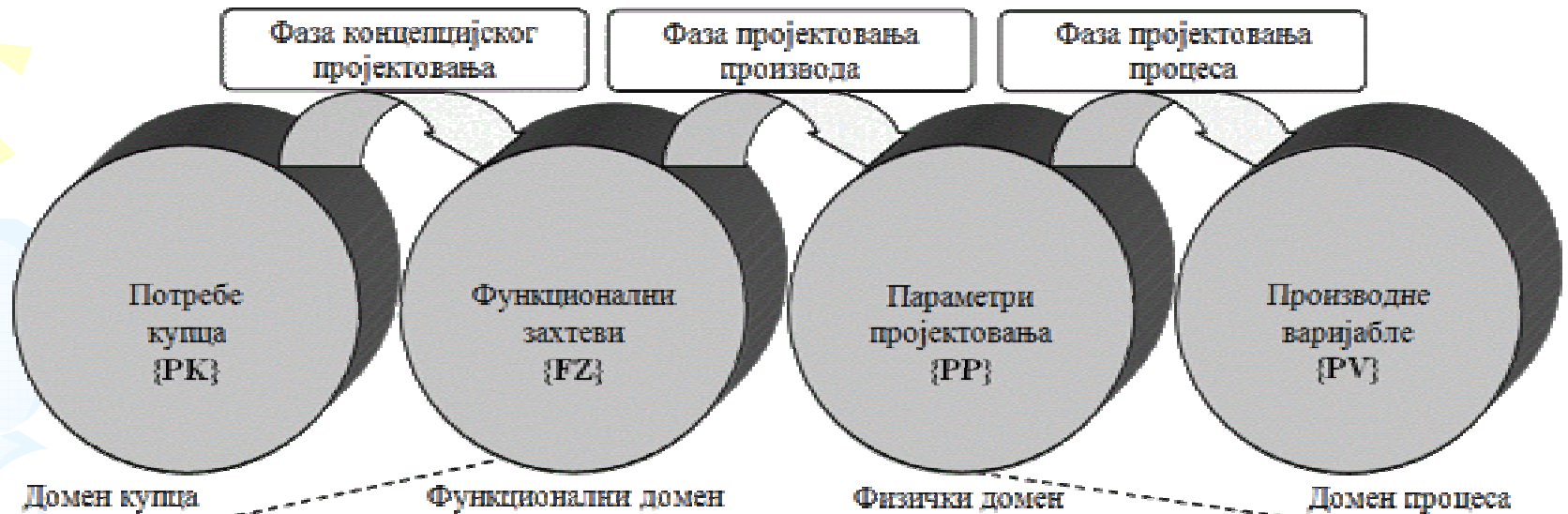
Предметни сарадници: *Најдан Вуковић, дипл. инж. маш.*

Марко Митић, дипл. инж. маш.

асист. Милица Петровић, дипл. инж. маш.

Концепцијско пројектовање

- Аксиоматска теорија пројектовања



Концепцијско пројектовање

- Идентификација функционалних захтева;
- Идентификација параметара пројектовања.

Ф31 Интелигентан транспорт материјала

Ф311-Избегавање препрека

Ф3111-Детекција препрека

Ф3112-Класификација препрека

Ф3113-Заобилажење препрека

Ф312-Оптимизација и учење оптималног технолошког процеса

Ф3121-Оптимизација процеса

Ф3122-Машинско учење технолошких процеса

ПП1 Интелигентни мобилни робот

ПП11-Модул за избегавање препрека

ПП111-Ултразвучни сензор

ПП112-ВНМ за класификацију препрека

ПП113-Алгоритам за заобилажење препрека

ПП12-Модул за оптимизацију и учење оптималног технолошког процеса

ПП121-Теорија графова

ПП122-ВНМ за учење технолошког процеса

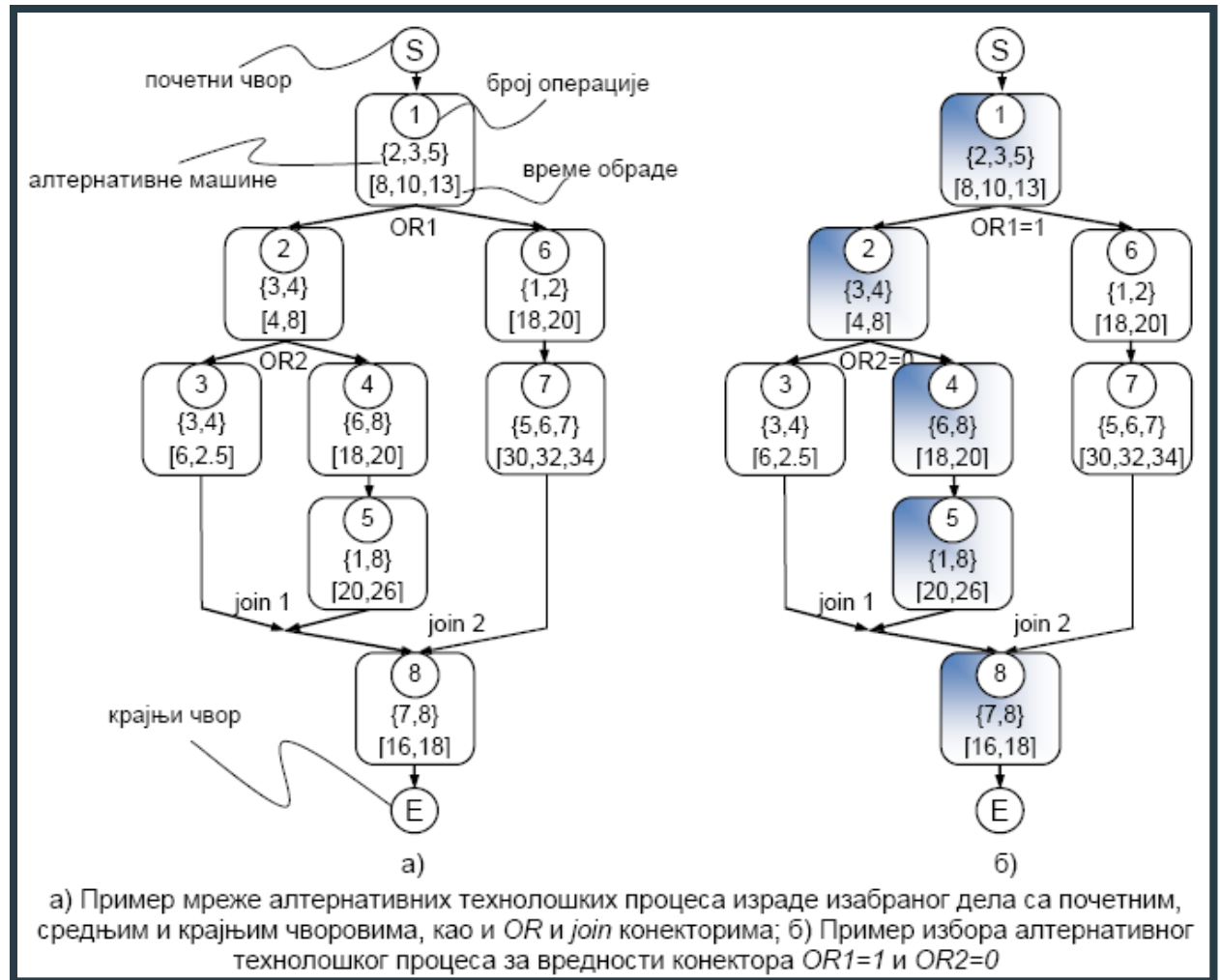
Представљање технолошког процеса

- Врсте чворова:

- Почетни чвор
- Средњи чворови
- Крајњи чвор

- Врсте конектора:

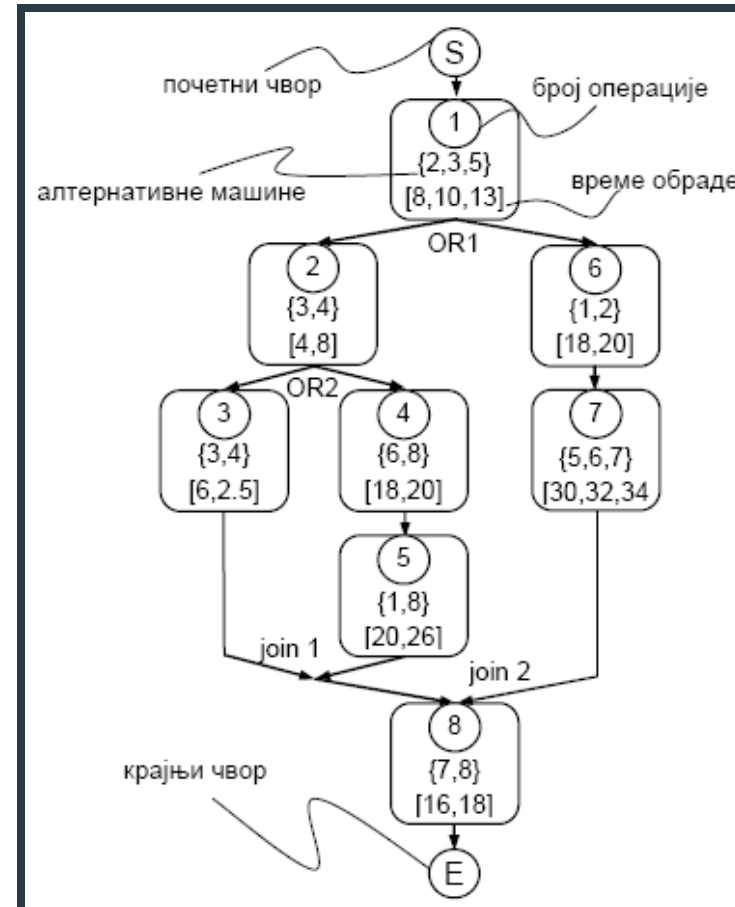
- *OR* конектор – почетак алтернативног технолошког процеса
- *join* конектор - завршетак алтернативног технолошког процеса



Оптимизација технолошког процеса

- Софтвер Matlab – одабир алтернативних машина

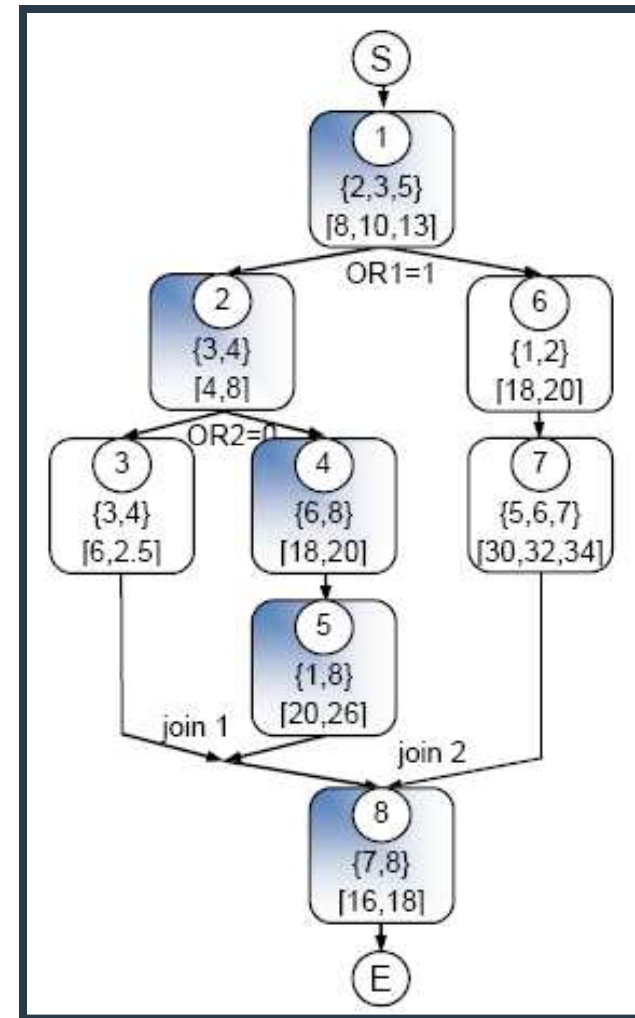
```
OMT = [1 1 1 2 2 3 3 4 4 5 5 6 6 7 7 7 8 8;  
       2 3 5 3 4 3 4 6 8 1 8 1 2 5 6 7 7 8;  
       8 10 13 4 8 6 2.5 18 20 20 26 18 20 30 32 34 16 18];  
n=10;  
operacije = [1 2 3 4 5 6 7 8]; masine = []; OPERACIJE = [];  
for u = 1 : n  
    for i = 1 : size (operacije,2)  
        if operacije(1,i) == 1;  
            t = randnumber(1,3);  
            if t == 1 ch2 = 2;  
            elseif t == 2 ch2 = 3;  
            else ch2 = 5;  
            end  
            masine = [masine ch2];  
        elseif operacije(1,i) == 2;  
            t = randnumber(1,2);  
            if t == 1 ch2 = 3;  
            else ch2 = 4;  
            end  
            masine = [masine ch2];  
        ...  
        else operacije(1,i) == 8;  
            t = randnumber(1,2);  
            if t == 1 ch2 = 7;  
            else ch2 = 8;  
            end  
            masine = [masine ch2];  
        end  
    end  
    operacije = [operacije; masine]  
    OPERACIJE = [OPERACIJE; operacije]  
    masine = []  
    teh_procesi = OPERACIJE  
end
```



Оптимизација технолошког процеса

- Софтвер Matlab – избор алтернативних процеса за различите вредности OR конектора

```
OR1 = []; OR2 = []; n=10;  
for u = 1 : n  
    or1 = randnumber(0,1)  
    if or1 == 0;  
        or2 = 10;  
        operation = [1 6 7 8]  
    else or1 == 1; or2 = randnumber(0,1)  
        if or2 == 1;  
            operation = [1 2 3 8]  
        else operation = [1 2 4 5 8]  
        end  
    end  
    OR1 = [OR1 or1]  
    OR2 = [OR2 or2]  
    or1 = []; or2 = [];  
end
```



Оптимизација технолошког процеса

- Софтвер Matlab – одабир вектора машина и времена

```
machine = [];  
for j = 1 : size(operation,2)  
    l = operation(1,j);  
    ttt = find (operacije == l);  
    MACHINE = OPERACIJE(2,ttt);  
    h = find (operacije == l);  
    machine = [machine MACHINE];  
end
```

```
time = [];  
for k = 1 : size(operation,2)  
    for qq = 1 : size(OMT,2)  
        if (operation(1,k) == OMT(1,qq) ...  
            && machine(1,k) == OMT(2,qq));  
            operation;  
  
            TIME = OMT(3,qq);  
            time = [time TIME];  
        end  
    end  
end
```

```
OMTr = [operation; machine; time];
```

вектор машина M за
одабране операције

вектор времена T за
одабране операције и
машине M

Оптимизација технолошког процеса

- Софтвер Matlab – одређивање функције одлучивања

- Производно време: збир времена обраде дела на машини и времена транспорта дела од машине до машине;

- Функција циља (функција одлучивања) $f(i)$;

$$\max f(i) = \frac{1}{TP(i)}$$

$$TP(i) = \sum_{j=1}^{P_{ij}} TW(i, j, k, l) + \sum_{j=1}^{P_{ij}-1} TT(i, l, (j, k_1), (j+1, k_2)), i \in [1, n], j \in [1, P_{ij}],$$

```
OMTr = [operation; machine; time];
load R; R = round(R);
machine_code = R(1:9,1:9);
TW = sum(time);
TT = [];
for i = 1 : (size(operation,2)-1);
    tt = ...
machine_code(machine(1,i),machine(1,i+1));
    TT = [TT tt];
end
TT = sum(TT);
TP = TW + TT;
TP;
tp = [tp; TP];
FITNESS = 1/TP;
fitness = [fitness; FITNESS];
operation = [];
machine = [];
TW = []; TP = [];
```


Вештачке неуронске мреже у учењу оптималних технолошких процеса

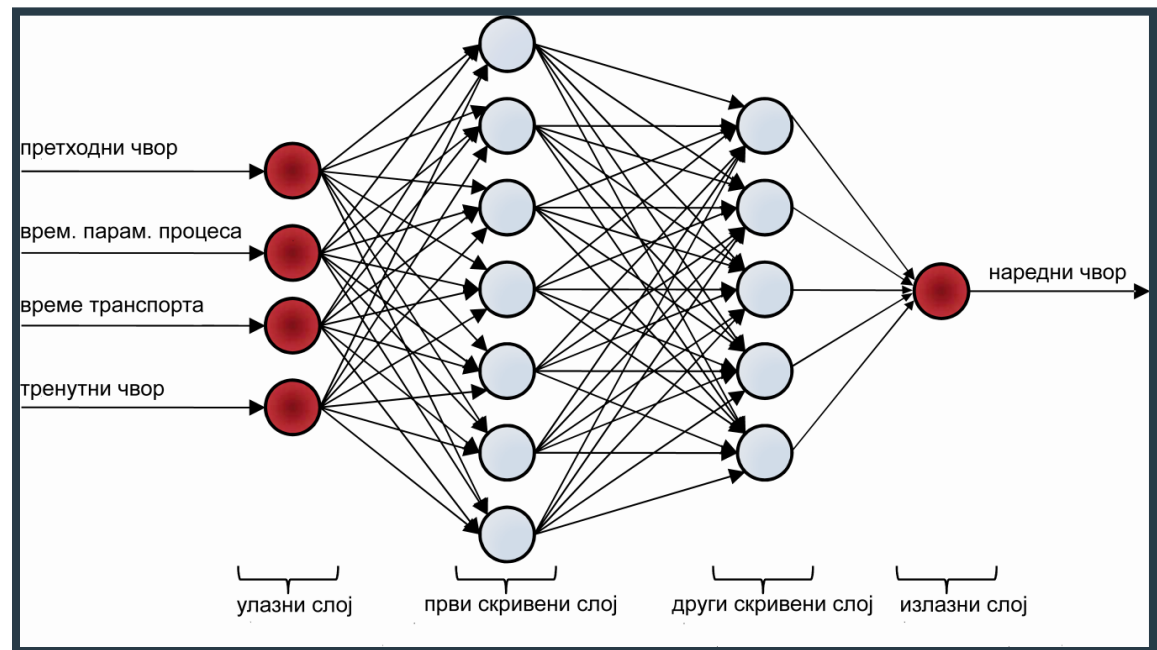
- Дефинисање архитектуре вештачке неуронске мреже:

- **улазни слој:**

- претходни чвор (чвор у ком се робот претходно налазио),
 - временски параметри процеса (укупно време обраде),
 - време транспорта (укупно време кретања робота између машина)
 - тренутни чвор (чвор у ком се робот тренутно налази).

- **излазни слој:**

- наредни чвор путање у који се транспортује део.



Вештачке неуронске мреже у учењу оптималних технолошких процеса

```
clc; clear all; close all;
%=====
% nominalni vremenski parametri
%=====
npn1 = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17]';
mt1 = [20 32 48 97 97 144 144 162 162 175 175 ...
       200 200 220 220 242 242 275]';
tt1 = [79.0000 79.0000 99.0000 121.0711 143.1421 163.1421 ...
       163.1421 211.4264 233.4975 255.5685 275.5685 302.6396 309.7107...
       316.7817 343.8528 370.0660 399.2082 406.2792]';
node1 = [3 3 5 9 5 3 3 8 9 5 3 6 5 6 3 9 6 5]';
%=====
% vremenski parametri # 1
%=====
npn2 = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17]';
mt2 = [25 37 353 402 402 420 420 433 433 458 458 ...
       505 505 525 525 558 558 580]';
tt2 = [79.0000 79.0000 99.0000 121.0711 147.2843 195.5685 ...
       217.6396 239.7107 259.7107 286.7817 293.8528 313.8528 ...
       333.8528 340.9239 340.9239 347.9949 367.9949 394.2082]';
node2 = [3 3 5 9 3 8 9 5 3 6 5 3 5 6 6 5 3 9]`;
...
%=====
% vremenski parametri # 4
%=====
npn5 = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17]';
mt5 = [25 37 53 102 102 149 149 167 167 480 480 ...
       505 505 527 527 560 560 580]';
tt5 = [79.0000 79.0000 99.0000 121.0711 143.1421 ...
       163.1421 163.1421 211.4264 233.4975 255.5685 275.5685 ...
       302.6396 329.7107 355.9239 385.0660 392.1371 392.1371 399.2082]';
node5 = [3 3 5 9 5 3 3 8 9 5 3 6 3 9 6 5 5 6]';
```

Вештачке неуронске мреже у учењу ОПТИМАЛНИХ ТЕХНОЛОШКИХ ПРОЦЕСА

```
%=====
%ULAZI
%=====
previous_nodes = [npn1; npn2; npn3; npn4; npn5];
machining_time = [mt1; mt2; mt3; mt4; mt5];
transportation_time = [tt1; tt2; tt3; tt4; tt5];
node = [node1; node2; node3; node4; node5];

all_inputs = [node previous_nodes machining_time transportation_time];

%=====
%IZLAZI
%=====
node1 = [3 3 5 9 5 3 3 8 9 5 3 6 5 6 3 9 6 5]';
nn1 = [3 5 9 5 3 3 8 9 5 3 6 5 6 3 9 6 5 3]';

node2 = [3 3 5 9 3 8 9 5 3 6 5 3 5 6 6 5 3 9]';
nn2 = [3 5 9 3 8 9 5 3 6 5 3 5 6 6 5 3 9 3]';

node3 = [3 3 5 9 3 8 5 3 3 6 9 5 6 5 3 9 5 6]';
nn3 = [3 5 9 3 8 5 3 3 6 9 5 6 5 3 9 5 6 3]';

node4 = [3 3 5 9 3 8 9 5 5 3 3 6 5 6 3 9 6 5]';
nn4 = [3 5 9 3 8 9 5 5 3 3 6 5 6 3 9 6 5 3]';

node5 = [3 3 5 9 5 3 3 8 9 5 3 6 3 9 6 5 5 6]';
nn5 = [3 5 9 5 3 3 8 9 5 3 6 3 9 6 5 5 6 3]';
next_node = [nn1; nn2; nn3; nn4; nn5];

all_outputs = [next_node zeros(1,size(next_node,1))' ...
zeros(1,size(next_node,1))' zeros(1,size(next_node,1))']];
```

Вештачке неуронске мреже у учењу ОПТИМАЛНИХ ТЕХНОЛОШКИХ ПРОЦЕСА

```
vnm_tehnoski_proces = newff(all_inputs,all_outputs,[8,4],{'tansig','tansig'},'trainlm');

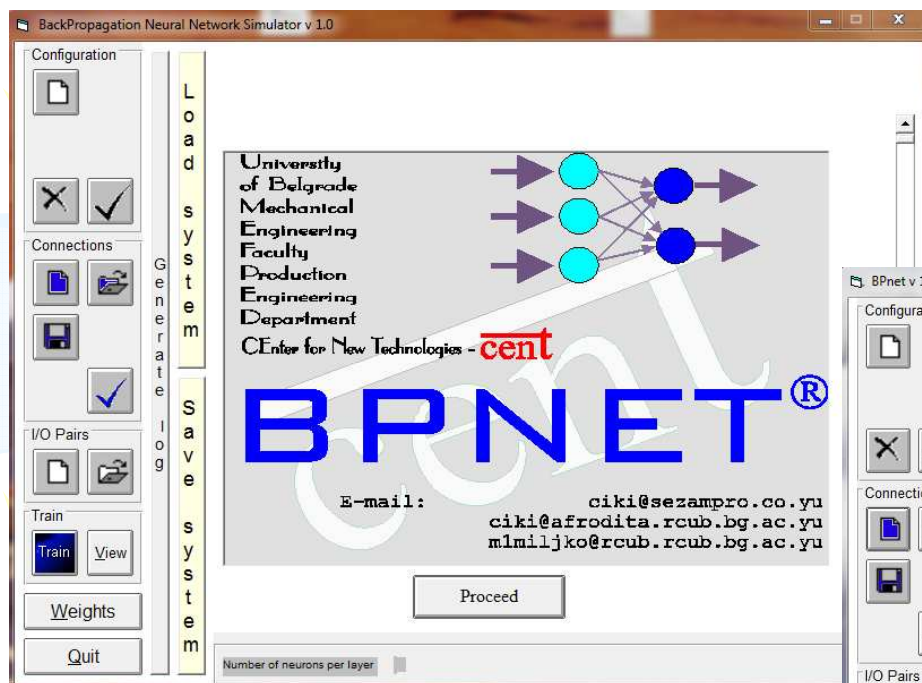
vnm_tehnoski_proces.trainParam.show = 50;
vnm_tehnoski_proces.trainParam.lr = 0.8;
vnm_tehnoski_proces.trainParam.mc = 0.9;
vnm_tehnoski_proces.trainParam.mu = 0.001;
vnm_tehnoski_proces.trainParam.epochs = 1000;
vnm_tehnoski_proces.trainParam.goal = 1e-7;
vnm_tehnoski_proces.trainParam.max_fail = 10;

vnm_tehnoski_proces = train(vnm_tehnoski_proces,all_inputs,all_outputs)
put_sim = sim(vnm_tehnoski_proces,all_inputs)

figure(1)
subplot(5,1,1)
plot(1:18,nn1,'-r*'); hold on
subplot(5,1,2)
plot(1:18,nn2,'-g*'); hold on
subplot(5,1,3)
plot(1:18,nn3,'-b*'); hold on
subplot(5,1,4)
plot(1:18,nn4,'-c*'); hold on
subplot(5,1,5)
plot(1:18,nn5,'-y*'); hold on
figure(2)
plot(put_sim(:,1),'or','MarkerSize',8),hold on
plot(all_outputs(:,1),'b*')
xlabel('Vreme kretanja od cvora do cvora')
ylabel('Cvor')
title('Obucavajuci skup za generisanje putanje mobilnog robota')
h = legend('Nauceni cvor','Zeljeni cvor',1);
```

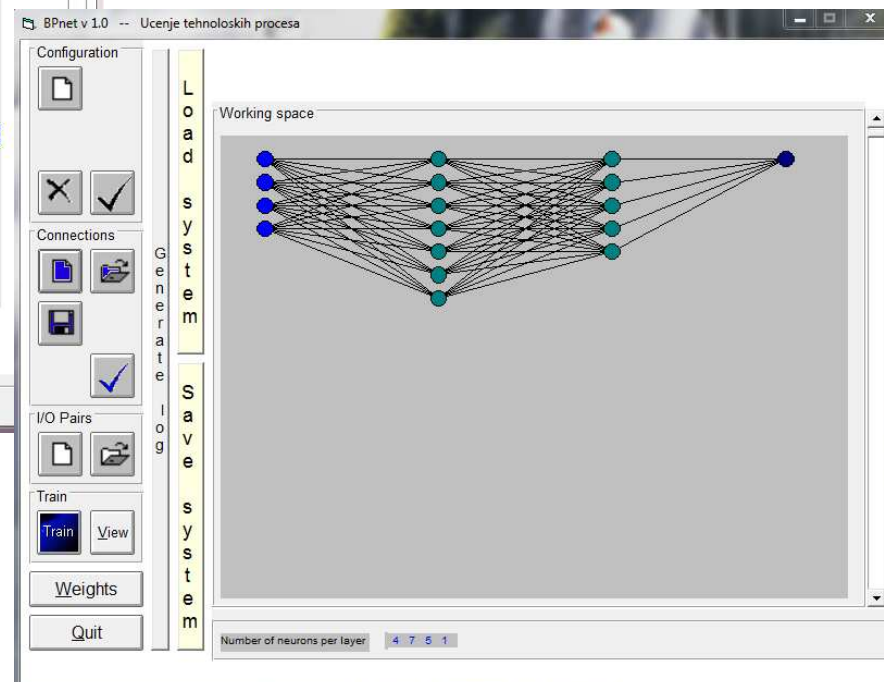
Вештачке неуронске мреже у учењу оптималних технолошких процеса

- Софтвер BPnet



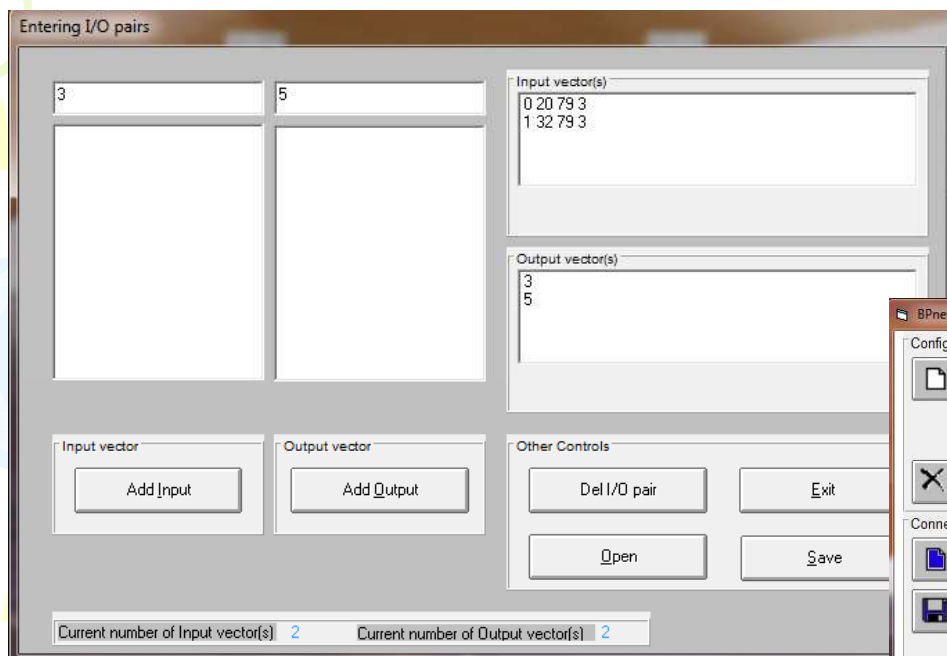
радно окружење софтвера

архитектура формиране ВНМ



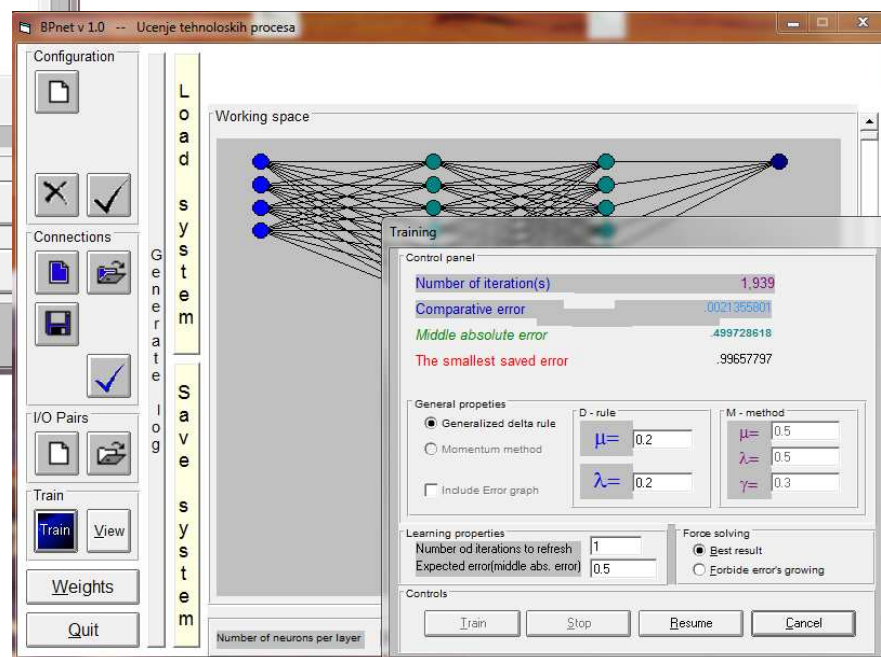
Вештачке неуронске мреже у учењу оптималних технолошких процеса

- Софтвер VPnet



- унос параметара учења и очекиване вредности грешке
- приказ вредности грешке

приказ уношења улазних и излазних вредности вектора – обучавајућих парова





Хвала на пажњи!

Питања?

: Методе одлучивања:

Параметри пројектовања базирани на
оптимизацији токова материјала