

## 1. ОБЛАСТ НА КОЈУ СЕ ТЕХНИЧКО РЕШЕЊЕ ОДНОСИ

Развијени софтвер припада области машинства и индустријског софтвера, а директно се односи на развој и примену вештачке интелигенције (Листа ИР елемената у области развоја софтвера под бр. (7) - вештачка интелигенција; Извор: *Frascati Manual, OECD*), тачније „backpropagation” вештачке неуронске мреже, у домену решавања проблема сензорско-моторне координације робота у технолошком окружењу, предикције квалитета обрађених површина, интелигентног надзора технолошких постројења, итд. и може се уврстити у напредне производне технологије. Проблеми сензорско-моторне координације робота, тј. одређивања положаја енд-ефектора руке робота у односу на објекат у технолошком окружењу, представља неопходну основу за развој интелигентних робота аутономних у погледу континуалног људског надзора, а који у оквиру технолошког окружења могу бити примењени за потребе манипулације деловима, монтаже и сл. С обзиром да се естимација положаја енд-ефектора обавља у реалном времену, било је неопходно да се развије одговарајући софтвер за примену вештачких неуронских мрежа у функцији успостављања сложених неуронских модела сензорско-моторне координације руке робота, а остварене су успешне примене овог софтвера и у домену интелигентних система надзора у области филтрације воде за пиће, затим технолошког препознавања типских технолошких форми машинских делова, као и за предикцију квалитета обрађених површина [12,13,14,15,16]. У овом извештају приказан је развијени софтвер за машинско учење интелигентних система под називом „BPnet V1.0”, остварен у *Visual Basic*<sup>®</sup> окружењу [17]. Развијени софтвер пружа јасан увид у реализацију интелигентног понашања робота или система надзора и указује на тачност спроведене процедуре машинског учења. Софтвер „BPnet V1.0” се такође успешно користи и у додипломској настави у оквиру предмета: *Компјутерска симулација и вештачка интелигенција* (BSc), *Методе одлучивања* (MSc) и *Интелигентни технолошки системи* (MSc), као и за последипломце-магистранте и докторанте у изради магистарских теза и докторских дисертација, а од школске 2008/09. године користи се и на докторским студијама (на енглеском језику) на Машинском факултету у Београду.

## 2. ТЕХНИЧКИ ПРОБЛЕМ

Управљачка стратегија аутономног робота са визуелним сензором-камером подразумева такво процесирање нумеричких података да се информације од система препознавања, уз машинско учење робота преко вештачких неуронских мрежа, користе за остваривање жељеног позиционирања и оријентације енд-ефектора у односу на препознати и идентификовани објекат-радни предмет. Позиција и оријентација објекта у односу на референтни координатни систем робота није позната, али је зато, кроз обучавање робота, позната жељена релативна позиција и оријентација енд-ефектора у односу на објекат. Предложени систем хијерархијског интелигентног управљања, остварује кретање аутономног робота тако да он може несметано да приступи објекту из произвољног иницијалног положаја, уз одговарајуће позиционирање и оријентацију енд-ефектора. Иначе, релације између података о објекту добијених од система препознавања базираног на камери и углова ротације у зглобовима робота вертикалне зглобне конфигурације, за жељено позиционирање и оријентацију енд-ефектора, су изразито нелинеарног карактера. Развијени интелигентни управљачки систем [1] се организује самостално при решавању те нелинеарности, тако што користи способности вештачких неуронских мрежа, за које се зна да су у стању да превазиђу проблем нелинеарних корелација применом познатих алгоритама учења [4,6,7,8,10,11]. Овакав хијерархијски интелигентни управљачки систем директно интегрише визуелне податке-информације у серво-управљање робота.

Концепт управљачке стратегије се заправо односи на проблем препознавања и манипулације препознатих објеката, при чему је познато да су радни предмети потпуно геометријски одређени. Камера, која снима објекте, може да буде фиксно постављена изнад објеката или може да се налази на енд-ефектору робота. За камеру која је постављена на енд-ефектор робота, што је далеко повољнија ситуација (због

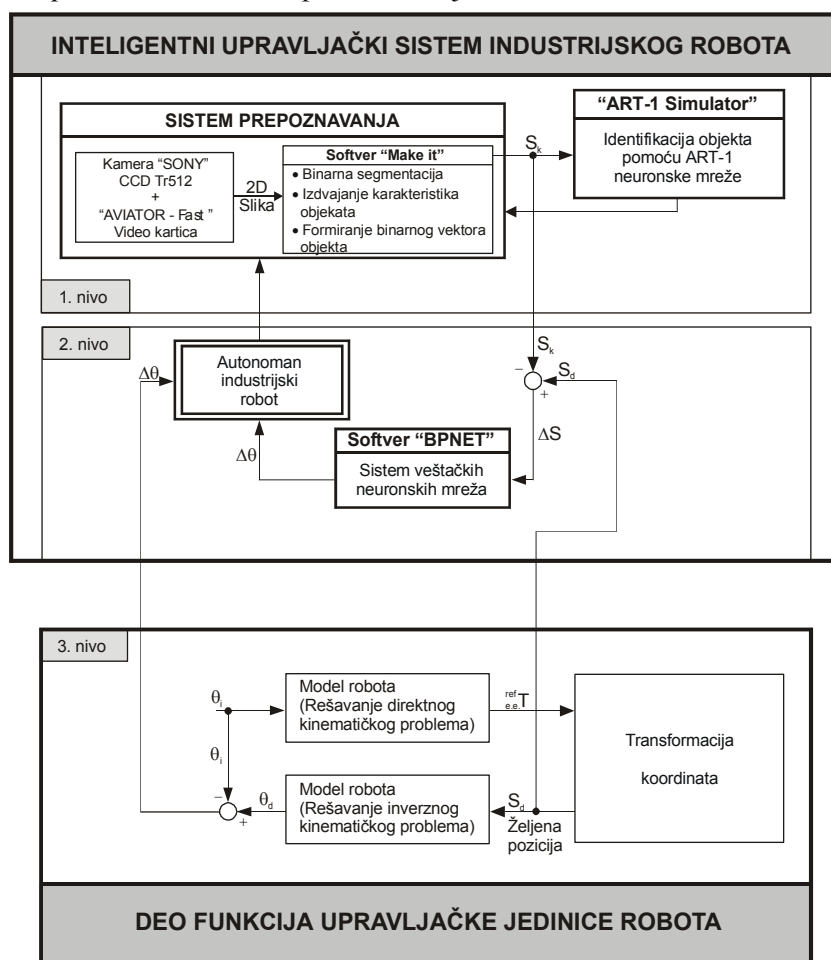
<sup>1</sup> Ванредни професор, Универзитет у Београду-Машински факултет. [zmljkovic@mas.bg.ac.rs](mailto:zmljkovic@mas.bg.ac.rs)

<sup>2</sup> Истраживач-сарадник, „ECOLAB Hygiene“, д.о.о., Београд. [atlant@ptt.rs](mailto:atlant@ptt.rs)

<sup>3</sup> Редовни професор, Универзитет у Београду-Машински факултет. [bbabic@mas.bg.ac.rs](mailto:bbabic@mas.bg.ac.rs)

могућности подешавања фокалног растојања и делимичног решавања проблема дисторзије слике), важи да се оптичка оса камере поклапа са  $z$ -осом локалног координатног система који је везан за врх (TCP-tool center point) робота [1].

Развијени интелигентни управљачки систем подразумева да је покрет робота детерминисан коришћењем информација о објектима добијеним од камере и система вештачких неуронских мрежа и заснива се на два процедурама које су приказане блок-дијаграмом на слици 1. Прва процедура подразумева коришћење дела функција постојеће управљачке јединице (нпр. MITSUBISHI-MOVEMASTER-EX робота), са надограђеним софтвером за рад у *Windows*<sup>®</sup> окружењу [1], а друга представља квалитативно виши ниво управљања јер користи интелигентно управљање базирано на систему вештачких неуронских мрежа. Хијерархијска структура интелигентног управљања робота је организована тако да се у оквиру ове две процедуре могу уочити три хијерархијска нивоа (слика 1). Да би се боље разумела успостављена хијерархијска управљачка стратегија треба нагласити да се, после анализе слике објеката и издвајања њихових карактеристика помоћу развијеног система препознавања [1], врши идентификација тих објекта коришћењем ART-1 неуронске мреже (Adaptive Resonance Theory) на првом нивоу, а затим се систем BP неуронских мрежа („backpropagation”) примењује за нелинеарна пресликавања ОБЈЕКАТ  $\Rightarrow$  енд-ефектор РОБОТА на другом нивоу. У току машинског учења робота користи се део функција постојеће управљачке јединице (трећи ниво), како би се кроз обучавање робота одредиле неопходне унутрашње координате (углови ротације  $\theta_i$  у зглобовима) за успешно позиционирање и оријентацију енд-ефектора робота релативно у односу на препознати и идентификовани објекат.

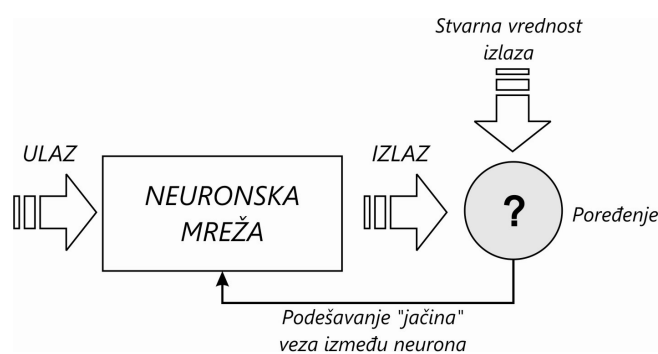


Слика 1. Блок-дијаграм хијерархијског интелигентног управљања аутономног робота

### Теоријске основе проблема

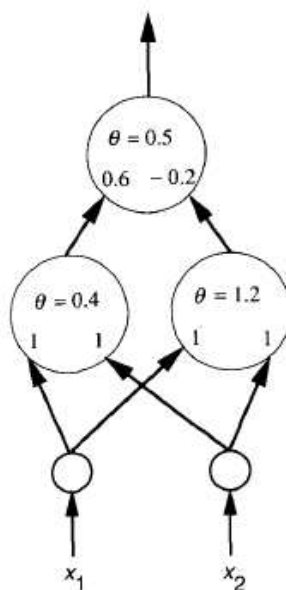
Вештачке неуронске мреже представљају моћан, паралелно повезани процесор који има „природну” способност ка прикупљању знања из експерименталних података и коришћењу тих знања слично људском мозгу [1,2,4,6,7]. Ово је посебно важно због тога што ниво знања који мрежа поседује, а који је добијен кроз процес њеног обучавања, односно кроз итеративно одређивање неуронских јачина веза у мрежи,

представља еквивалент прикупљеног знања (слика 2.). Када је вештачка неуронска мрежа обучена, могуће је тестирати њен рад довођењем на улаз нових, непознатих улазних величина са циљем „генерисања” излаза, што је потпуно аналогно раду људског мозга. За разлику од дигиталних рачунара, где је прорачунавање централизовано, серијско и синхронизовано, у вештачким неуронским мрежама прорачунавање је заједничко, паралелно и несинхронизовано. Како је то на слици 2. и показано, основни проблем у раду са вештачким неуронским мрежама које раде са „*backpropagation*” алгоритмом је везан за квалитет и брзину подешавања тежинских односа у мрежи [2]. Вештачке неуронске мреже, као прорачунске методе [1,2], способне су да науче и генерализују природу појединих феномена на основу познатих експерименталних резултата. Оне могу бити погодне за моделирање и предвиђање промене функционалних карактеристика посматраних система или процеса [1,2], с обзиром да могу бити обучаване да нађу решење, препознају моделе понашања, класификују податке и предвиде будуће догађаје. Адаптивно процесирање података помоћу вештачких неуронских мрежа може бити искоришћено за развој тзв. неуронских модела. Основна предност примене вештачких неуронских мрежа у овој области је та што се оне могу применити у ситуацији када су узроци појаве одређених феномена непознати, а сходно томе и ефекти које ти феномени изазивају. То заправо значи да се вештачке неуронске мреже могу применити у ситуацијама када имамо непотпуне и/или нејасне информације о одређеним појавама и њиховим узрочно - последицим везама.



Слика 2. Принцип рада вештачких неуронских мрежа

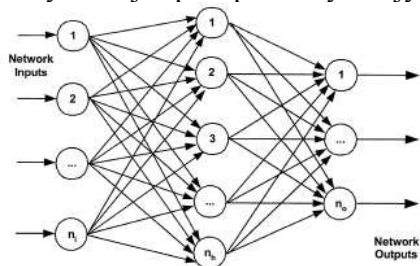
Вештачке неуронске мреже свој почетак везују за крај 1950.-их година, када се уводе појмови *перцептрона* (Frank Rosenblatt) и „*Adaline*” неуронских модела (*Adaptive Learning - Linear Neuron*) [3]. Перцептрон је представљен као парадигма у домену области сензорисања, обраде сигнала и одговора на њихове утицаје. Идеја, која се крила иза саме парадигме, подразумевала је да „надражаји” различитих интензитета долазе у сваки перцептрон, те да ниво „надражаја” (преко функције преноса) дефинише стање самог перцептрона. Анализа (Minsky и Seymour, 1969. године) је показала да систем перцептрона може да решава оне задатке код којих је могуће спровести само линеарно раздвајајуће класе узорака (слика 3.), а за сложеније је неопходно развити вишеслојне неуронске мреже.



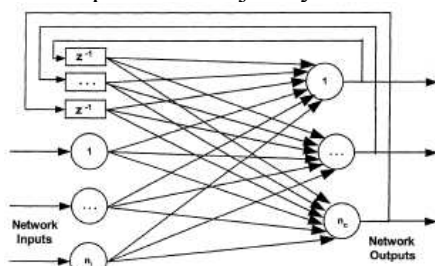
Слика 3. Модел перцептрона

## Вишеслојне Неуронске Мреже (ВНМ)

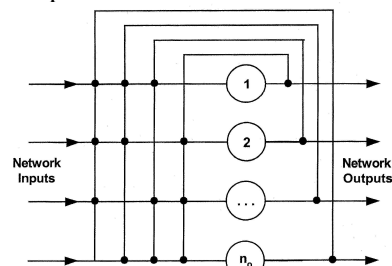
Данас су ВНМ најзаступљеније (слика 4.) [9]. Свака мрежа се састоји од више слојева неурона. Неурони у слоју нису међусобно повезани. Сваки неурон из средњег слоја је повезан са сваком неуроном из претходног и наредног слоја. Ове мреже имају пропацију података/сигнала унапред. Ову мрежу одликује крајња универзалност величине улазних података. Уз адекватно пројектно решење неуронских модела, могуће их је тренирати обучавајућим векторима код којих су чланови веома различити.



Слика 4. ВНМ структура



Слика 5. Рекурентна НМ



Слика 6. Хопфилдова РНМ

## РНМ – рекурентне неуронске мреже (Recurrent Neural Networks)

Код ових врста мрежа, неурони у улазном слоју добијају информације како из окружења, тако и од неурона из наредних слојева, тј. ово су тзв. мреже са пропацијом података и унапред и уназад (слика 5). Најпознатији систем ових врста мрежа је Хопфилдова рекурентна ВНМ (слика 6). Код ове мреже су сви неурони међусобно повезани. Поменута структура може да запамти (асоцијативна меморија) неколико узорака општих стања, тако да уз малу присутност шума (нека величина стања није упамћена), лако конвергира ка решењу. Недостатак се огледа у томе што ради само са биполарним улазним вредностима.

Ослањајући се на теоријске поставке дате у [1,2], у оквиру развијеног софтвера **BPnet V1.0**, уведен је ефикасан начин одређивања градијента функције циља у односу на улазни вектор – „backpropagation” поступак. Овај начин одређивања градијента функције циља базиран је на примени правила одређивања извода сложене функције [2,4,6,11]. Нека је дата континуална, непрекидна и диференцијабилна функција  $z=g(y)$ . Уколико усвојимо да је  $y=f(x)$  тада је извод функције  $g(\cdot)$  по променљивој  $x$ :

$$\frac{\partial g}{\partial x} = \frac{\partial g}{\partial y} \frac{\partial y}{\partial x} \quad (1)$$

Пре него што уведемо овај алгоритам учења неопходно је да уведемо и следећу нотацију. Уколико сходно основној идеји моделирања проблема применом вештачких неуронских мрежа уведемо нову функцију  $v$  као:

$$v = \sum_{j=1}^n w_{ij} x_j \quad (2)$$

и назовемо је потенцијалом  $i$ -тог неурона, онда „излаз” из вештачке неуронске мреже можемо дефинисати на следећи начин:

$$z = \text{net}(X^{(k)}, W) = f(v) \quad (3)$$

Сходно правилу одређивања извода сложене функције, као и на основу ново-уведене величине  $v$ , извод функције циља  $J(W)$  може бити одређен у следећој форми:

$$\frac{\partial J(W)}{\partial w_{ij}} = \frac{\partial J(W)}{\partial v_i} \frac{\partial v_i}{\partial w_{ij}} \quad (4)$$

где су:

$$\frac{\partial J(W)}{\partial v_i} - \text{извод функције циља по потенцијалу } i\text{-тог неурона,}$$

$\frac{\partial v_i}{\partial w_{ij}}$  - промена потенцијала  $i$ -тог неурона у односу на тежинске коефицијенте  $w_{ij}$ .

Након увођења ове нове величине у могућности смо да дефинишемо градијент функције циља као производ извода сложене функције у следећем облику:

$$\frac{\partial J(W)}{\partial w_{ij}} = \frac{\partial J(W)}{\partial \varepsilon_j} \frac{\partial \varepsilon_j}{\partial z_j} \frac{\partial z_j}{\partial v_j} \frac{\partial v_j}{\partial w_{ij}} \quad (5)$$

Да бисмо могли да одредимо градијент функције циља  $J(W)$  по тежинским коефицијентима  $w_{ij}$  неопходно је да анализирамо и сваки појединачни извод у једнакости (5). Први извод  $\frac{\partial J(W)}{\partial \varepsilon_j}$  се крајње једноставно одређује као:

$$\frac{\partial J(W)}{\partial \varepsilon_j} = \varepsilon_j \quad (6)$$

Други члан у једначини (5) представља извод вектора резидуала  $\varepsilon_j$  по функцији  $z$  која представља „коначан излаз” из вештачке неуронске мреже. Дакле, с обзиром да је резидуал дефинисан са  $\varepsilon = y - \text{net}(x, W)$ , следи да је извод:

$$\frac{\partial \varepsilon_j}{\partial z_j} = -1 \quad (7)$$

Трећи извод представља промену излазне вредности вештачке неуронске мреже у односу на сваки потенцијал  $v_j$ . Другим речима:

$$\frac{\partial z_j}{\partial v_j} = f'(v_j) \quad (8)$$

На крају, последњи извод је:

$$\frac{\partial v_j}{\partial w_{ij}} = x_j \quad (9)$$

одређен на основу једнакости (2). На основу свих једнакости градијент функције циља је:

$$\frac{\partial J(W)}{\partial w_{ij}} = -\varepsilon_j f'(v_j) x_j \quad (10)$$

С обзиром да на овај начин одређени градијент функције циља (10) можемо директно увести у алгоритам најмањег градијента, следи да се модификација тежинских коефицијената одвија сходно следећој једнакости:

$$W_{j+1} = W_j - \mu \nabla J = W_j - \mu \frac{\partial J(W)}{\partial w_{ij}} = W_j + \mu \varepsilon_j f'(v_j) x_j \quad (11)$$

Дакле, као што се из ове анализе може видети, једнакост (10) представља ефикаснији начин одређивања градијента функције циља у поређењу са директним методама као што су градијентни поступак или Левенберг-Маркеов алгоритам [11]. На крају, важно је нагласити да простирање грешке уназад (познатије као „*backpropagation*”) није суштински алгоритам обучавања [8], што се може видети из једнакости (11), зато што се приликом примене израза (10) не одређују тежински односи већ *искључиво* градијент функције циља. Да би се одредили тежински коефицијенти неопходно је применити градијентни поступак у коме ће се преко „*backpropagation*” алгоритма израчунавати градијент функције циља.

### 3. ФОРМИРАЊЕ ВИШЕСЛОЈНИХ НЕУРОНСКИХ МРЕЖА - ПОСТОЈЕЋЕ СТАЊЕ У СВЕТУ

Вештачке неуронске мреже се састоје од елемената за процесирање података (информација)/сигнала груписаних у слојеве [1,2], а одликују се:

1. Великим бројем процесирајућих елемената слично правим неуронима.
2. Великим бројем веза између неурона са различитим тежинским односима у којима се креира знање акумулирано током обучавања мреже.
3. Принципом паралелног процесирања информација.

У начелу, постоје четири фазе у стварању вештачке неуронске мреже [2]:

1. Спровођење експеримента (неопходно је прикупити улазне величине за обучавање и тестирање мреже).
2. Пројектовање мреже (дефинисање шта је то што ће бити обрађено кроз мрежу, број слојева у мрежи, функције преноса, алгоритам учења, итд.).
3. Повезивање изабраног броја неурона у сваком од слојева.
4. Имплементација (претварање улазних величина у жељени формат, претпроцесирање података, тестирање рада мреже).

Да би вештачка неуронска мрежа извршила неки задатак потребно је повезати неуроне у одговарајућу конфигурацију, поставити тежинске односе између неурона и изабрати улазно - излазну функцију. Најједноставнији облик вештачке неуронске мреже се састоји од улазног слоја, који је повезан са средишњим или скривеним слојем, и излазног слоја. Основни принцип рада мреже подразумева да се расположиви подаци прослеђују улазном слоју мреже, с обзиром да вредности сигнала које примају неурони у скривеном слоју зависе од модификације улазних сигнала, а представљени су јачинама веза између неурона, као и тиме на који начин ће тако формиран улаз у неурон бити промењен избором функције преноса у неурону. Оптималан број скривених неурона зависи од много фактора, броја улаза и излаза, броја обучавајућих парова, величине шума у обучавајућим паровима, сложености функције грешке, архитектуре мреже и алгоритма обучавања [4,6,7,8,10]. Постоје различити предлози како одредити број неурона ( $n$ ) у скривеним слојевима:

$$n \in [1, N_y]$$

$$n = \frac{2(N_x + N_y)}{3}, \text{ две трећине од збира броја улаза и броја излаза,}$$

$$n < 2N_x, \text{ два пута мање од броја улаза,}$$

$$n = \sqrt{N_x \cdot N_y}, \text{ квадратни корен из производа броја улаза и излаза.}$$

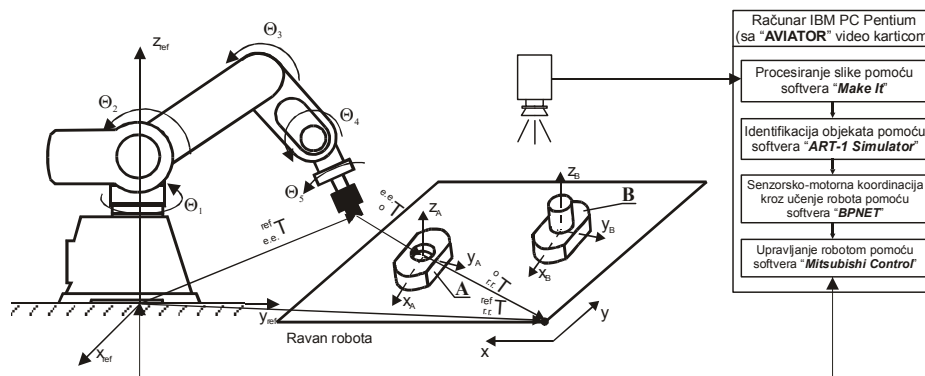
Ипак, најбољи приступ за проналажење броја неурона у скривеним слојевима је обучавање већег броја мрежа са различитим бројем неурона, уз постепено повећавање броја неурона до достизања жељеног резултата. У пракси, тежински односи између неурона се подешавају за вредност која је пропорционална степену промене грешке на излазу. Ова величина се зове извод грешке тежинског односа и често ју је врло тешко израчунати. Minsky и Papert су 1969. године показали да двослојна неуронска мрежа са простирањем сигнала унапред може превазићи многе проблеме и ограничења у раду неуронских мрежа, али нису дали решење како подесити тежинске односе у мрежи. Одговор на ово питање су дали Rumelhart, Hinton и Williams 1986. године, мада су се слична решења појавила још раније (Werbos 1974; Parker 1985; Cun 1985) [1,2,6,7,8]. Основна идеја ових решења се огледа у томе да се грешке које чине процесирајући елементи у скривеним слојевима мреже одређују простирањем грешака уназад, и то оних које су настале у њеном излазном слоју. Због тога се овај метод често зове „*backpropagation*” правило учења које се још назива и генерализано делта правило за нелинеарну активациону функцију и вишеслојне неуронске мреже. Ово решење је било предложено још 1974. године од стране Paul Werbos-а и тај метод подешавања тежинских односа у мрежи он је назвао „*backpropagation* - простирање уназад”.

### 4. СУШТИНА ТЕХНИЧКОГ РЕШЕЊА – ПРИМЕНА И ПРИКАЗ СОФТВЕРА „VPnet V1.0”

На основу развијеног концепта хијерархијског интелигентног управљања (слика 1.) за аутономне индустријске роботе, реализован је интегрисани хардверско-софтверски експериментални систем. Чине га: едукациони индустријски робот MITSUBISHI-MOVEMASTER-EX<sup>1)</sup>, камера Sony CCD TR512E, IBM PC

<sup>1)</sup> Овај едукациони индустријски микро-робот (модел RV-M1) произвела је Mitsubishi Electric Корпорација.

рачунарска платформа опремљена *FAST-AVITOR*<sup>2)</sup> - картицом за аквизицију сигнала од камере и софтвери „*Make it*”, „*ART Simulator*”, „*BPnet*” и „*MITSUBISHI Control*” (слика 7.) [1].



Слика 7. Експериментални систем робота *MITSUBISHI-MOVEMASTER-EX*

На слици 7. је показана свеобухватна процедура (са блок-дијаграмом) по којој је експеримент изведен. Наиме, фиксно постављена камера снима објекте и генерише се 2D слика. Развијени софтвер „*Make it*” после процесирања 2D слике, коришћењем поступка сегментације преко региона [1], издваја карактеристике објеката и формира бинарни вектор којим се представља препознати објекат. Након тога се реализује идентификација објеката применом развијеног софтвера „*ART Simulator*”, који на основу генерисаног бинарног вектора поступком поређења, односно на основу сличности са објектом који се „тражи”, идентификује објекат коме робот треба да приступи. Када је поступак идентификације објекта завршен и када је позната његова позиција и оријентација у унапред познатој равни радног простора индустријског робота, применом система вештачких неуронских мрежа, односно коришћењем софтвера „*BPnet*” сопственог развоја, као и софтвера „*MITSUBISHI Control*” за надзор и управљање едукационим индустријским роботом „*MITSUBISHI-MOVEMASTER-EX*”, реализује се сензорско-моторна координација енд-ефектора робота при манипулацији објекта.

Кроз обучавање робота и процес машинског учења одређују се позиција и оријентација енд-ефектора робота релативно у односу на положај објекта коме у датом тренутку треба да приступи. Ако се упореди слика 7. са сликом 1. уочава се да је број хомогених трансформација смањен елиминацијом оних које су најкомплесније (између објекта и камере, као и камере у односу на енд-ефектор робота), јер предложена и развијена управљачка стратегија хијерархијског интелигентног управљања индустријским роботом то омогућава [1]. Практично је, само при обучавању робота да приступи објекту са одговарајућим позиционирањем и оријентацијом енд-ефектора, односно приликом одређивања унутрашњих координата (углови  $\theta_i, i=1,2,\dots,5$ ), коришћен део управљачких функција постојеће управљачке јединице робота „*MITSUBISHI-MOVEMASTER-EX*” (слика 1.). На тај начин су, за различите положаје објекта у изабраној равни радног простора робота, одређене унутрашње координате робота, како би се спровео процес машинског учења робота помоћу развијеног софтвера „*BPnet*”.

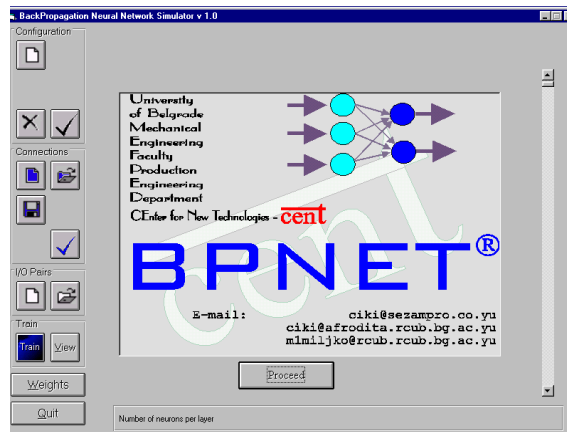
Управљачка стратегија подразумева да је покрет робота детерминисан коришћењем информација о објектима од камере, а које се укључују у процес машинског учења система вештачких неуронских мрежа, како би се остварила сензорско-моторна координација при реализацији технолошког задатка манипулације препознатим објектом. За реализацију те сензорско-моторне координације, у погледу учења робота да успешно приступи идентификованом објекту, развијен је софтвер „*BPnet*” у Лабораторији за индустријску роботiku и вештачку интелигенцију [1,2] Катедре за производно машинство Машинског факултета у Београду, који се такође користи и за калибрацију камере. Овај софтвер сопственог развоја је проблемски оријентисан и остварен је програмирањем у *Visual Basic*<sup>®</sup>-у [17] у складу са развијеним софтверима за примену вештачких неуронских мрежа [2,5]. Приликом реализације софтвера примењен је описани алгоритам учења за „*backpropagation*” (BP) вештачку неуронску мрежу.

Стартовањем софтвера „*BPnet*” појављује се основни прозор који је приказан на слици 8. Види се да је комуникација корисника овог програма са самим софтвером уобичајена за рад у *Windows*<sup>®</sup> оперативном систему. Отварање појединих прозора иде сукцесивно, у складу са потребама корисника у погледу генерисања топологије мреже пре свега, што је приказано на слици 9, а касније и свих осталих опција.

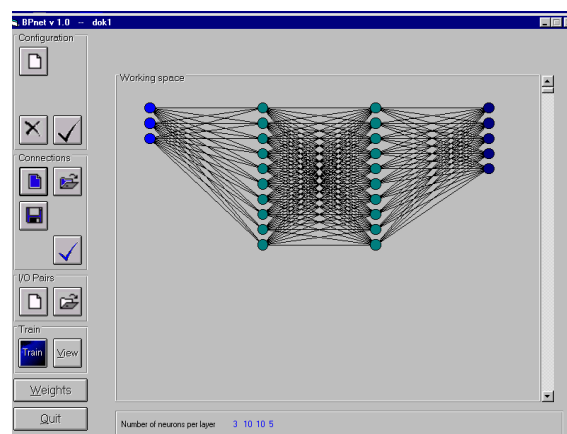
<sup>2)</sup> Произвођач картице је *FAST Electronic GmbH*, чија је *Internet* адреса: <http://www.fastmultimedia.com>.



Топологију мреже бира корисник, а у току реализације спроведених експеримената најбоље резултате је дала четворослојна мрежа [1].



Слика 8. Основни прозор софтвера „BPnet”

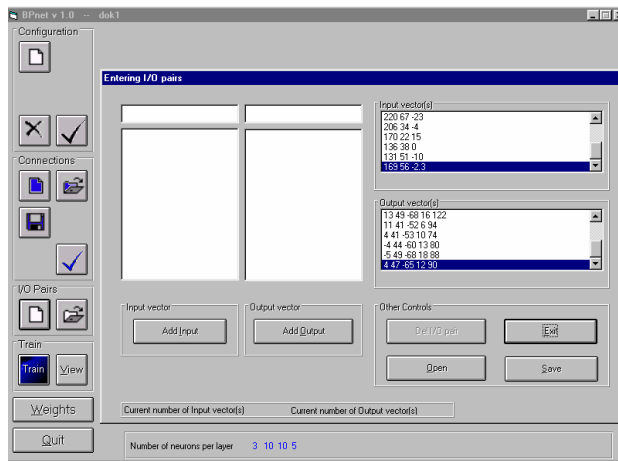


Слика 9. Одређивање изабране топологије ВР неуронске мреже

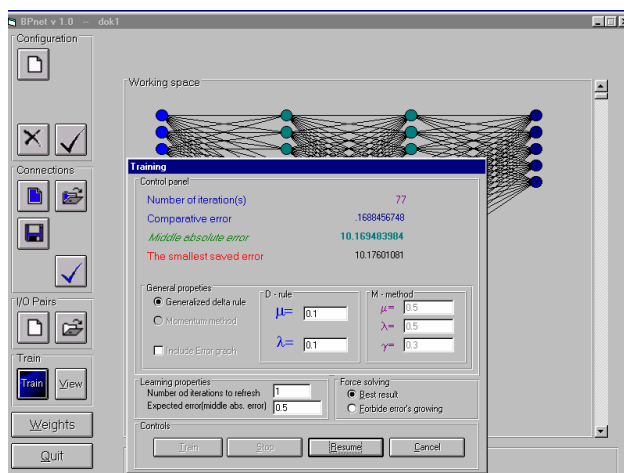
Затим корисник, помоћу опције „BPnet”-а везане за унос обучавајућих парова ВР неуронске мреже, дефинише улазно-излазне облике узорака, на основу којих ће се мрежа обучавати по описаном алгоритму учења. Прозор који показује ту опцију софтвера је приказан на слици 10. После успешно унетих парова за обучавање ВР неуронске мреже и дефинисаних иницијалних тежинских односа између неурона у суседним слојевима мреже, као и *bias*-а [1,2], корисник активира процедуру за стартовање процеса обучавања. Она подразумева избор одговарајућег параметра учења, као и жељену тачност, односно дозвољену грешку по завршетку обучавања мреже. Поред тих основних задавајућих параметара, корисник може да дефинише и учесталост „освежавања” бројача итерација које се у току обучавања ВР мреже остварују. Овај веома битан део софтвера „BPnet” је приказан на слици 11.

У току обучавања ВР неуронске мреже могуће је зауставити сам процес учења, јер софтвер „BPnet” то обезбеђује. Понекад је значајно проверити да ли је конвергенција, односно минимизација грешке при учењу, одговарајућа. Показало се да је чест случај тзв. „претренирања” мреже, што заправо значи да се непотребно од ВР мреже тражи, да већ сасвим добро пресликавање из улазног у излазни простор додатно, кроз увећан број итерација, дотерује. „BPnet” обезбеђује и визуелно јасан приказ тренутних вредности тежинских односа, као и тренутни ниво обучености. Могуће је проверити колико одступају тренутне вредности од жељених, у излазном слоју ВР неуронске мреже. Ако су те вредности у задовољавајућем опсегу који се толерише у конкретној апликацији, онда се даље обучавање прекида, а достигнути ниво обучености ВР неуронске мреже меморише у одговарајућем фајлу помоћу *Save* опције софтвера „BPnet”. Прозор, који илуструје ове корисне могућности софтвера „BPnet”, дат је на слици 12. Види се да је софтвер реализован тако да је корисникова комуникација са опцијама максимално олакшана, јер је кроз више верзија у току стварања овог програма уочено колико је она битна [2,14]. Како је овај софтвер практично намењен било којој апликацији која има потребу за коришћењем резултата обучавања ВР вештачке неуронске мреже, то се уложени труд на вишегодишњем дотривању софтвера „BPnet” може валоризовати кроз широки спектар досадашње [12,13,14,15,16], а и будуће мултидисциплинарне примене.

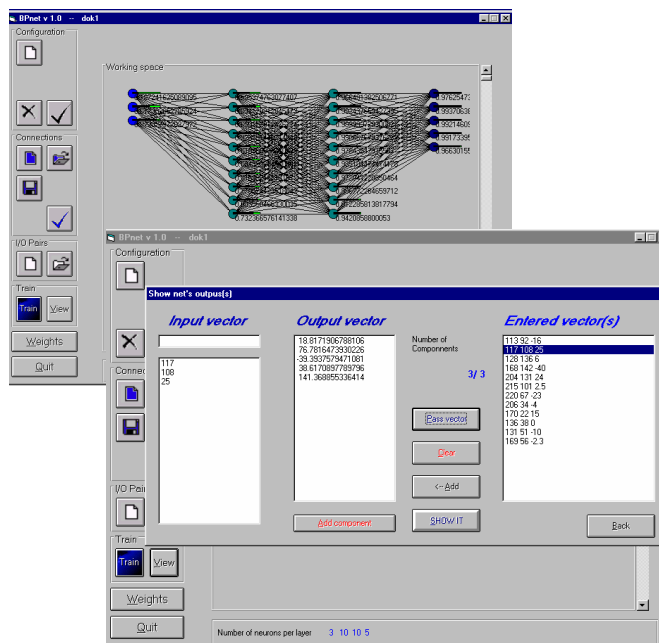




Слика 10. Унос улазно-излазних парова за обучавање ВР неуронске мреже



Слика 11. Процес обучавања ВР неуронске мреже – основни прозор



Слика 12. Праћење остварених резултата током обучавања ВР неуронске мреже

## 5. ЗАКЉУЧАК

Проблем сензорско-моторне координације робота привукао је велику пажњу научника крајем прошлог и почетком овог века. Основни разлог популарности проблема је наведена могућност примене потпуно аутономних робота у различитим технолошким окружењима у којима је рад у континуитету за раднике исувише ризичан (рудници, нуклеарке, и сл.), или пак у оним окружењима у којима примена ових машина може побољшати постојећи рад унутар великих складишта или производно оријентисаних компанија.

У оквиру пројекта ТР-14031 [18] разматрана је интеграција робота у производно оријентисаном окружењу у циљу побољшања транспорта и манипулације материјала, сировина и готових делова [2]. Даљи развој индустријских робота, са циљем очекиване примене за потребе манипулације сировина, материјала и готових делова у оквиру интелигентног технолошког система производно оријентисане компаније, омогућио би ефикасније обављање основних технолошких задатака погодних за роботизацију. Познавање сопственог положаја, али и положаја карактеристичних објеката у окружењу, омогућило би виши степен аутономности индустријских робота у погледу свих технолошких задатака које треба извршити.

Развој софтвера „**BPnet V1.0**” за машинско учење интелигентних система је управо имао претходно речено за циљ, а омогућује и једноставније, као и ефикасније тестирање нових неуронских модела за многе корисне примене [13,14,15,16]. У том смислу, софтвер „**BPnet V1.0**” обезбеђује естимацију тражених излазних величина, остварених после машинског учења, након чега корисник може упоредити перформансе интелигентног система за задате почетне услове и извршити анализу резултата учења.

На крају овог извештаја, важно је истаћи остварену примену развијеног софтвера „**BPnet V1.0**” у образовне сврхе, као помоћног наставног средства за извођење лабораторијских вежби, на свим нивоима академских студија *Модула за производно машинство* Машинског факултета у Београду.

## ЛИТЕРАТУРА

- [1] Миљковић,З.: Системи вештачких неуронских мрежа у производним технологијама, Монографија, Универзитет у Београду - Машински факултет, Београд, 2003.
- [2] Миљковић,З., Александрић,Д.: Вештачке неуронске мреже - збирка решених задатака са изводима из теорије, Универзитет у Београду - Машински факултет, Београд, 2009.
- [3] Widrow B., Lehr M., (1990) 30 Years of Adaptive Neural Networks: Perceptron, Madaline, and Backpropagation, - Proc. IEEE Vol. 78, 9.
- [4] Freeman J. A., Skapura D. M.: Neural Networks – Algorithms, Applications and Programming Techniques, - Addison–Wesley Publishing Company 1991.
- [5] Freeman J. A.: Simulating Neural Networks with Mathematica®, - Addison–Wesley Publishing Company 1994.
- [6] Haykin S., Neural networks: A comprehensive foundation, - MacMillan, NY 1994.
- [7] Skapura D. M.: Building Neural Networks, - Addison–Wesley Publishing Company 1996.
- [8] Dreyfus G.: Neural Networks – Methodology and Applications, - Springer–Verlag Berlin Heidelberg 2005.
- [9] Marti R., Fallahi A.E., (2004) Multilayer neural networks: An experimental evaluation of on–line training methods, - Computers and Operational Research, 31, 1491–1513.
- [10] Philippides A., Introduction to Neural networks, - Centre for Computational Neuroscience and Robotics, 2003.
- [11] Levenberg K., (1944) A Method for the Solution of Certain Non–Linear Problems in Least Squares, - The Quarterly of Applied Mathematics 2, 164–168.
- [12] Miljković,Z., Babić,B., Empirical Control Strategy for Learning Industrial Robot, *FME Transactions* (ISSN 1451-2092), New Series, Vol.35 No.1, pp. 1-8, University of Belgrade – Faculty of Mechanical Engineering, 2007.
- [13] Нешић,Н., Бабић,Б., Миљковић,З., Лазаревић,И., Софтвер за претпроцесирање улаза у вештачку неуронску мрежу у систему за технолошко препознавање призматичних делова, 33. ЈУПИТЕР Конференција, 20. симпозијум „CAD-CAM“, Зборник радова - CD, стр. 2.79-2.87, Златибор, 2007.
- [14] Babić,B., Nešić,N., Miljković,Z., A Review of Automated Feature Recognition with Rule-Based Pattern Recognition, *Journal Computers in Industry* (ISSN 0166-3615), Vol.59 (4), pp. 321-337, Elsevier, April 2008.
- [15] Иван Б. Лазаревић.: Развој интелигентног система надзора у специфичном технолошком постројењу, Универзитет у Београду - Машински факултет, Магистарска теза - одбрањена 28.09.2009. године.
- [16] Miljković,Z., Vojović,B., Babić,B., Application of Artificial Neural Network and Fractals in Biomedical Materials Surface Behaviour Prediction, *Часопис ТЕХНИКА-Нови материјали* (ISSN 0354-2300), Вол. 19 бр. 3, стр. (прихваћен рад, у штампи), 2010.
- [17] Visual Basic 5.0, Enterprise edition, *Microsoft Corporation*, 1997.
- [18] Бабић,Б., Миљковић,З., Бојовић,Б., Вуковић,Н., Флексибилна аутоматизација и имплементација интелигентних технолошких система у домену производње делова од лима, Пројекат технолошког развоја који финансира Министарство за науку и технолошки развој Републике Србије: ТР-14031, Београд, 2008-10.