O soluţie bioinspirată pentru generarea unui comportament autonom local, reflex, de evitare a obstacolelor

Dobrea Monica-Claudia, Dobrea Dan Marius

Abstract: Scopul principal al prezentei lucrări îl constituie găsirea unei soluţii online, simple pentru problema de evitare a obstacolelor întâlnită în cazul roboţilor mobili. Soluţia propusă de noi este una ce permite robotului dezvoltarea unui comportament autonom, local, de evitare a obstacolelor de fiecare dată când comanda motorie de nivel înalt care-l conduce (de ex., comanda *mergi înainte* sau *înapoi*) îl pune pe acesta într-o situaţie de coliziune iminentă. Soluţia propusă de noi pentru un robot cu 36 de senzori infraroşii, distribuiţi uniform în jurul acestuia, este una foarte simplă, bazată pe o reţea neuronală artificială minimală antrenată cu un algoritm derivat din algoritmul clasic de propagare înapoi a erorii (în engl. *backpropagation*, **BP**). Cu o încărcare computaţională mică, algoritmul de învăţare on-line implementat de noi s-a dovedit a fi unul foarte eficient atât în mediile aglomerate, statice cât şi în cele dinamice. Rezultatele raportate aici au fost obţinute în mediul de simulare MobotSim 1.0.03 – un simulator 2D, configurabil, destinat roboţilor mobili cu roţi, cu acţionare diferenţială.

1. Introducere

În cazul roboţilor mobili o abilitate fundamentală a acestora constă în capacitatea autonomă locală a acestora de a evita obstacole. Până în prezent, numeroase metode **[Tan, 2008]**, **[Gutnisky, 2004]** au fost propuse în vederea implementării acestui tip de comportament. În timp ce parte dintre aceste metode au fost considerate în cadrul unui simplu comportament de deplasare la întâmplare, alte metode au fost discutate şi implementate în cadrul unor comportamente mai complexe cum ar fi cele în care roboţilor li se cere să ajungă într-o locaţie specificată, să-şi planifice calea sau să urmeze fie pereţii, fie o ţintă mobilă, fie o linie marcată pe podea etc.

În cele ce urmează, problema evitării locale a obstacolelor este una tratată ca parte integrantă a unui proiect mai larg, al cărui scop final este acela de a construi un scaun cu rotile, inteligent, controlat mental de persoane cu deficienţe. Scaunul inteligent va asista utilizatorii în navigare; astfel, de fiecare dată şi imediat după ce utilizatorii vor da o comandă (de exemplu, comanda *mergi înainte*), scaunul inteligent o va executa pe aceasta însă, va avea simultan grijă să evite, în mod adecvat şi autonom, obstacolele aflate în cale.

Spre deosebire de multe alte abordări existente, abordarea noastră ia în considerare nu strategii de planificare a evitării obstacolelor ci însăşi comportamentul reflex de apărare care, la oameni, este declanşat atunci când o modificare neaşteptată şi bruscă apare în mediul imediat înconjurător – mai exact, în aşa-numitul spaţiu peripersonal (SPP) **[Graziano, 2006]**. Rolul cheie al acestui mecanism de adaptare rapidă – mecanism implementat la nivelul cel mai de jos al execuţiei motorii – constă în plasarea, cu prioritate, a robotului în afara oricărui pericol de ciocnire cu obstacole dinamice şi/sau statice. Conform unei arhitecturi reactive de tip de jos-în sus, decizia de la nivelul cel mai de jos al evitării unui obstacol este luată, mai departe, în

considerare de către nivelurile superioare de control motor ce sunt, la rândul lor, dedicate unor scopuri din ce în ce mai abstracte. În cazul nostru, următorul scop ce trebuie urmărit în cadrul proiectului global este acela ca robotul să aibă, pe lângă capacitatea de a evita (în mod autonom), şi capacitatea de a ocoli (în acelaşi timp) obstacolul întâlnit astfel încât să fie păstrată, în continuare, direcţia iniţială de mers. În lucrarea prezentată în cele ce urmează doar comportamentul autonom reflex, de evitare a obstacolului (dezvoltat la nivelul motor cel mai de jos), integrat cu comanda superioară (în particular, cea care dictează o anume direcţie globală de mers) va face obiectul studiului nostru, urmând ca, pe viitor, să fie luată în studiu şi problema ocolirii obstacolului (scop motor intermediar ce implică revenirea robotului pe direcţia iniţială de mers).

2. Câteva consideraţii tehnice şi biologice

2.1 Consideraţii biologice

În vederea dezvoltării unui comportament reflex (bio-inspirat) de evitare a obstacolelor au fost avute în vedere o serie de rezultate biologice şi psihologice raportate în literatură. Dintre acestea, elementele cheie folosite în implementarea noastră sunt cele prezentate mai jos.

La om, majoritatea celulelor sistemului nervos central (SNC) sunt formate înainte de naştere însă cea mai mare parte a conexiunilor dintre celulele nervoase se realizează după naştere, în timpul copilăriei timpurii. Modul în care se realizează aceste conexiuni este unul dictat, în mod esenţial, de: (**a**) *interacţiunea constantă a copilului cu mediul său înconjurător* şi, mai mult, (**b**) *de cronologia experienţelor acumulate de copil* (astfel, de exemplu, o experienţă timpurie, de interacţiune a copilului cu mediul său, este de aşteptat să aibă o influenţă mai mare în dezvoltarea sistemului nervos central al copilului, prin aceea că ea dă naştere unor conexiuni nervoase care vor dicta ulterior maniera în care copilul va realiza noi achiziţii – fie acestea sub forma unor informaţii, abilităţi noi etc.).

Reflexele – principalele răspunsuri motorii involuntare de apărare: (**1**) pot fi *înnăscute* sau (**2**) pot fi *comportamente învăţate* ce permit realizarea, într-un mod automat, a unor activităţi mai complexe. Acestea din urmă: (***i***) sunt fie învăţate de fiecare individ, în mod independent, fie sunt predate de către terţe persoane, (***ii***) vin din experienţă (fie într-o manieră de tip *încercare şi eroare*, fie folosind memoria experienţelor trecute şi observarea terţilor) şi (***iii***) pot fi rafinate printr-o practică continuă, în scopul unei mai bune adaptări la condiţiile de mediu aflate într-o permanentă schimbare.

La apărarea corpului şi *menţinerea unei margini de siguranţă* a acestuia îşi aduc aportul *două arii corticale importante*, având răspunsuri neuronale cu latenţă mică. Aceste arii corticale sunt: **a**) *aria ventrală intra-parietală* (VIP) şi **b**) aria *frontală polisenzorială* (PZ) **[Graziano, 2006]**. În ariile VIP şi PZ (arii unde neuronii răspund la tipuri similare de stimuli senzoriali şi a căror stimulare conduce la ieşiri de tip defensiv[1] similare [**Graziano, 2002**]): ***i***) cei mai mulţi neuroni sunt bimodali şi trimodali, răspunzând simultan la stimuli vizuali, tactili şi auditivi, ***ii***) independent de mărimea stimulilor, circa jumătate din celulele VIP

[1] Mişcările evocate sunt mişcări de evitare, retragere sau de protejare a părţii corpului pe care se află localizat câmpul tactil receptiv.

răspund cel mai bine la stimuli vizuali situaţi la mai puţin de 30 cm de corp iar multe dintre aceste celule răspund doar la stimuli aflaţi la foarte puţini centimetri de corp; în mod similar, în aria PZ circa 46% dintre neuroni prezintă un răspuns puternic şi susţinut doar pentru acei stimuli vizuali situaţi la mai puţin de 5 cm de corp în timp ce alţi 40% dintre neuroni răspund la stimuli vizuali situaţi la mai puţin de circa 20 cm de corp; ***iii***) răspunsurile celulelor multimodale pun în evidenţă un gradient al ratei de descărcare ca o funcţie de distanţa la stimulii localizaţi în spaţiul peripersonal (de exemplu, răspunsurile descresc neliniar ca o funcţie de distanţa la stimuli); ***iv***) unii neuroni VIP multisenzoriali primesc şi intrare vestibulară şi, din acest motiv, se consideră că aceştia ar fi implicaţi în detectarea direcţiei mişcării autogenerate a subiectului; ***v***) neuronii PZ proiectează direct la nivel spinal, fiind astfel implicaţi în ieşirea motorie defensivă.

Comenzile motorii ce coboară de la nivel cortical sunt direct responsabile de iniţierea şi generarea mişcărilor voluntare. Reflexele complexe (aşa cum este, de exemplu, reflexul de evitare a obstacolelor, care cel mai frecvent sub-serveşte alte mişcări voluntare cu scop precis) apar prin proiecţia intrărilor senzoriale la nivelul circuitelor neuronale spinale. Proiecţia intrărilor senzoriale se realizează prin intermediul inter-neuronilor spinali iar circuitele spinale la nivelul cărora se face proiecţia sunt frecvent reprezentate de generatorii centrali de pattern[2] (GCP). În această reţea de conexiuni, rolul inter-neuronilor spinali este acela de a integra intrările descendente, venite de la creier, cu intrările senzoriale aferente şi, în consecinţă, acela de a adapta reflexele şi activitatea neuronilor motori spinali la diferitele cerinţe funcţionale apărute.

Teoria celor doi factori ai lui Mowrer privind evitarea postulează faptul că învăţarea procesului de evitare implică *două etape*:

(1) *O primă etapă*, în care subiectul care învaţă experimentează *condiţionarea clasică/bazată pe sentimentul de frică generat*. Mai exact, un stimul de avertizare, aşa cum este o distanţă mică (sau un anumit prag) până la cel mai apropiat obstacol, este asociat cu o situaţie neplăcută, aşa cum este coliziunea; în acest mod, un stimul iniţial neutru devine un stimul condiţionat, SC, capabil să producă un puternic răspuns condiţionat, de frică.

(2) *O a doua etapă*, în care subiectul experimentează *condiţionarea operantă*. În această etapă subiectul adoptă o acţiune de răspuns la *stimulul condiţionat aversiv* şi, astfel, elimină – prin *întărire negativă* (în engl., negative reinforcement) – *evenimentul aversiv*. Această ultimă etapă corespunde însuşi procesului de învăţare a evitării iar în această etapă comportamentul de evitare nu este întărit prin evitarea situaţiei neplăcute (ciocnirea de obstacol) ci, prin terminarea stimulului condiţionat aversiv – adică, a stimulului care a evocat sentimentul de frică . Cu alte cuvinte, *stimulul aversiv este cel care întăreşte acele răspunsuri care îl elimină pe el însuşi*.

Teoria ecologică a lui Gibson privind percepţia vizuală **[Shumway-Cook, 2007]** reclamă faptul că pentru a genera acţiuni adecvate oamenii au nevoie de informaţia perceptuală (nu senzorială!) legată de factorii de mediu care sunt importanţi pentru task-ul motor de executat. În cazul nostru ne-am folosit de o

[2] Aceştia sunt implicaţi în generarea secvenţelor motorii stereotype.

percepţie a adâncimii medii până la obstacolele din jur, alături de percepţiile legate de direcţia mişcării şi viteza de deplasare.

2.2 Consideraţii tehnice şi practice

În cele prezentate în continuare, implementarea propusă pentru procesul de învăţare a evitării obstacolelor va trebui privită doar ca un model foarte reducţionist al analogului său uman, fără a se pierde însă şi din relevanţa sa.

Dintre modelele bio-inspirate propuse în literatură ca soluţii pentru învăţarea evitării obstacolelor, *modelele de învăţare cu întărire* (în engl. *reinforcement learning*, RL), precum şi *modelele de tip comportament operant* (în engl. *operant behavior*, OB) par a fi cele mai promiţătoare.

Aşa cum am precizat anterior, în teoria celor doi factori ai lui Mowrer vorbim de *învăţare* prin *evadare* (învăţăm să sfârşim ceva aversiv) şi *învăţare* prin *evitare* (învăţăm să prevenim ceva aversiv). Ca şi în *învăţarea* prin *evadare*, cele două tipuri de modele, RL şi OB:

(1) nu necesită o cunoaştere completă a mediului şi/sau cunoaşterea acţiunii ce trebuie luată în fiecare context particular de mediu (oricum, nu putem vorbi de o soluţie unică pentru acţiunea ce poate fi abordată la un moment dat);

(2) aceste două modele necesită, mai presus de toate, cât mai multe interacţiuni posibile ale robotului cu mediul său înconjurător; din acest punct de vedere, aceste modele se încadrează în clasa *algoritmilor de învăţare de tip on-line*;

(3) interacţiunea cu mediul (static şi/sau dinamic) are la bază tehnica de învăţare *bazată pe încercare şi eroare*;

(4) ambele modele furnizează robotului mobil autonom o *evaluare permanentă* şi corespunzătoare a performanţelor sale, evaluare ce se face în termeni de *pedeapsă* şi *recompensă*.

Cu toate acestea, în timp ce rezultatele obţinute cu fiecare dintre aceste două modele par a fi foarte promiţătoare, fiecărui model, în parte, fie îi lipsesc unele detalii importante, fie acesta se confruntă cu unele probleme practice, de implementare; iar toate acestea fac ca, în final, soluţia obţinută să fie una departe de a exprima un comportament asemănător celui întâlnit la subiecţii umani. În particular, în cazul metodei RL şi a celei mai larg utilizate implementări a ei (vorbim aici de *algoritmul Q-learning*[3] **[Watkins, 1992]**) menţionăm cel puţin următoarele dezavantaje:

(i) În primul rând, această metodă implică manipularea unui tabel foarte mare. Acest tabel este unul utilizat în actualizarea valorilor-Q. Astfel, spre exemplu, pentru un robot cu numai 8 senzori, cu 5 acţiuni posibile de executat şi cu un domeniu al valorilor de intrare de [0, 1022] pentru fiecare senzor în parte, avem nevoie de un tabel-Q cu nu mai puţin de 1.1995 x 10^{24} x 5 intrări. Recent, o alternativă la tabelele de tip look-up (utilizate

[3] În care se estimează pentru fiecare pereche posibilă [*stare*, *acţiune*] un semnal mediu de tip recompensă numerică.

iniţial pentru a stoca valorile-Q) o reprezintă reţelele neuronale artificiale (RNA). Acestea din urmă sunt utilizate în principal datorită celor două capacităţi ale lor, respectiv: **a**) capacitatea de a oferi o reprezentare mai compactă a valorilor-Q şi **b**) capacitatea de a interpola valorile-Q pentru perechile stare-acţiune care nu au fost vizitate niciodată. Totuşi, în cazul utilizării diferitelor paradigme RNA noi probleme pot să apară **[Tan, 2008]**, aşa cum este, spre exemplu, cazul instabilităţii raportate pentru arhitecturile de tip perceptron multistrat (MLP), antrenate cu algoritmul BP. Mai precis, este foarte dificil de a garanta faptul că învăţarea noilor pattern-uri nu erodează cunoştinţele acumulate anterior.

(ii) Un al doilea dezavantaj important este dat de numărul foarte limitat al acţiunilor posibil de executat de către robot – număr ce este determinat, în principal, ca urmare a constrângerilor de calcul.

(iii) Un alt dezavantaj major îl constituie numărul mare de parametri necunoscuţi atât ai algoritmului Q-learning (de ex., rata de actualizare, parametrul de temperatură iniţial, funcţia de recompensă etc.), cât şi ai implementării RNA; pentru mai multe dezavantaje vezi şi **[Gutnisky, 2004]**.

Spre deosebire de *metodele RL*, unde într-o primă fază de explorare (*fază de învăţare*), în mod independent de starea curentă, robotul explorează mediul prin selectarea unor *acţiuni non-greedy*[4] (folosind, în acest sens, distribuţia de probabilitate Boltzmann), în *condiţionarea operantă* **[Gutnisky, 2004]** stimulii primiţi de robot sunt utilizaţi nemijlocit pentru a învăţa ce acţiuni să realizeze mai mult (este vorba de acţiunile ce au primit recompensă) şi ce acţiuni să realizeze mai puţin (respectiv, acţiunile care au fost penalizate). În consecinţă, în timp ce ambele tipuri de metode sunt considerate metode de tip on-line, *operarea în timp real* rămâne o caracteristică doar pentru metodele de tip OB. În plus, în **[Gutnisky, 2004]** găsim raportat faptul că nici una dintre metodele menţionate aici nu conduce la performanţe maxime; o explicaţie posibilă pentru aceasta ar putea fi faptul că în faza de antrenare termenul de recompensă sau pedeapsă este unul utilizat în detrimentul semnalului de intrare senzorial care este unul mult mai adecvat dat fiind marele său potenţial informativ.

3. Procesul de învăţare (bio-inspirată) a evitării reflexe a obstacolelor

În cele ce urmează o nouă metodă bio-inspirată, de învăţare a evitării reflexe a obstacolelor va fi prezentată, alături de implementarea ei practică pe o platformă robotică. Platforma robotică cu roţi şi cu acţionare diferenţială este una dotată cu 36 de senzori infraroşii, IR, distribuiţi în mod echidistant în jurul ei (vezi Figura 1.a). Un controller bazat pe o reţea minimală de tip MLP şi antrenată cu un nou algoritm bio-inspirat (**BBP**) – derivat din algoritmul BP – va constitui, în cele ce urmează, elementul cheie folosit în vederea întrunirii criteriului de învăţare on-line şi în timp real, întâlnit dealtfel la subiecţii umani.

[4] *Strategia* (*metoda*) *greedy* presupune efectuarea unei alegeri. Dintre toţi paşii următori posibili de ales, se alege acel pas care asigură un maximum de "cîştig", de unde şi numele metodei: *greedy* = lacom.

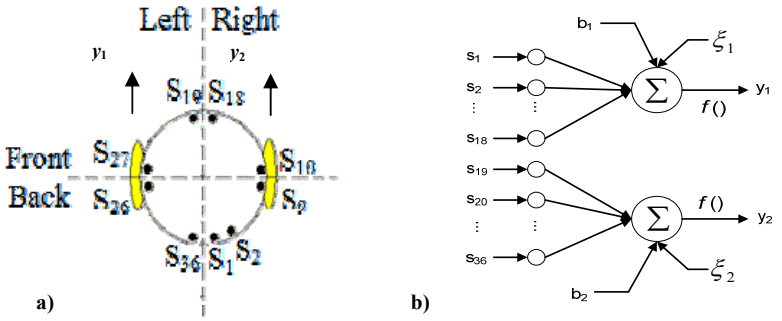

Figura 1: a) Dispunerea senzorilor IR, b) arhitectura reţelei de tip MLP.

Arhitectura minimală a reţelei de tip MLP, **Figura 1.b**, a constat în 36 de intrări (corespunzător celor 36 de valori, s_i, citite de la senzorii IR) şi în 2 ieşiri (y_1, y_2, ce furnizează comenzile pentru cele două motoare ce acţionează roţile difrenţiale ale robotului). Raza scurtă de acţiune a senzorilor IR, de doar 30 cm, i-a făcut pe aceştia să fie foarte adecvaţi în cadrul mecanismelor de adaptare rapidă la mediu.

Într-o primă etapă (de preprocesare), valorile senzorilor au fost liniarizate iar, mai apoi, ele au fost normalizate în intervalul [0, 0.9], cu 0 desemnând lipsa oricărui obstacol în vecinătatea imediată a robotului şi 0.9 denotând coliziune. În cadrul topologiei reţelei MLP, funcţiile de transfer adoptate pentru cele două elemente de procesare de ieşire au fost de tip *tanh*; acest din urmă tip de funcţii prezintă, suplimentar, avantajul că limitează valorile de ieşire în intervalul [-1,+1].

Comenzile globale de deplasare a robotului – codate, după cum urmează: *înainte* (1,1), înapoi (-1,-1), la stânga (0,0.5), la dreapta (0.5, 0) şi stop (0,0) – au fost luate în considerare, la nivelul reţelei MLP, prin intermediul bias-urilor (b_1, b_2) ale neuronilor de ieşire. Direcţiile de deplasare înainte şi înapoi ale roţilor robotului au fost şi ele codate prin valori pozitive şi, respectiv, negative în timp ce virajele la dreapta şi, respectiv, la stânga, au fost obţinute prin comenzi diferenţiale adecvate transmise la nivelul celor două motoare. Informaţia de viteză, y_j, a fost una furnizată ca o valoare proporţională în intervalul 0 (*stop*) şi 1/-1 (*viteză maximă înainte/înapoi*). La calculul activării neuronilor de ieşire ai reţelei s-a adăugat, de asemenea, un termen reprezentând un mic zgomot, ξ, cu rol în a preveni robotul de a se bloca în anumite condiţii particulare de mediu.

În lumina celor prezentate mai sus, o analogie grosieră a sistemului nostru robotic cu sistemul neuro-motor uman poate arăta astfel: stratul de intrare al reţelei neuronale este omologul ariei frontale polisenzoriale PZ, neuronii de ieşire sunt inter-neuronii care integrează informaţia senzorială primară (ce a trecut prin aria PZ) cu comanda corticală iar sistemul de acţionare a roţilor (controller-ul implementat cu o reţea MLP) ar corespunde generatorilor centrali de pattern-uri.

În forma sa standard (1), algoritmul BP impune existenţa unui profesor care ştie sau care poate calcula ieşirea dorită pentru fiecare intrare (condiţie de mediu) dată. Totuşi, acesta nu este şi cazul nostru întrucât noi nu putem defini valorile dorite pentru comenzile y_1, y_2, acestea din urmă depinzând de factori multipli care, adesea, sunt şi imprevizibili.

$$BP:\quad \Delta w_{ij} = -\eta \frac{\delta E}{\delta w_{ij}} = -\eta (d_j - y_j) s_i (1 - y_j^2),\quad \Delta b_j = -\eta (d_j - y_j)(1 - y_j^2)$$
$$E = \frac{1}{2}\sum_{j=1}^{2} (d_j - y_j)^2 \qquad (1)$$

În relația (1) w_{ij} sunt ponderile rețelei neuronale artificiale, E reprezintă eroarea iar η este rata de învățare (valoarea ei a fost aleasă fixă, și anume 0.7).

Pentru a putea face, totuși, uz de algoritmul BP (algoritm ce descrie cel mai bine modul în care copiii învață exersând, într-o manieră de tip *încercare și eroare*) am ținut cont și exploatat, în mod corespunzător, trei dintre teoriile psihologice, și anume: ***i***) *teoria ecologică a percepției vizuale*, ***ii***) *teoria motivației interne* **[Oudeyer, 2007]** și ***iii***) *teoria evitării a lui Mowrer* (teoria celor doi factori). În cele din urmă am ajuns la o nouă paradigmă, și anume, am substituit *paradigma centrată pe professor* (algoritmul BP) cu o *nouă paradigmă, centrată pe elev.* Relația de calcul aferentă noului algoritm, BBP (algoritm pe care noi l-am numit, oarecum impropriu, algoritmul BP-bioinspirat) este cea prezentată în relația (2).

$$Forward:\quad y_j = \tanh\left(\sum_i w_{ij} s_i + b_j + \xi_j\right),\ i = \overline{1,18}\ pt.\ j = 1\ si\ i = \overline{19,36}\ pt.\ j = 2$$
$$BBP:\quad \Delta w_{ij} = -\eta T_j s_i (1 - y_j^2),\qquad \Delta b_j = 0$$
$$T_j = E_j\ Crit_j$$
$$E_j = A_j - A_j^d = A_j,\quad A_j = \frac{\sum_i s_i}{n_j},\ i = \overline{1,18}\, pt.\, j = 1, i = \overline{19,36}\, pt.\, j = 2 \qquad (2)$$
$$Crit_j = sign(\Delta E_j) sign(y_j) = sign(\Delta A_j) sign(y_j)$$

În noua relație de actualizare a valorilor ponderilor (2), variația ponderilor w_{ij} diferă față de variația ponderilor din relația (1) doar prin termenul compozit T_j pentru care termenul echivalent în relația (1) este $(d_j - y_j)$. În relația (1) acest termen din urmă este o măsură a cât de mult din valoarea erorii se datorează ieșirii j a RNA. Spre deosebire de algoritmul BP standard (1), în care acest termen al erorii are o formă analitică clară ce depinde de valoarea dorită, d_j, și de ieșirea actuală, y_j, a RNA, în algoritmul BBP modul de calcul al termenului său echivalent, T_j, este unul ce ține cont:

- pe de o parte, de *maniera* particulară în care „creierul elevului" *percepe stimulii din mediul imediat înconjurător* (în particular, vorbim aici de termenul $\boldsymbol{E_j}$) și,
- pe de altă parte, de *modul* în care elevul *manipulează*, la nivel cortical, *aceste percepții* (reflectat, la nivel de relație, prin termenul $\boldsymbol{Crit_j}$).

Acești ultimi termeni nou introduși vor fi explicați mai târziu în cadrul acestei lucrări.

În mod natural și firesc creierul uman folosește diferența dintre modul în care lumea este percepută și modul în care ea este de așteptat să fie percepută (corespunzător scopurilor noastre) ca o informație de eroare funcție de care încearcă, apoi, să corecteze actul motor (mișcarea).

Pentru a genera acțiuni adecvate oamenii utilizează informația perceptuală (nu pe cea senzorială) legată de factorii de mediu care prezintă importanță pentru task-ul motor ce trebuie făcut. În cazul nostru, *percepția stimulului aversiv*, A_j – obținută ca *percepția adâncimii medii*[5] până la obstacolele aflate în partea contralaterală dreaptă/stângă a robotului – a fost cea utilizată în faza de învățare (2). Pentru fiecare neuron de ieşire al RNA s-a calculat o valoare diferită A_j corespunzătoare, scopul fiind acela de a facilita învățarea reflexului de evitare a obstacolului pe ambele părți (în acest mod se asigură evitarea obstacolului pe partea cea mai potrivită). În relația (2), parametrul $(n_j)_{j=1,2}$ reprezintă numărul senzorilor de valoare non-nulă dispuşi pe partea contralaterală a robotului. Valoarea acestui parametru variază de la un moment de timp la altul, funcție de condițiile de mediu (respectiv, prezența sau absența obstacolelor în spațiul peripersonal al robotului).

În cadrul algoritmului de învățare BBP, *stimulul condiționat aversiv* dat de percepția A_j, creşte cu cât adâncimea medie până la obstacolele aflate în spațiul peripersonal (SPP) drept/stâng scade şi, respectiv, dispare (devine zero) atunci când valorile citite de la nivelul tuturor senzorilor robotului devin nule. Acest din urmă caz corespunde percepției dorite a mediului, $A_j^d = 0$; în această situație ideală robotul este situat la cel puțin 30 cm distanță de orice obstacol înconjurător. În mod corespunzător, *eroarea apreciată de elev* (robot), E_j, devine, în final (conform relației (2)), egală cu percepția A_j care nu reprezintă altceva decât însăşi stimulul aversiv care trebuie eliminat. Această eroarea calculată de către robot (şi care ia valori în intervalul [0, 0.9]) este una conformă cu paradigma fundamentală care defineşte eroarea BP[6].

În continuare, corespunzător aceleiaşi etape a condiționării operante, elevul (robotul) va întări doar acele acțiuni care elimină stimulii aversivi. În acest sens, robotul evaluează la fiecare pas consecințele acțiunilor sale ca răspuns la stimulii din mediu şi, apoi, *generează intern o recompensă scalară*, $sign(A_j[n] - A_j[n\text{-}1])$; aceasta din urmă trebuie înțeleasă ca o măsură a *progresului înregistrat în procesul de învățare*. Această măsură calitativă, împreună cu percepția direcției de mişcare a roților, formează aşa-numitul termen pe care noi l-am denumit *critic*, ***Crit***. Termenul ***Crit*** este cel care controlează maniera în care rețeaua neuronală artificială îşi actualizează parametrii. Mai exact, actualizarea ponderilor rețelei MLP în sensul menținerii sensului de mers de la pasul anterior[7] este încurajată ori de câte ori robotul se mişcă astfel încât să elimine/diminueze stimulii aversivi, $A_j[n] \leq A_j[n\text{-}1]$, şi această actualizare a ponderilor se face în sensul schimbării sensului de mers de la pasul anterior în caz contrar.

La oameni am văzut că, într-o primă fază (numită şi de *condiționare clasică*), comportamentul de evitare este unul bazat, în principal, pe *sentimentul de frică generat de consecințele evenimentului aversiv* (în cazul nostru, coliziunea). În momentul, însă, în care comportamentul de evitare începe să fie unul realizat, în mod repetat, cu succes, sentimentul de frică începe să dispară (subiectul uman începe să aibă controlul asupra situației) iar procesul de învățare a comportamentului de evitare încetează.

---

[5] A se vedea semnificația dată valorilor citite de la senzori.

[6] Atunci când sistemul adaptiv reuşeşte să rezolve cu succes problema, eroarea calculată devine zero; în caz contrar, eroarea măsoară distanța dintre rezultatele dorite şi ieşirile curente ale sistemului adaptiv.

[7] Vorbim aici de sensul de rotire (înainte sau înapoi) a fiecărei roți în parte.

În cazul nostru, pentru a putea surprinde în cadrul soluției propuse de noi și acest comportament întâlnit la oameni, o modelare a sentimentului de frică menționat mai sus apare, practic, ca o condiție necesară. În lipsa, însă, a unei astfel de modelări (reamintim că algoritmul BBP implementează condiționarea operantă, nu și etapa premergătoare, cea a condiționării clasice), soluția adoptată – și rezultată din practică – a constat în impunerea ca după primii 300 de pași de antrenare (în care regula de actualizare BBP s-a aplicat la fiecare pas), actualizarea ponderilor rețelei MLP să aibă loc în continuare doar atunci când cel puțin o intrare a RNA avea o valoare mai mare de 0.8. Această din urmă situație s-ar putea traduce prin aceea că robotul ajunge să se confrunte cu o situație nemaiîntâlnită până atunci (o situație nouă[8]), pe care nu o poate rezolva și pentru care este necesară o nouă învățare.

În mediul de simulare, implementarea soluției s-a făcut ținând cont și de caracteristicile și limitările tehnice ale robotului fizic pe care se urmărește implementarea, într-o etapă următoare, a soluției simulate. Din aceste considerente, viteza maximă de deplasare a robotului a fost setată la 0.3 m/s iar timpul scurs între fiecare acțiune (mișcare) a robotului și următoarea citire a valorilor senzorilor a fost aleasă de 400 ms. Valorile inițiale ale ponderilor rețelei nu au fost generate în mod aleator ci, dimpotrivă, ele au fost setate la zero pentru a simula procesul de conectare neurală progresivă ce are loc în sistemul nervos al copiilor începând de la naștere și ținând pe tot parcursul procesului de dezvoltare a controlului neural.

4. Rezultate și discuții

Testarea algoritmului de învățare a evitării obstacolelor propus mai sus s-a făcut în mediul de simulare MobotSim 1.0.03, cu medii atât statice cât și dinamice (de exemplu, cu doi și, respectiv, cinci roboți mobili diferiți). Un exemplu de comportament de evitare a obstacolelor învățat cu noul algoritm este și cel prezentat în **Figura 2.a** (comportament obținut după primii 2527 de pași de antrenare) sau în **Figura 2.b** (comportament obținut după primii 70517 de pași de antrenare).

În urma analizei modului în care robotul a învățat comportamentul de evitare putem trage următoarele concluzii. La fel ca și la oameni:

(1) Metoda propusă a permis *implementarea online* și *în timp real* a procesului de învățare. Aceasta trebuie privită și ca o consecință a faptului că algoritmul este unul foarte simplu, ce nu implică costuri computaționale foarte mari.

(2) *Învățarea are loc relativ repede* (după doar una sau două coliziuni) și ea este foarte durabilă, fără să presupună alte coliziuni ulterioare.

(3) Pentru procesul de învățare *foarte importante sunt primele interacțiuni ale robotului cu mediul său înconjurător*, aceste interacțiuni fiind cele care conturează, în linii mari, comportamentul de evitare de mai târziu.

(4) O consecință majoră a soluției propuse pentru algoritmul de evitare a obstacolelor constă în *construirea unei zone de siguranță* în jurul robotului. Acest fapt vine să confirme ipoteza că "spațiul personal" al omului este rezultatul unui mecanism defensiv al cărui rol este acela de monitorizare a obiectelor aflate în jurul corpului, obiecte care pot aduce

[8] Nici o situație dintre cele întâlnite de robot până atunci nu seamănă (în anumite limite) cu cea prezentă.

atingere integrității acestuia.

(5) Robotul reacționează prompt indiferent de direcția din care vine stimulul mobil (în cazul nostru, stimulul îl reprezintă un alt robot aflat în mișcare).

(6) Maniera în care este generată mișcarea[9] împreună cu modul în care sunt calculate cele două ieșiri ale RNA[10] (reprezentând comenzile pentru motoare) oferă suportul real pentru obținerea unei varietăți foarte mari de mișcări. Așa se explică marea flexibilitate a mișcării robotului obținută utilizând algoritmul nou propus.

(7) Preluarea controlului în mod automat, fie de către comanda „corticală", fie de către comportamentul autonom local, s-a făcut în mod corespunzător, ori de câte ori contextul de mediu a cerut-o.

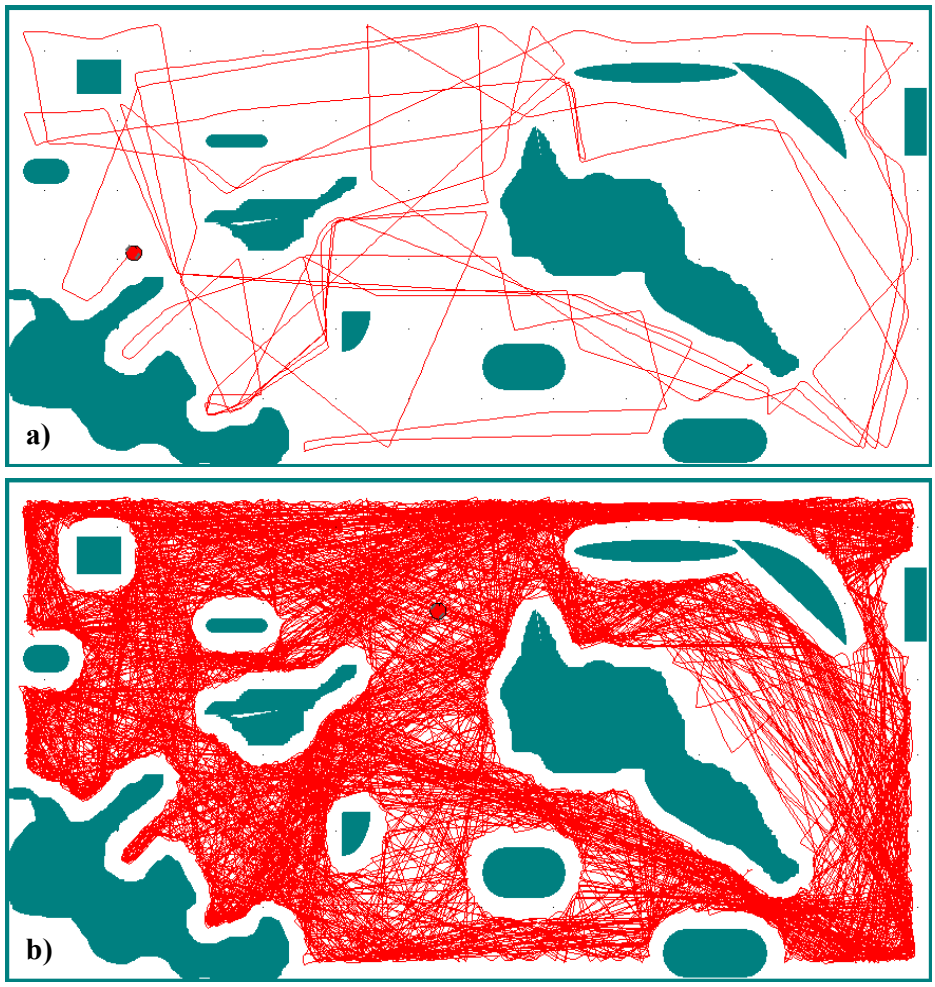

Figura 2: Comportamentul de evitare a obstacolelor învățat de robot: a) după primii 2527 pași de antrenare și b) după primii 70517 pași de antrenare ai algoritmului BBP.

[9] Și anume, prin intermediul celor două roți acționate diferențial.
[10] Avem în vedere aici valorile obținute – valori care baleiază, practic, intervalul [-1,+1].

5. Concluzii

În această lucrare s-a prezentat, pe scurt, un nou concept pentru controlul robotic de tip *comportament reactiv bio-inspirat*. Plauzabilitatea biologică a modelului, alături de simplicitatea ridicată a tehnicii şi a paradigmei de învăţare, au făcut sistemul robotic astfel obţinut să fie unul foarte eficient: ***i***) acesta învaţă repede şi de o manieră consistentă; ***ii***) comportamentul autonom este obţinut online, în timp real şi în medii dinamice şi nestructurate; ***iii***) comanda „corticală" şi comportamentul local autonom reuşesc cu success să preia, în mod automat şi exclusiv (unul îl exclude pe celălalt) controlul sistemului, ori de câte ori contextul de mediu o cere.

Referinţe

1 [Tan, 2008] Tan A. H., Lu N. şi D. Xiao: *Integrating temporal difference methods and self-organizing neural networks for reinforcement learning with delayed evaluative feedback*, în IEEE Transactions on Neural Networks, 19: 230-244, 2008

2 [Gutnisky, 2004] Gutnisky D. A. şi Zanutto B. S.: *Learning obstacle avoidance with an operant behavior model*, în Artificial Life, 10: 65-81, 2004

3 [Graziano, 2006] Graziano M. S. şi Cooke D. F.: *Parieto-frontal interactions, personal space, and defensive behavior*, în Neuropsychologia, 44: 845–859, 2006

4 [Graziano, 2002] Graziano M. S., Taylor C. S. şi Moore T.: *Complex movements evoked by microstimulation of precentral cortex*, în Neuron, 34: 841–851, 2002

5 [Shumway-Cook, 2007] Shumway-Cook A. şi Woollacott M. H.: *Motor control: translating research into practice*, 3rd ed., Lippincott Williams & Wilkins, U.S.A., pp. 16, 2007

6 [Watkins, 1992] Watkins C. şi Dayan P.: *Q-Learning*, în Machine Learning, 8: 279-292, 1992

7 [Oudeyer, 2007] Oudeyer P.-Y. şi Kaplan F.: *What is intrinsic motivation? A typology of computational approaches*, în Frontiers in Neurorobotics, 1(6): 1-14, 2007