**Capitolul 1**

**INTRODUCERE. AGENȚI INTELIGENȚI**

Cuprins – Cap. 1

[1. INTRODUCERE. AGENȚI INTELIGENȚI. 11](#_Toc528594161)

[1.1 Generalități 11](#_Toc528594162)

[1.2 State of the Art 17](#_Toc528594163)

[1.3 Operatorul Harris 22](#_Toc528594164)

[1.4 Detectorul SIFT 25](#_Toc528594165)

[1.5 Inteligența artificială distribuită 29](#_Toc528594166)

[1.6 Agenți inteligenți 34](#_Toc528594167)

[BIBLIOGRAFIE – cap. 1 41](#_Toc528594168)

[ANEXE – cap. 1 54](#_Toc528594169)

[FIGURI – cap. 1 54](#_Toc528594170)

[TABELE – cap. 1 54](#_Toc528594171)

[NOTAȚII ȘI SIMBOLURI – cap. 1 54](#_Toc528594172)

[ABREVIERI – cap. 1 56](#_Toc528594173)

Contribuțiile autorilor

|  |  |
| --- | --- |
| *Conferențiar universitar*  Dr.ing. Lucian Ștefăniță GRIGORE | *Conferențiar universitar*  Dr.comp.sc. Iustin PRIESCU |
| 1.1 – pg. xxx÷xxx | 1.1 – pg. xxx÷xxx |

|  |  |
| --- | --- |
| *Cercetător grd.I*  Dr.comp.science Angela IONIȚĂ | *Lector universitar*  Dr.comp.sc. Dan-Laurențiu GRECU |
| 1.1 – pg. xxx÷xxx | 1.1 – pg. xxx÷xxx |

# INTRODUCERE. AGENȚI INTELIGENȚI.

## Generalități

Inteligența artificială (AI) este un ansamblu de reprezentări, tehnici si arhitecturi folosite pentru a construi sisteme aplicate din punct de vedere computațional. AI studiază regulile de înlănțuire, căutare euristică, logică, propagare a constrângerilor, căutare constrânsă, și alte paradigme de rezolvare a problemelor. În plus, utilizează arbori de decizie, **rețele neurale RN** (ANN - A*rtificial Neural Network*) – este forțată definirea RN ca rețea *neuronală* (<https://youtu.be/baI9q2--q7s> ), SVMs[[1]](#footnote-1) (Support Vector Machines) și alte paradigme de învățare. Rețeaua neuronală reprezintă procesul de învățare/memorare și este specific unei ființe umane, care dispune de un organ central (creier) constituit din neuroni organizați, astfel încât, să poată coordona funcțiile vitale ale organismului uman. Referitor la ființe există două mari tipuri de rețele neuronale:

* Read Only – sunt transmise de la o generație la alta prin ADN – ele dezvoltându-se pe parcursul a milioane de ani și a zeci de mii de generații și care permit indivizilor dintr-un grup să aibă anumite caracteristici unice – de exemplu – capacitatea de a vorbi;
* Dinamic – se dezvoltă pe parcursul vieții unui individ prin învățare, exersare și repetiție și nu se transmit genetic la următoarea generație, însă modifică codul ADN al rețelelor de tip Read Only (*în sensul îmbunătățirii*). Aceste rețele permit de exemplu rezolvarea ecuațiilor de gradul 2.

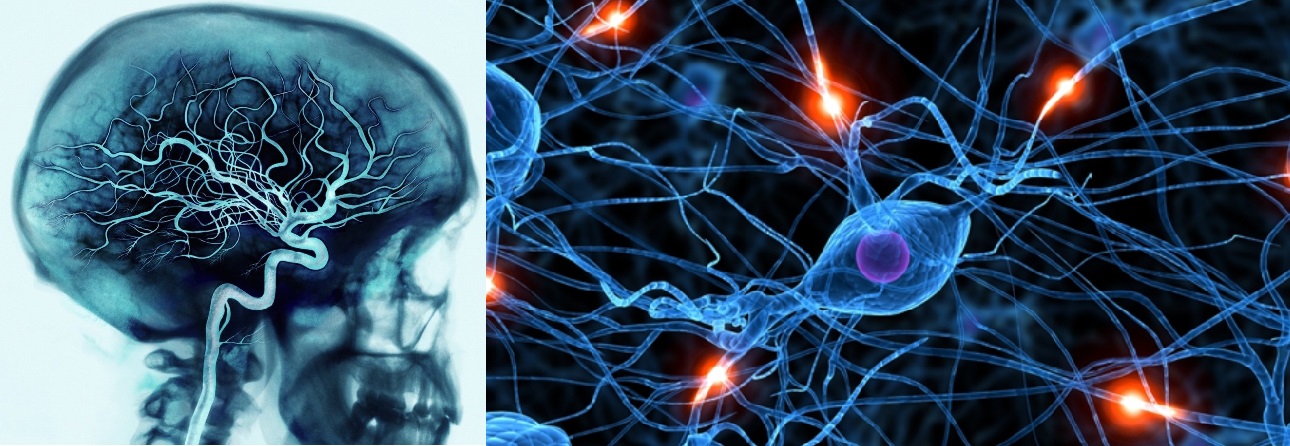


Fig. 1‑1 Rețea neuronală specifică unei ființe umane.

<http://seniorjournal.com/NEWS/Aging/2011/20110919-SignsOfAging.htm>

<https://www.mta-portal.de/aktuell/21-12-2015-aus-genetischen-erkrankungen-lernen/> © Eraxion / iStock / Thinkstock

Definirea noțiunii de IA are în vedere modul în care funcționează procesele de gândire umană, raționamentele și comportamentele. Deși există mult orientări din punct de vedere semantic, IA este o formă artificială de percepție, analiză și acțiune față de stimulii introduși în ecuație.

Literatura de specialitate folosește în acest moment câteva clase de definiții:

* sisteme bazate pe gândirea umană;
* sisteme bazate pe un mod de gândire rațional;
* sisteme similare ca mod de acțiune cu cel uman;
* sisteme al căror mod de acțiune are la bază algoritmi raționali.

Aceste clase de „*interpretare*” a realității funcționează în baza unor algoritmi polinomiali, care reprezintă aproximări ale unor probleme pentru care nu există soluţii polinomiale, a unor sisteme fizic bazate pe testul Turing, a modul în care interacționează o mașină și a tehnicilor computaționale.

Testul Turing[[2]](#footnote-2) este în fapt un proces simplu de verificare a modului în care un interlocutor uman poartă un dialog cu un om și cu un calculator. Dacă interlocutorul este separat de om și de calculatorul autonom este posibil ca operatorul să nu știe de la cine primește răspunsul. Carcateristicile calculatorului trebuie să poată procesa limbajul utilizat, să poată efectua o reprezentare a informației, să raționeze independent și mai ales să poată învăța, astfel încât să se poată adapta.

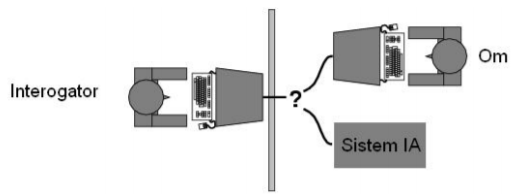


Fig. 1‑2 Reprezentarea experimentului Turing

<http://id.inf.ucv.ro/~cstoean/courses/ia/c1.pdf>

Domeniile inteligenței artificiale şi roboticii mobile au fost întotdeauna strâns legate. Chiar înainte de conferinţa de la Dartmouth College[[3]](#footnote-3), din anul 1956, când termenul de inteligență artificială a fost utilizat în mod oficial, se cunoştea faptul că roboţii mobili pot fi programaţi să îndeplinească taskuri complexe şi să înveţe. Marvin Minsky şi John McCarthy, pionieri în studiul inteligenței artificiale, au fost interesaţi de robotica mobilă imediat după aceasta conferinţă şi, la sfârşitul anilor '50, împreună cu Richard Greenblood şi William Gosper, încearcă să construiască un robot care să poată juca ping-pong. Dificultăţile tehnice datorate unui hardware neadecvat fac însă ca robotul să poată prinde doar mingea într-un coş de baschet.

În foarte multe cazuri, roboţii mobili, sunt utilizați sub forma unor vehicule capabile să opereze autonom și/sau semiautonom, după cum urmează:

* tereștri - **UGV** - Unmanned Ground Vehicle;
* aerieni - **UAV** - Unmanned Aerial Vehicle;
* acvatici/maritime - **UUV** - Unmanned Undersea Vehicle.

În prezent, sistemele robotizate mobile se află la intersecţia mai multor domenii ale ştiinţei. Remarcăm dintre acestea:

* **Knowledge Representation** - *Reprezentarea cunoştinţelor*: este domeniul inteligenței artificiale care ajută la obţinerea unui model intern al lumii asociat taskului pe care îl execută robotul. Se folosesc structuri de date şi algoritmi specifici pentru reprezentarea caracteristicilor obiectelor din mediu;
* **Natural language understanding (NLU)** - *înţelegerea limbajului natural*: limbajul natural se schimbă cu o frecvenţă destul de mare, astfel încât, simpla recunoaştere a cuvintelor nu mai este suficientă pentru înţelegerea sensului real al unui text prelucrat;
* **Learning** - *Învăţarea*: în mod ideal, un robot ar putea învăţa task-urile pe care le are de îndeplinit prin simpla repetare a acţiunilor similare executate de un operator uman;
* **Planning and problem solving** - *Planificare şi soluţionare de probleme*: inteligenţa artificială este în mod inerent asociată abilităţii de a planifica acţiunile necesare pentru a îndeplini un anumit obiectiv, precum şi abilităţii de a rezolva probleme apărute în cazul nefuncţionării acestor planuri de acţiune;
* **Inference** - *Inferență*: prin inferenţă, este posibilă completarea informaţiei care lipseşte pentru soluţionarea unei anumite probleme;
* **Search** - *Căutare*: în termenii inteligenței artificiale, prin căutare se înţelege examinarea eficientă a unei reprezentări a cunoştinţelor, specifică unei anumite probleme (căutare într-un „*spaţiu*”), în scopul determinării unei soluţii;
* **Vision** - *Vedere artificială*: prin realizarea sistemelor de vedere artificială, acţiunile robotului mobil pot deveni mai precise şi mai complexe.

Roboţii mobili semiautonomi/autonomi reprezintă în viziunea actuală cea mai apropiată aproximare a agenţilor inteligenţi. Oamenii au fost interesaţi întotdeauna de construirea unor maşini care să imite fiinţele biologice. Începând cu realizările mecanice ale vechilor orologii şi până la agenţii software şi agenţii fizici actuali, căutarea răspunsului la eterna întrebare „*Ce este viața?*” a constituit întotdeauna un motiv de impulsionare a cercetărilor ştiinţifice. În prezent, pentru a interacţiona cu mediul înconjurător, astfel de sisteme inteligente anticipează rezultatul propriilor acţiuni şi fac predicţii asupra comportamentului altor obiecte. Această strânsă conexiune între percepţie şi acţiune motivează investigarea comportării inteligente prin intermediul roboţilor mobili. Ne situăm aici la intersecţia ştiinţelor cognitive, psihologiei şi inteligenţei artificiale. Din acest punct de vedere, roboţii mobili care operează în mod autonom/semiautonom oferă o excelentă soluţie de testare a ipotezelor referitoare la comportarea inteligentă, percepţie şi cunoaştere.

O mare parte a roboţilor utilizaţi astăzi în industrie sunt de tip manipulator (denumiţi şi roboţi de asamblare), care operează într-un spaţiu mărginit şi nu se pot mişca. Spre deosebire de aceştia, roboţii mobili îşi pot schimba locaţia prin locomoţie.

În sensul larg al definiţiei, un **robot mobil** este un **sistem de integrat timp-real** (*embedded real-time system*), echipat cu senzori pentru:

* percepţia mediului;
* elemente de execuţie;
* un sistem de conducere pentru maparea percepţiei în acţiune.

Percepţia, acţiunea şi aplicaţia între ele sunt cele trei aspecte majore care definesc un robot mobil. În funcţie de modul de organizare a acestor trei aspecte funcţionale, se poate considera că robotul mobil are un grad mai mare sau mai redus de autonomie.

Cel mai cunoscut tip de robot mobil este vehiculul ghidat automat A*GV - Automated Guided Vehicle*. AGV[[4]](#footnote-4) utilizate pe scară larga în industrie pentru transportul materialelor, se pot deplasa prin mijloace proprii, traseul parcurs de acestea fiind în mod tipic pre-programat. Deoarece astfel de vehicule mobile operează în medii pre-programate, ele sunt inflexibile şi „*fragile*” în operare, orice modificare neprevăzută a traseului (de exemplu, obiecte aflate pe calea de rulare) putând conduce la o compromitere a întregii misiuni.

Alternativa o reprezintă construirea unor roboţi mobili cu autonomie în mişcare.

Autonomia se traduce prin următoarele caracteristici:

* abilitatea robotului mobil de a se deplasa în mediul înconjurător:
  + pentru a îndeplini diferite taskuri;
* abilitatea de a se adapta modificărilor din mediu;
* de a învăţa din experienţă;
* de a-și modifica în mod corespunzător comportamentul;
* abilitatea de a-și construi reprezentări interne ale lumii înconjurătoare care să poată fi utilizate pentru procese de luare de decizie (ex. navigaţia).

Aceste caracteristici permit utilizarea acestor structuri în aplicaţii specifice, dintre care pot fi amintite operaţiuni de transport, explorare, supraveghere, orientare, inspectare, evoluţie în medii inaccesibile sau ostile operatorului uman (de exemplu: roboţi destinaţi combaterii acţiunilor teroriste, roboţii care operează în medii contaminate, roboţii subacvatici, vehiculele de explorare planetară), etc.



Fig. 1‑3 Google autonomous vehicle

<http://www.comunic.ro/article/sistemele-de-inteligen%C5%A3%C4%83-artificial%C4%83-ale-ma%C5%9Finilor-autonome-ale-google-pot-fi-considerate>

IA este un domeniu complex bazat pe sisteme expert, algoritmi genetici, sisteme fuzzy, sisteme hibride și neuronale. Întrucât sistemele de azi utilizează date și informații care trebuie analizate în timp real, IA vine în întâmpinarea acestui fenomen. Apariția mașinilor care învață, înțeleg și trag concluzii ne determină să acceptăm faptul că IA este deja un lucru fără de care azi nu se mai poate evolua.

Deontologia inginerilor nu mai este suficientă azi. Domeniul este unul deja transdisciplinar. Cunoașterea în domeniu este depozitată pe mașini inteligente care pot funcționa în mod autonom, astfel încât accesul la informație trebuie să fie efectuat pe baza unor proceduri foarte clare.

Componentele care permit sistemului inteligent să efectueze inferențe și să elaboreze concluzii, soluții, decizii etc. necesită algoritmi care să permită IA să rașioneze pe baza datelor deja adunate.

Pe lângă puterea de calcul, capacitatea de stocare, de căutare sistemele trebuie să reproducă într-un anumit fel capacitățile omului: să văadă, să audă, să miroasă, să vorbească, să înțeleagă, să valideze etc. Prin urmare domeniul IA migrează spre sisteme robotizate. Acestea din urmă fiind cele mai vizibile din punct de vedere al modului de operare. Spre exemplu sistemele expert din domeniul economic sunt utilizate, în schimb ele nu sunt percepute ca niște elemente de IA, mai mult ca un sistem de calcul și atât.

Atât timp cât roboții sunt construiți să poată învăța, să lucreze în mod autonom, înseamnă că principiile educaționale ale omului trebuie să sufere transformări radicale. Dinamica acestor sisteme dotate cu IA vor suplini nevoile de creație ale omului, fapt deja evident, a devenit un profesor virtual pentru multe domenii economice, dar mai ales s-a reușit stabilirea unei legături între reprezentarea acestor procese, prin intermediul calculatorului, robotilor sau a altor masini inteligente și percepția umană asupra mediului real.

Domeniile științifice care stau la baza conceptului de IA sunt următoarele:

1. Filosofia: logica, metode de raționament, mintea ca sistem fizic bazat pe învățare, limbaj, rațiune;
2. Matematica: reprezentare formală și algoritmi de dezvoltare, calcule, incertitudine, probabilități;
3. Economia: utilizare, arbori de decizie;
4. Neurologia: substrat fizic pentru activitatea mentală;
5. Psihologia: fenomenul de percepție, tehnici de experimentare;
6. Ingineria calculatoarelor: construcția de calculatoare hiper-rapide;
7. Teoria controlului: sisteme de maximizare a unei funcții obiectiv de timp;
8. Lingvistica: gramatică, reprezentarea cunoștințelor.

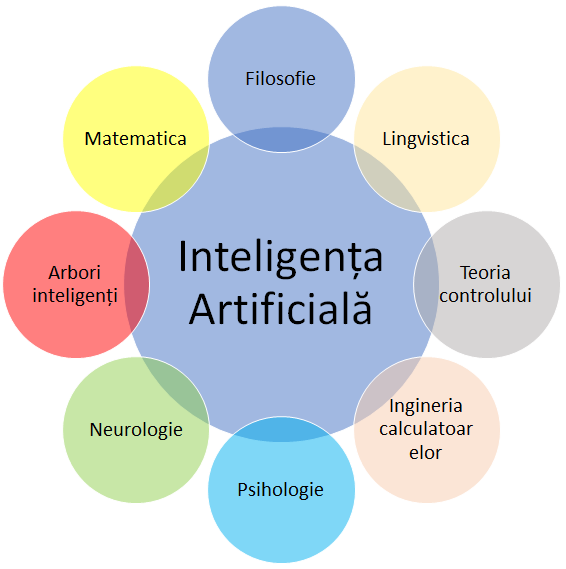


Fig. 1‑4 Domenii de bază ale IA

<http://www.rasfoiesc.com/educatie/informatica/calculatoare/Sisteme-Expert-Inteligenta-Art49.php>

IA se bazează pe un aparat matematic foarte bine dezvoltat. Utilizează noțiuni precum:

logica matematică:

calculul propoziţional;

calculul cu predicate;

teoria inferenţelor logice;

teoria probabilităţilor, reprezentarea cunoaşterii incerte spre exemplu cu ajutorul reţelele Bayesiene prin explicitarea dependenţelor condiţionale între diferite componente;

teoria algorimilor de căutare, de control;

reţele neuronale:

recunoaşterea formelor;

sinteza vorbirii;

lingvistica matematică înţelegerea limbajului natural, traduceri;

teoria grafurilor:

reprezentarea tip arbore.

Știința și tehnologia se schimbă rapid. "Vechile" științe, cum ar fi fizica, sunt relativ bine înțelese, calculatoarele sunt omniprezente

Provocări majore în domeniul științei și tehnologiei:

* înțelegerea creierului: raționament, cogniție, creativitate;
* crearea de mașini inteligente.

Este posibil?

* care sunt provocările tehnice și filosofice?
* AI pune cele mai interesante provocări și întrebări din domeniul informaticii de azi.

Ce este AI:

* INTELIGENȚA: "Capacitatea de a învăța și rezolva problemele" (dicționarul Webster) în special: abilitatea de a rezolva probleme noi; abilitatea de a acționa rațional; capacitatea de a acționa ca oamenii.
* INTELIGENȚA ARTIFICIALĂ[[5]](#footnote-5): construiește și înțelege entități inteligente sau agenți inteligenți și se bazează pe 2 abordări principale: „*inginerie versus*” „*modelare cognitivă*”.

Nu există o definiție solidă a inteligenței care să nu depindă de faptul că este legată de inteligența umană? Problema este că încă nu putem caracteriza în general ce fel de proceduri computaționale vrem să le numim inteligente. Înțelegem unele dintre mecanismele de inteligență și nu altele.

Aplicațiile AI în domeniul Infograficii sunt foarte multe.

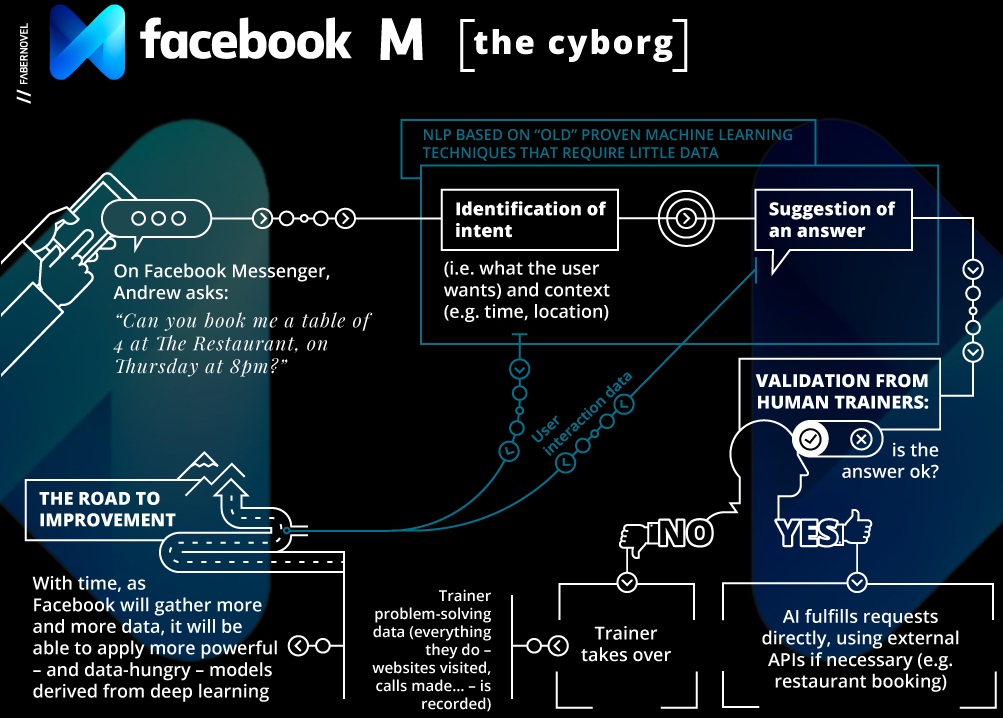


Fig. 1‑5 Aplicație AI în domeniul socializării Facebook.

<https://en.fabernovel.com/insights/tech-en/engines-of-ai>

Revenind la tehnica realizării de instrumente dotate cu AI vom continua cu o descriere succintă a rețelelor neurale.

În principiu o reţea neurală este caracterizată de ansambluri de elemente de procesare simple, care sunt interconectate şi care operează, din punct de vedere al calculului, în paralel. Aceste instrumente trebuie să interacţioneze cu mediul înconjurător concomitant cu dezvoltarea de capacități de învățare MachineLearning, DeepLearning etc.

Metodologia reţelelor neurale facilitează, putem spune, proiectarea, realizarea și dezvoltarea de sisteme neliniare cu *m* – intrări și *n* – ieșiri, relaţiile dintre instanţele de tipul intrare-ieşire ca perechi de vectori caracteristici și structurali . Vectorul de intrare conţine măsurători ale caracteristicilor unui domeniu de interes, iar ieşirea dorită poate reprezenta clasele structurii, valori prezise sau valori ţintă.

Utilizarea rețelelor neurale poate fi descrisă, astfel:

* se dă o mulțime de date:

1.1

* se urmărește un obiectiv și anume acela de a determina o ipoteză *h* – funcție de predicție, dintr-un anumit spațiu al ipotezelor;
* aplicația trebuie să îndeplinească – funcția de predicție *h* prezice o etichetă corespunzătoare vectorului caracteristicilor .

Extragerea datelor din mulțiemade date (date sau antrenare) reprezintă un algoritm de învățare, care urmărește modificarea ponderilor de transfer la nivelul așa ziselor sinapse (zona de transfer de date) pentru a putea atinge obiectivul droit.

Algoritmii de învățare se impart în două mari categorii:

1. învăţarea supervizată, care presupune aplicarea unei intrări reţelei, după care se compară ieşirea produsă de reţea cu ieşirea dorită şi se modifică ponderile astfel încât să se minimizeze diferenţa dintre cele două:
   1. antrenarea de tip bach constă în modificarea ponderilor şi a bias-ului presupunând ca fiind dată o mulţime de vectori de intrare;
   2. antrenarea incrementală modifică ponderile şi bias-ul unei reţele imediat după introducerea unui vector de intrare, antrenarea incrementală poartă numele şi de antrenare „on-line” sau „adaptivă„;
2. învăţarea nesupervizată, în care mulţimea de antrenare constă numai din vectori de intrare, scopul algoritmului fiind de a produce vectori consistenţi, în sensul că două semnale foarte apropiate să producă răspunsuri identice/asemănătoare.

Rețelele neurale sunt caracterizate de trei elemente principale:

* model adoptat pentru elemental de procesare individual;
* structura interconexiunilor - arhitectura;
* reguli de ajustare a legăturilor – algoritmii de învățare.

Perechile (vector intare, vector ieşire) similare sunt grupate în clustere.

Fig. 1‑6 Modelul neural și arhitectura unei rețele.

## State of the Art

În [1] este prezentată evoluția principiilor de Inteligență Artificială, plecând de la primul soft care a putut învinge într-o partidă de șah un maestru de șah – concurs orgnizat în anul 1989 între maestrul intenațional Arnold Denker și robotul HITECH. Detalii au fost publicate de H.J. Berlner[[6]](#footnote-6).

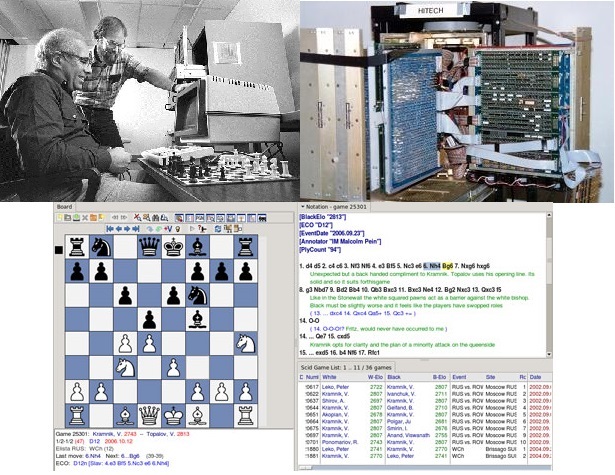


Fig. 1‑6 Arnold Denker în competiție cu robotul HITECH

<https://chessprogramming.wikispaces.com/HiTech>

Un alt exemplu de utilizare a IA în folos comunitar îl reprezintă programul de vorbire PEGASUS, folosit în vederea punerii în vânzare a biletelor de tren. Clientul vorbește la un microfon și solicită un bilet pentru o destinație . Robotul funcționează astfel:

* analiza vorbirii;
* răspuns pentru înțelegere;
* căutare baze de date;
* oferta propusă.

Algoritmul aparent este simplu, dar este posibil să ofere un preț foarte mare, sau un preț mic și un timp foarte lung. Software-ul de analiză a vocii introduce un set de erori, astfel încât este posibil să se interpreteze greșit informația primită de la om. Elaborarea, dezvoltarea și analiza secvențială (pe segmente) a problemelor de identificare automată lingvistică (automatic language identification) LID se bazează pe analiză fonetică, acustică, informații prosodice asamblate cu un model matematic pe baze probabilistice unificat. Apoi pe baze empirice se stabilesc contribuțiile relative ale diferitelor surse de informații concomitent cu stabilirea și asigurarea mecanismului de combinare a acestora.

Există o altă situație, cum ar fi, centrul de comandă a zborului pentru avioane de linie. Apariția unui semnal de atenționare (culoare roșie ex.) sugerând ideea unei defecțiuni. Faptul că analistul programator este atent, el va putea interveni preluând comanda automată și trecând pe modul manual.

Astăzi se utilizează sistemul expert în timp real MARVEL, care este capabil să monitorizeze un flux masiv de date.

Vehiculele autonome cu personal uman la bord, reprezintă o altă formă de aplicație a IA. În acest caz viteza de transfer a datelor trebuie să fie foarte mare, iar capacitatea de calcul de asemenea foarte mare, întrucâ sistemele de senzori trebuie să acumuleze cât mai multe date din mediul exterior și să poată interpreta imaginile: stopuri, intersecții, linie continuă, pietoni, mașinile din trafic, prognozarea diferitlor posibile traiectorii ale celorlalte mașini etc.

Roboții echipați cu vedere artificială[[7]](#footnote-7), utilizează așa zisele sisteme „servoing visual”. Acestea folosesc aplicații de proiectare în timp real pentru a putea controla traiectoriile mișcării roboților cu funcții de manipulare. Imaginile recepționate sunt definite ca proprietăți ale obiectelor care compun o imagine și se pot achiziționa cu ajutorul unor senzori de vedere EO/IR, senzori montați fie pe brațul robotizat fie pe mecanismul efector al brațului. Din această configurație se nasc două arhitecturi fundamentale ale tehnicii „servoing visual”:

* ***arhitectură bazată pe poziţie*** – bazată pe calculul erorilor reprezentate în sistemul cartezian și care necesită obligatoriu un model virtual al obiectului;
* ***arhitectură bazată pe imagine*** – bazată pe utilizarea unei camere perfect calibrate pentru a obține poziția și orientarea obiectului.

Tehnica „servoing visual” este de fapt o soluție hibridă, flexibilă și robustă pentru a crește acuratețea și versatilitatea unei arhitecturi de control, bazată pe feedback visual [2].

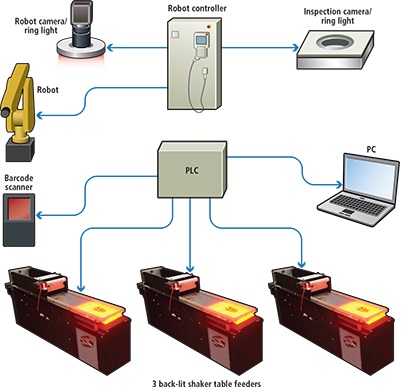


Fig. 1‑7 Controlerul robotului: un PLC master și două PLC slave

<http://www.vision-systems.com/articles/print/volume-20/issue-8/features/robot-vision-automates-connector-assembly.html>

Controlul mișcării brațului unui robot manipulator, dintr-o poziție inițială într-una finală, prin intermediul unui sistem „*servoing visual*” se face având ca informații datele provenite de la senzori vizuali. Mișcarea, din punct de vedere matematic, implică definirea unui spațiu de configurare al task-urilor (*stărilor*) [3]:

1.1

Adaptarea mișcării brațului la un sistem de referință se efectuează prin transformarea cordonatelor (vectoriale de cele mai multe ori) de la un sistem de referință la altul.

Ulterior, achiziționării de imagini, începe procedura de extragere a caracteristicilor vizuale. Senzorii de achiziție date sunt de mai multe tipuri: convenționale, cu ultrasunte și omnidirecționale [4].

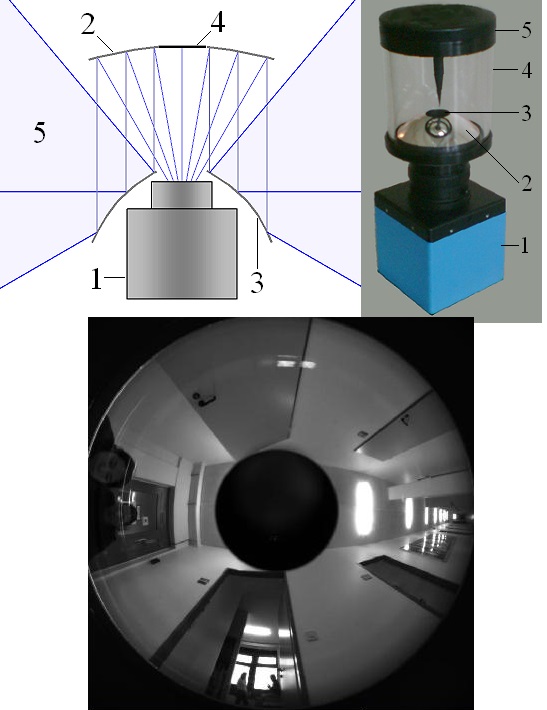


Fig. 1‑8 Schema funcțională a unei camere video omnidirecționale

1. Camera; 2. Oglinda superioară; 3. Oglinda inferioară; 4. Pata neagră; 5. Câmpul visual.

<https://www.revolvy.com/page/Omnidirectional-camera>

Descompunerea imaginilor în vectori specifici unor trăsături, reprezintă cheia de boltă a introducerii IA în prelucrarea imaginilor. Întrucât noțiunea de IA a fost împinsă către roboții autonomi, înseamnă că trebuie realizată o corelație între preluarea imaginilor, mișcările senzorului, mișcările robotului și trăsăturile care se doresc a fi sintetizate. Trăsăturile pot fi 2D sau 3D și în funcție de senzor se apelează la instrumente matematice diferite. Cea mai convenabilă soluție o reprezintă soluția hibridă 2D/3D. Se extrag trăsăturile 2D pentru care se definesc: coordonatele față de sistemul de referință a căror reprezentare geometrică poate fi sub forma unor linii, elipse contururi. De asemenea pentru o acuratețe mai mare a prelucrării se introduce și noțiunea de trăsătură de moment. Astfel se introduce și a patra dimensiune – timpul. Proiectarea unui sistem servoing pe baza celor descrise mai înainte, permite minimizatrea neliniarităților introduse de matricea de interacțiune.

Trăsătura geometrică a unui punct de interes se poate asimila unui punct dintr-o imagine, care este bine definit geometric, astfel încât, el poate fi integrat sau izolat față de linia, curba, elipsa etc. din care a fost preluat. Pe de altă parte dacă imaginea prelevată reprezintă doar un mic colț din ansamblu, acest colț se va defini ca o intersecție a două muchii. Instrumentele de detectare (aparatul matematic) nu este foarte robust, de aceea analiza implică suprapunerea de date care diferă prin: luminozitate, translație, rotație sau altele. Două dintre cele mai uzuale sisteme de detectare a trăsăturilor sunt: *operatorul Harris* și *detectorul SIFT*.

Mitsubishi Electric Corporation a dezvoltat o tehnologie, bazată pe modele AI, pentru controlul autonom[[8]](#footnote-8) al echipamentelor. Această tehnologie MAISART[[9]](#footnote-9) constă în:

* proiectare și realizarea echipamentului propriu-zis;
* testare și introducerea de erori repetate;
* apoi sistemul este pus să învețe regulile de control bazate pe aceste modele.

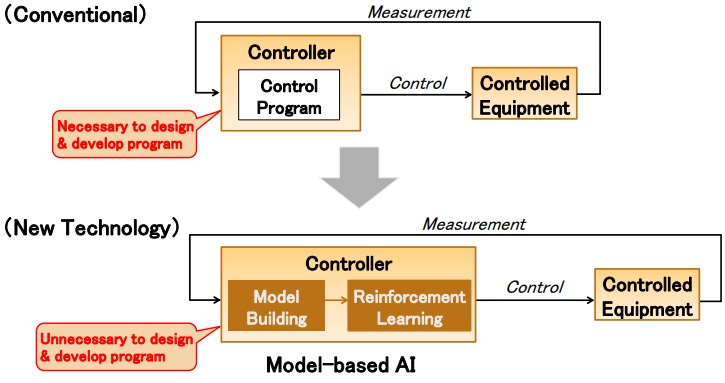


Fig. 1‑9 Principiul functional al MAISART.

<http://www.mitsubishielectric.com/news/2018/pdf/0214-b.pdf>

Principiul descris în Fig. 1‑9 este o tehnologie AI care învață regulile de control în mod automat. Acest model, pentru dinamica obiectelor 3D, învață prin încercări și erori succesive, pe baza cărora regulile de control necesare atingerii obiectivelor predefinite, sunt învățate în mod autonom.

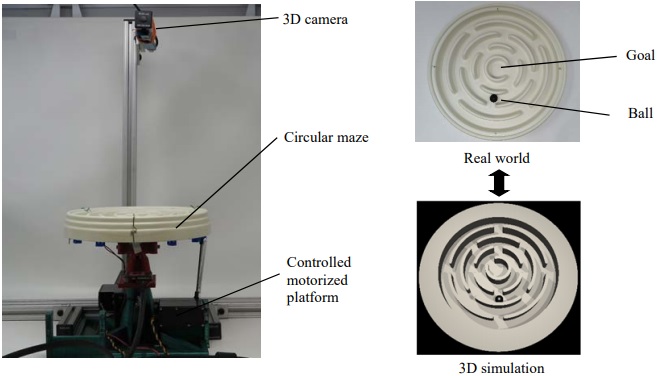


Fig. 1‑10 Labirintul circular folosit pentru demonstrarea tehnologiei MASIART.

<http://www.mitsubishielectric.com/news/2018/pdf/0214-b.pdf>

Una dintre probele folosite pentru testatrea echipamentelor se referă la conducerea unei mingii într-un labirint. Labirintul este de formă circulară (Fig. 1‑10) și în timp ce mingea este împinsă pe culoarele labirintului platforma acestuia este înclinată. Mingea este împinsă de la exterior către centrul labirintului (care este de formă circulară). Prin algoritmii introduși s-a reușit conducerea mingii fără a fi nevoie de intervenția umană, respectiv fără a fi nevoie de a reprograma software-ul.

## Operatorul Harris

IA urmărește ca elementele de achiziție date, de prelucrare și de interpretare a acestora să permită unui sistem (fie el robotizat) să poată prelua o parte din funcțiile omului și de a le transpune în elemente ce pot facilita unui operator uman decelarea informațiilor despre un fenomen.

Operatorul Harris se bazează pe utilizarea funcţiei de auto-corelaţie. Acest algoritm a fost propus pentru prima dată de Moravec[[10]](#footnote-10), îmbunătăţit de către Harris şi Stephens (1988). Algoritmul Harris de detecție a punctelor de interes este format din două etape:

* prima constă în calcularea valorilor funcţiilor de autocorelaţie pentru fiecare pixel din imagine;
* a doua constă în determinarea valoarilor de maxim local a acestei funcţii intr-o vecinătate definită de utilizator.

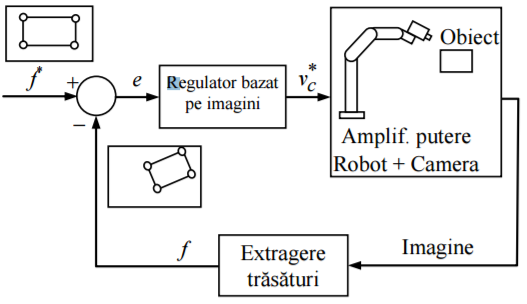


Fig. 1‑10 Schema funcțională a unui system servoing

<http://www.ace.tuiasi.ro/users/103/2009-Burlacu_Adrian_PhD.pdf> , fig. 2.1, pg. 20

Identificarea unor puncte de interes (pixelii asociaţi valorilor de maxim local) cu anumite proprietăţi bine definite:

* număr redus în comparaţie cu trăsăturile tip muchie;
* invarianţă la:
  + rotaţie şi a unor mici fluctuaţii de iluminare;
  + modificări liniare ale factorului de scalare;
  + transformări afine.

Principiul de detecţie se bazează pe analiza proprietăţilor funcţiei de autocorelaţie locală pentru o poziţie generică și a unei funcții imagine , a cărei matrici de auto-corelație se determină astfel:

, 1.2

unde: – reprezintă nucleul Gaussian; – reprezintă gradienții funcției imagine pe direcțiile :

, 1.3

Prin urmare detecția punctelor de interes se va efectua cu ajutorul relației următoare:

, 1.4

unde: – determinantul matricei A; - parametru de acordare; – urma matricei A.

Se poate constata la o primă evaluare, că o valoare mare a punctului de interes corespunde unor valori mari ale valorilor proprii. Un colț este detectat dacă matricea A este de rang 2 și ar valori semnificative.

În Fig. 1‑11 se disting trei cazuri:

* ***regiunea omogenă*** – valori proprii mici;
* ***regiunea marginilor*** - valorile proprii una mare iar cealaltă mică;
* ***regiunea colțurilor*** - dacă ambele valori proprii sunt mari.

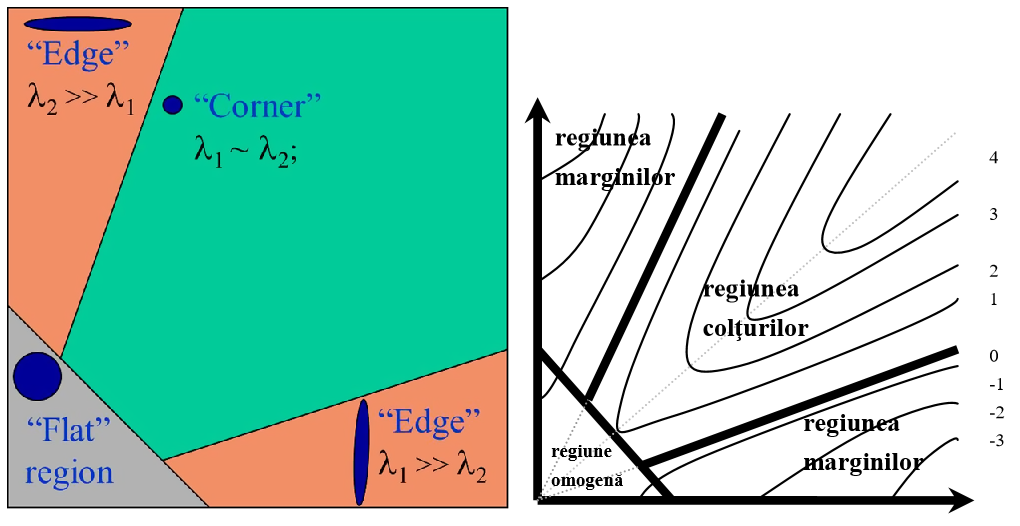


Fig. 1‑11 Principiul funcției de autocorelație

<http://opticalengineering.spiedigitallibrary.org/article.aspx?articleid=2595182>

Valoarea „***de colț***” pentru fiecare pixel în parte se calculează cu relația:

, 1.5

unde: – sunt gradienții funcției imagine pe direcțiile , gradienți supuși unui filtru cu nucleu Gaussian:

. 1.6

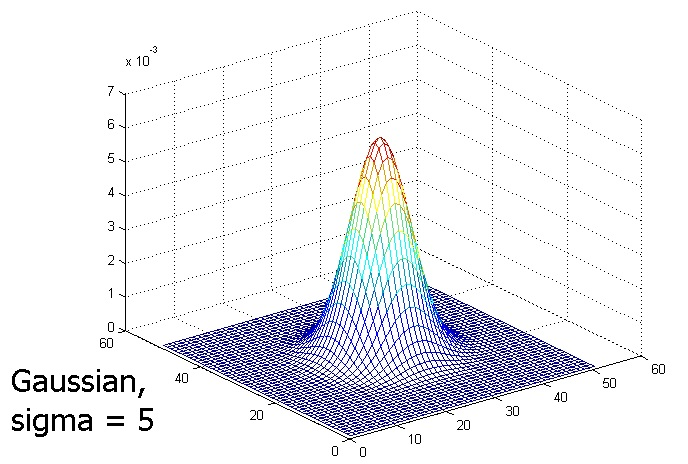


Fig. 1‑9 Aplicarea filtrului cu nucleu Gaussian pentru

## Detectorul SIFT

Trăsăturile punctiforme extrase cu algoritmul Harris își păstrează proprietățile doar în cazul secvențelor de imagini cu același factor de scalare[[11]](#footnote-11). Secvențele de imagini au în general un factor de scalare variabil astfel, punctele de interes detectate trebuie să includă proprietăți de invariață la scalare.

Algoritmul SIFT (Scale Invariant Features Transform) propus de Lowe (Lowe, 1999; 2003), se compune din patru etape:

1. detectarea extremelor în spaţiul scalării;
2. localizarea trăsăturilor;
3. determinarea magnitudinii și a orientării pentru fiecare trăsătură;
4. crearea descriptorului.

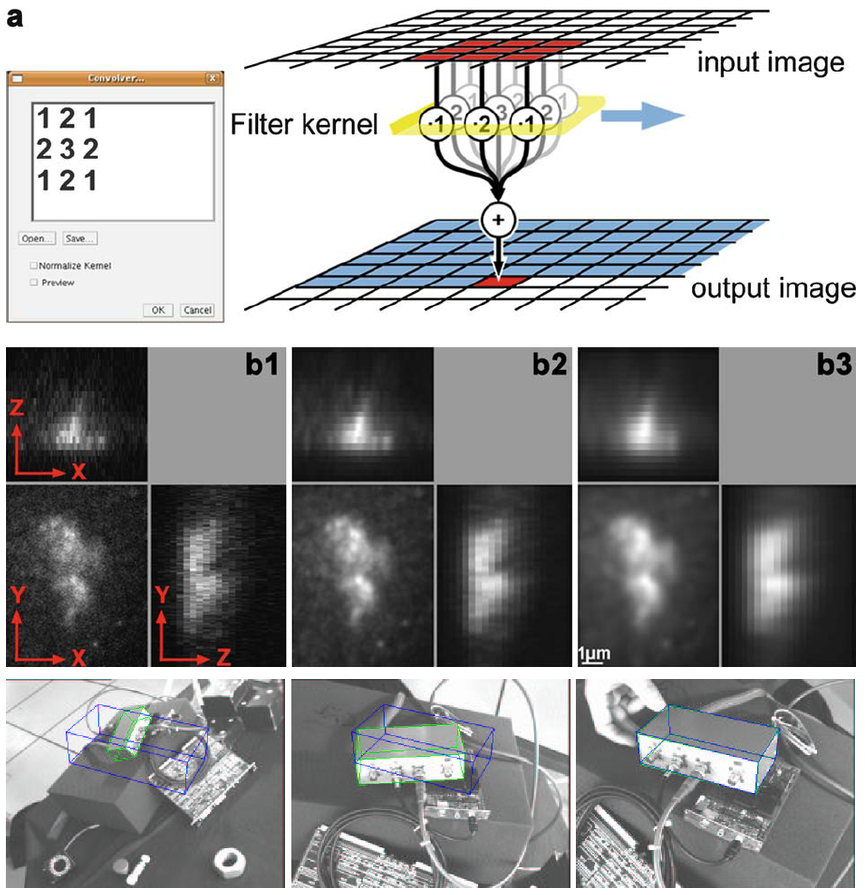


Fig. 1‑10 Aplicarea unui filtru liniar și a unuia Gaussian

<https://www.researchgate.net/figure/5392508_fig3_Figure-3-Linear-filtering-and-Gaussian-filter-a-Kernel-of-a-2D-linear-filter>

În Fig. 1‑10 avem reprezentarea unor filtrări liniarea și Gaussine:

1. ***filtru liniar 2D*** - intensitatea unui pixel (marcat cu roșu) în imaginea de ieșire este determinată de intensitățile acesteia și a pixelilor învecinați (marcați cu roșu) în imaginea de intrare. Matricea din partea stângă prezintă ponderile pentru pixeli. Intensitatea de ieșire a pixelilor țintă este suma intensităților ponderate ale pixelilor acoperite de către nucleu;
2. ***filtru gaussian*** - filtrarea Gauss pentru înlăturarea neclarității presupune o nouă focalizare a imaginii, netezirea față de distribuția Gauss și o reducere a zgomotului, astfel încât se poate obține o anume claritate, dar detaliile fine sunt eliminate.

De aceea s-a propus o altă metodă de filtare SIFT, care după cum se observă în paragrefele anterioare, urmează următorul algoritm: primele două etape detectează expremele din spațiul scalărilor și localizează cu acuratețe trăsăturile, iar următoarele două etape asignează orientarea și descrierea trăsăturilor.

Identificarea unor poziţii în spaţiul scalărilor ce reprezintă puncte de extrem ale unei anumite funcţii se bazează pe găsirea trăsăturilor stabile în toate nivelurile de scalare. Spaţiul scalărilor se construieşte folosind o succesiune de filtrări cu un anumit nucleu [4]. S-a demonstrat că singurul nucleu care se poate folosi pentru generarea unui spaţiu tridimensional al scalărilor este nucleul Gaussian [5]:

, 1.7

unde: – reprezintă dispersia nucleului Gaussian.

Astfel, spațiul scalărilor , este definit ca un produs dintre funcția imagine și nucleul Gaussian :

. 1.8

Prin urmare, funcția spațiului scalărilor va genera o succesiune de filtrări ale imaginii dependente de parametrul considerat a fi o funcție , definită:

, 1.9

unde: – una dintre octavele axei din spațiul scalărilor; – numărul de niveluri ale fiecărei octave; – indexul nivelului din octava care se poate defini ca spațiul al scalărilor.

Spațiul scalărilor pentru o imagine se obține în urma unor filtrări succesive și a unor modficări de asemenea succesive ale dimensiunilor imaginii:

1. se consideră imaginea ;
2. se calculează ;
3. se determină imaginea ;
4. atunci când , imaginea în noua octavă se redimensionează.

În momentul în care spațiul scalărilor este complet. În continuare se urmărește constuirea spațiului diferențelor de filtrări Gaussiene DoG:

. 1.10

Din punct de vedere matematic DoG este obținut prin derivarea spațiului scalărilor. În acest spațiu DoG se vor identifica poziții

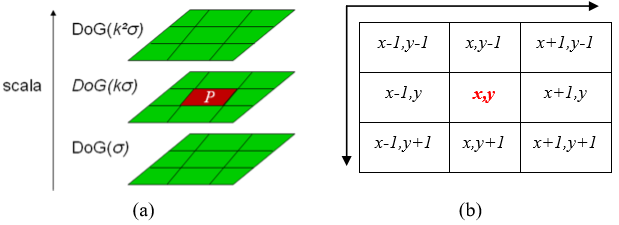


Fig. 1‑11 (a) Reprezentarea scalei pentru 3 niveluri ale unei octave;

(b) Fereastra de 3x3x3 din jurul punctului P

Pg. 15, fig. 2.2 [3]

Totalitatea punctelor de extrem formează mulțimea extremelor locale ale funcției :

. 1.11

Dezvoltarea Taylor până la elementul pătratic permite determinarea prin SIFT a offset-ului funcției :

, 1.12

unde: , iar determinarea extremelor funcției se obține prin egalarea cu zero a derivatei funcției (1.12):

. 1.13

Din (1.12) și (1.13) rezultă:

. 1.14

Pentru implementarea algoritmului de eliminare a muchiilor se aplică matricea Hessian:

. 1.15

Odată ce au fost detectate extremele în spațiul scalărilor și sau localizat trăsăturile pentru determinarea punctelor de interes, cu ajutorul descriptorului SIFT se vor determina orientarea și magnitudinea gradientului, pe baza proprietăților locale ale funcției imagine.

. 1.16

Putem aprecia că histograma orientărilor este formată din valorile gradientului orientărilor punctelor de interes (Fig. 1‑12).

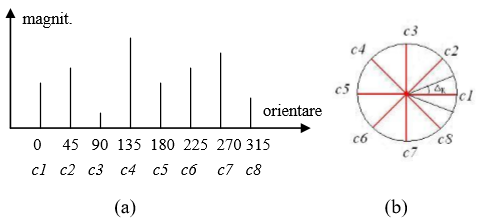


Fig. 1‑12 (a) Histograma orientărilor;

(b) Orientările fundamentale

Pg. 18, fig. 2.3 [3]

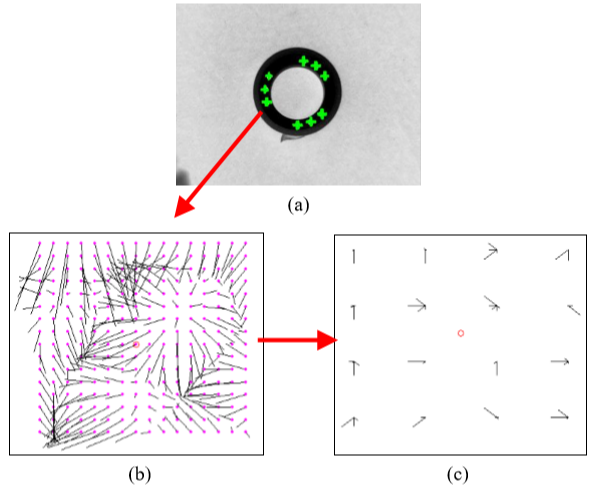


Fig. 1‑13 (a) Trăsăturile obiectului detectat cu SIFT;

(b) Orientarea și magnitudinea; (c) Descriptorul trăsăturii

pg. 18, fig. 2.4 [3]

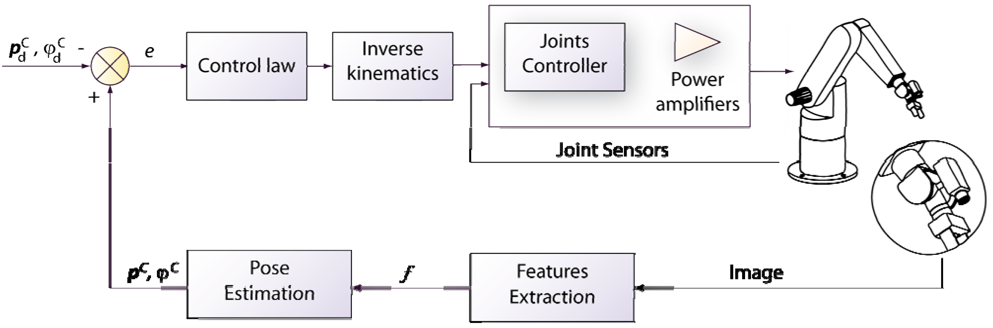


Fig. 1‑14 Sistem visual servoing pentru controlul roboților

## Inteligența artificială distribuită

Definiția largă a IA consideră că pentru a putea dezvolta sisteme complementare inteligenței umane este necesară crearea de sisteme S/W (software/hardware) autonome, sisteme asimilate agenților inteligenți. Altfel spus IA presupune interconectarea agenților inteligenți ca părți ale unui sistem. Se poate afirma, în baza celor spuse, că IA este proprietatea agenților inteligenți de a se interconecta.

Inteligenţa artificială distribuită se ocupă de studierea, dezvoltarea şi utilizarea de sisteme S/W în care mai mulţi agenţi inteligenţi urmăresc o mulţime de scopuri şi execută o mulţime de sarcini pentru a atinge respectivele scopuri.

Astăzi paradigmele, cum ar fi de descompunerea problemelor în subprobleme și apoi de repartizare pe module de intereracțiune, sau care utilizau metoda BlackBoard, au fost înlocuite de următoarele sisteme:

* *sisteme multiagent* în care mai mulţi agenţi îşi coordonează cunoştinţele şi activităţile, raţionând asupra procesului de soluţionare a unei probleme;
* *sisteme distribuite* pentru rezolvarea de probleme, în care o problemă particulară este soluţionată prin împărţirea ei în mai multe sarcini mai multor noduri similare, care îşi divid şi partajează cunoaşterea despre problemă.

O altă preocupare a Inteligenţei Artificiale Distribuite constă în proiectarea de arhitecturi paralele dedicate, care permit creşterea performanţei sistemelor inteligente distribuite, în care fiecare agent este privit ca un nod fizic dintr-o reţea de calculatoare.

Dezvoltarea infrastructurii informaţionale, permite sistemelor informatice să fie distribuite, mari, deschise, eterogene şi complexe.

IA distribuită se poate asimila, în funcție de mediul de lucru, cu o *inteligenţă colectivă*:

* agenții inteligenți nu pot interacţiona cu mediul lor şi cu alte programe inteligente;
* problemele reale sunt extrem de complexe;
* complexitatea programelor este simplificată prin module cuplate;
* programarea orientată pe obiecte.

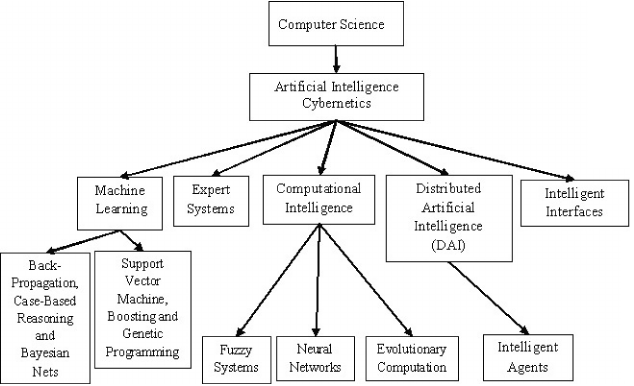


Fig. 1‑15 Diagrama schematică a clasificării IA

<https://www.researchgate.net/publication/232914399_Artificial_intelligence_for_energy_conservation_in_buildings>

Generarea de sisteme de agenți software, roboți, senzori, sisteme de calculatoare, care lucrează împreună cu aceeași eficiență și expertiză este foarte utilă în vederea rezolvării multiplelor provocări ale omului:

* căutare și salvare;
* activități militare;
* tehnologia jocurilor;
* securitatea calculatoarelor;
* securitatea comunicațiilor;
* transport;
* logistică;
* etc.

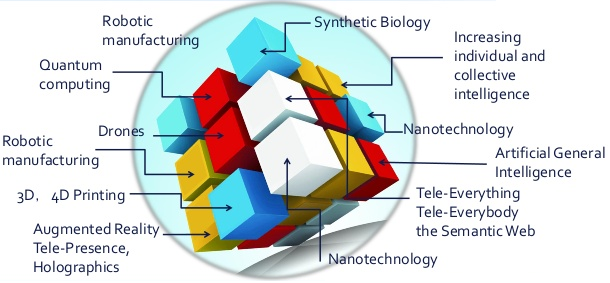


Fig. 1‑16 Diagrama schematică a clasificării IA

<http://www.slideshare.net/JeromeGlenn/wfs-pdf-july-2015>

Sistemele formate din entități multiple, uneori redundante, oferă posibilități de creștere a robusteței și fiabilității soluțiilor, datorită abilității pe care o singură entitate poate să o preia de la o altă entitate lipsă. Putem spune că se creează entități monolitice capabile să rezolve probleme cu costrui reduse și de reducere a complexității entităților individuale. Inteligența distribuită [6] are și unele dezavantaje, cum ar fi, în acest moment:

* lipsa unui control centralizat sau al unui depozit centralizat de informații la nivel mondial;
* necesita mai multă comunicare pentru a coordona toate entitățile din sistem;
* creșterea numărului de entități poate duce la creșterea interferenței între entități, deoarece acestea trebuie să acționeze fără o cunoaștere completă a intențiilor celorlalte entități;
* sistemele de mai multe entități experimentează o incertitudine crescută cu privire la starea sistemului în ansamblul său.

Provocarea este de a proiecta un sistem, astfel încât să se obțină o coerență globală prin interacțiunile locale ale entităților individuale.

Paradigmele inteligenței artificiale distribuite diferă de la un tip de interacțiuni la altele, acționând conform reprezentării din Fig. 1‑17, dacă entitățile au gradul de conștientizare a altora cu privire la ansmablu și dacă acțiunile unei entități au un avans față de obiectivele altor entități. Tipurile de obiective sunt:

* fiecare entitate are scopuri individuale;
* entitățile au obiective comune.

Conștientizarea axei altor entități se realizează prin împărțirea sistemelor în entități conștiente și care nu sunt conștiente. Conștiența se referă la fapul că motivul de acțiune și intențiile entităților diferă. Roboți care nu au conștiența mediului pot sesiza prezența entităților locale și astfel pot acționa fără să înțeleagă intenția sau planurile de viitor ale sistemului. Această metodă de funcționare se numește „*stigmergy*”[[12]](#footnote-12) (*mecanism de coordonare indirectă, prin intermediul mediului, între agenți sau acțiuni - este o forma de auto-organizare, produce structuri complexe, aparent inteligente, fără să fie nevoie de planificare, de control sau de comunicare chiar directă între agenții, sprijină colaborarea eficientă între agenți, care nu dispun de memorie, inteligență sau chiar conștiență individuală*).

Segmentările spațiului de domeniu sunt aproximative, dar utile în înțelegerea tipurilor primare de interacțiuni care pot apărea în aplicațiile tipice. Diferite zone ale acestui subspațiu reprezintă tipuri comune de interacțiuni observate în sistemele de inteligență distribuite:

1. colectiv;
2. cooperativa;
3. colaborativ;
4. coordonator.

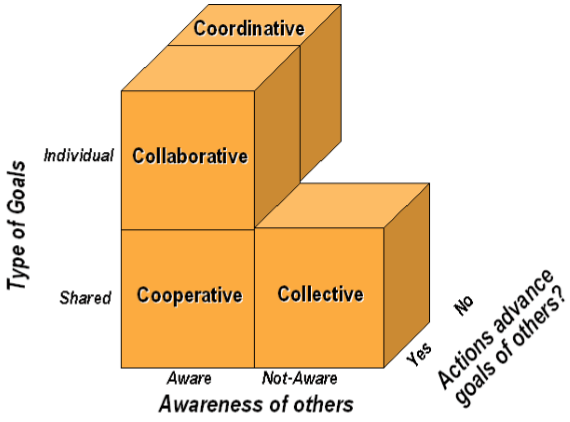


Fig. 1‑17 Tipuri de interacțiuni în sistemele de inteligență distribuite [6] fig 2. pag. 7

Paradigmele pentru inteligența artificială diferă prin modul de abstractizare a spațiului, astfel încât vizualizarea sistemului să poată genera strategii de proiectare-dezvoltare adecvate. Paradigmele diferă și din punct de vedere al semanticii, putând spune că nu toate pot fi aplicate în orice domeniu. Spre exemplu cele specifice interacțiunilor dinamice nu pot fi aplicate în domeniul lingvistic, etc.

Cele mai comune paradigme pentru inteligență distribuită sunt:

* bioinspired - sisteme emergente, algoritmi genetici etc.;
* organizaționale și sociale;
* bazate pe cunoaștere - ontologice, semantice.

Ca exemplu se prezintă cooperarea unor roboți mobili [7]. În Fig. 1‑18 grupul de UTVs se află în proximitatea unui obiectiv în vederea obținerii de informații și date privind: dimensiunile, localizarea terestră etc. UTVs comunică între ele printr-un LAN, fiecărui robot fiind-ui atribuit un interval de timp pentru a transmite date.

Dacă un vehicul devine inoperant, sistemul LAN relochează către unul dintre roboții aflați în funcțiune, ativitățile robotului devenit inoperant. UTVs operașionale continuă comunicarea și sarcinile de generare a traiectoriilor și de control [8]. Diferitele tipuri de comportament coordonat includ dispersie, lucrare în rețea de calculatoare (clustering), orbitare în jurul unui punct de referință, până la o nouă reorientare și așa mai departe.



Fig. 1‑18 Robotic All Terrain Lunar Exploration Rovers fig. 1, pag 17

<https://www.osti.gov/scitech/servlets/purl/10196260/>

Cooperare între UTVs (Fig. 1‑19), există și sub o formă de organizare tip progresie liniară. Acest model de cooperare se utilizează pentru acțiuni complee de identificare terestră, care includ: dispersia, orbita/traiectoria, clustere cu reguli predefinite. Cele de mai înainte permit informațiilor adunate de fiecare robot să fie distribuite și celorlalți roboți.



Fig. 1‑19 Exemplu de căutare a unui grup de Robotic All Terrain Lunar Exploration Rovers fig. 2, pag 17

<https://www.osti.gov/scitech/servlets/purl/10196260/>

Configurările, spațiale, trebuie modelate numeric, astfel încât, (Fig. 1‑20) algoritmul IA cu care este înzestrat controlerul să poată efectua simulări continue. Necesitatea acestui algoritm este data de faptul că roboții trebuie să se distribuie în arealul de interes, până a acoperirea acestuia – conform posibilităților date de senzorii cu care sunt echipate UTVs-urile. Comunicația RF cu fiecare robot (considerat slave) cât și cu cel (considerat master). Aceasta nu este tot, oricare dintre UTVs poate devein master, dacă cel anterior sufră vreo defecțiune.

Importanța modelării și simulări este data de considerentele ce decurg din realizarea fizică a unui cluster, deoarece cu cât numărul elementelor (roboți) din cadrul sistemului este mai mare cu atât numărul de erori crește. Pe de altă parte există dificultăți tehnice multiple, cheltuielimari privind testarea dinamicii cât și a celei H&W care ar face utilizarea acestor sisteme prohibită.

Simularea poate arăta cum anume pot fi implementate strategiile de control și de comunicare, precum și posibilele probleme de stabilitate și convergență. Fiecare robot este echipat cu un controler care are o strategie proprie și care îl include pe respectivul robot într-o ierarhie comportamentală de grup, după cum urmează:

* menținerea canalului de comunicație RF;
* triangularea locației cu două sisteme de localizare:
  + GPS;
  + senzor de accelerație pe trei direcții;
* detecție pentru evitarea obstacolelor;
* respectarea traiectoriei prestabilite
* etc.

În plus, ansamblul de vehicule are ca strategie de control, preprogramată, comunicarea printr-un master. Simularea oferă de asemenea rezultate grafice ale mișcărilor grupului , precum și date care pot fi analizate.



Fig. 1‑20 Modelarea numerică a dispersiei unui grup de Robotic All Terrain Lunar Exploration Rovers fig. 3, pag 18

<https://www.osti.gov/scitech/servlets/purl/10196260/>

## Agenți inteligenți

Agenţii inteligenți reprezintă entităţi computerizate care acţionează în locul operatorilor umani, care adună și stochează informații, care trimit sau filtrează mesaje de e-mail. Utilizează noțiunea de cuvânt „*cheie*” care în viitorul foarte apropriat va deveni foarte utilă, ajutând la identificarea informațiilor, articolelor, imaginilor sau a unor pachete de date ce pot conține elemente de similitudine.

Agentul inteligent reprezintă acel ceva care acționează într-un mediu. Un agent poate, de exemplu, să fie o persoană, un robot, un câine, un vierme, vântul, gravitația, o lampă sau un program de calculator ce cumpără și vinde. Uneori, agenții acționează numai într-un spațiu de informații și ei sunt numiți tot roboți [10].

Agenții interacționează cu mediul și aceștia de regulă au o structură fizică. Un robot este un agent care are de îndeplinit o misiune. Agenții primesc informații prin intermediul senzorilor lor. acțiunile agentului depind de informațiile pe care le primește de la senzorii săi. Acești senzori pot, sau nu pot, reflecta realitatea. Senzorii pot produce zgomote, pot fi deteriorați, au anumite limitări fizico-chimice-mecanice, astfel încât utilizarea lor la modul general nu este recomandată.

Un agent trebuie să acționeze pe informațiile pe care le are la dispoziție. Adesea, aceste informații sunt foarte slabe, respectiv mărimea fizică, care este de regulă un semnal electric, nu conține destulă informație pentru ca senzorul și elementele de traducere să descrie fenomenul fizic urmărit. Agenții acționează prin intermediul dispozitivelor de acționare – efectori sau actuatori.

Un agent de control este mesajul/comanda pe care agentul o trimite elementelor de acționare. Agenții de multe ori efectuează acțiuni pentru a aduna mai multe informații.

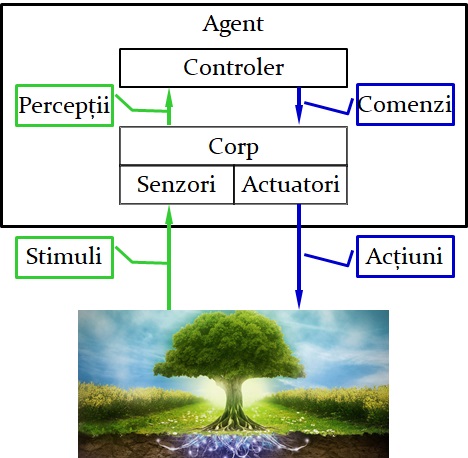


Fig. 1‑10 Interacțiunea între un agent și mediul real

[5] *fig. 2.1*, <http://www.mediaeducationcentre.eu/eng/?p=3393>

În Fig. 1‑10 avem o descriere succintă privind modul în care un agent interacționează cu mediul înconjurător, schema se referă la noțiune de sistem agent-mediu înconjurător. Agentul este alcătuit dintr-un corp și un controler. Controlerul primește percepții din corp și trimite comenzi către corp. Un corp include *senzori* care transformă stimulii în *percepții* și *actuatori* care convertesc comenzile în acțiuni. Stimulii se referă la lumină, sunet, cuvinte tastate pe o tastatură, mișcări ale mouse-ului, informații dintr-o pagină web sau bază de date, etc. Senzorii pot fi de: proximitate, atingere, camere video, infraroșii, sonici, microfoane, cititoare XML folosite pentru extragerea informațiilor din paginile web.

AI are în vedere raționamentul practic: raționamentul pentru a face ceva. Un agent reprezintă o combinație a percepției, raționamentului și acționării. Un agent acționează într-un mediu. Mediul unui agent poate include și alți agenți. Un agent ar putea fi:

* un robot, o cuplare a unui motor computational cu senzori și actuatori fizici;
* un sistem expert, în care avem un calculator și un om care furnizează informații perceptuale și îndeplinește sarcina;
* un program care să acționeze într-un mediu pur computațional - software.

Pentru a putea vorbi de agenți în contextul AI, suntem nevoiți să:

* avem o cunoașterea prealabilă a agentului și a mediului în care se desfășoară acțiunile;
* știm istoricul interacțiunii cu mediul, care constă în observații ale mediului actual și experiențele anterioare ale acțiunilor și observațiilor anterioare sau alte date de la care poate fi învățat;
* cunoaștem obiectivele ce trebuie atinse;
* se dezvolte abilitățile (acțiunile primitive) pe care agentul să le poată îndeplini.

Din punct de vedere determinist dacă avem doi agenți care dețin aceleași cunoștințe anterioare, istoric, abilități și obiective ar trebui să facă același lucru. Schimbarea oricăror dintre acestea ar putea genera la unul dintre cei doi agenți acțiuni diferite. Fiecare agent este caracterizat de:

* o anumită stare internă care poate codifica percepțiile despre mediu cât și despre el însuși;
* anumite obiective, proprii, de realizat;
* modalități de a acționa asupra mediului înconjurător pentru a atinge obiectivele;
* instrumente (actuatori și efectori) necesari pentru a modifica modul de acțiune în urma efectuării unui raționament bazat pe percepție și învățare.

Această viziune, cuprinzătoare, asupra agenților inteligenți este una variabilă în funcție de complexitatea fenomenelor fizice despre a căror percepție sunt responsabili senzorii, iar răspunsul (acțiunea) fiind corespunzător cerințelor stabilite de către operatori.

<https://artint.info/html/ArtInt_36.html>

Agenții inteligenți sunt înzestrați cu „*rațiune*” pentru a putea fi utilizați să efectueze lucruri utile. Determinarea elementelor de rațional este dată de ceea ce considerăm a fi o „*acțiune corectă*”. Implementarea *rațiunii* conferă agentului un anumit grad de decizie. Fiecare agent are o măsură privind criteriile de performanță. De aceea măsurarea performanței se va stabili conform unor proceduri ale unor autorități.

Standardele sunt definite în funcție de o multitudine de aspecte: mediu natural, mediu social, intervenții de urmărir, de decizie, economic etc. Toate fiind cuantificate în unități de timp, ținând cont de efectul urmărit și cauzele care pot conduce la diferite strategii.

O altă problemă pe care agenții inteligenți trebuie să o rezolve, o reprezintă distincția dintre „*raționalitate*” și „*omnisciență[[13]](#footnote-13)*”. Efectuarea unei acțiuni este rațională, dar lipsa unor elemente de decriptare a elementelor mediului de lucru, conduce la ideea că agentul respectiv nu poate „*să știe tot*”.

Definirea noțiunii de „*corect*” pentru algoritmul de funcționare al unui agent este similară cu noțiunile de abstractizare ce se regăsesc în: „*jocul cu bile de sticlă[[14]](#footnote-14)*”, „*GO[[15]](#footnote-15)*” sau „*Labirint[[16]](#footnote-16)*” și depinde de patru elemente [1]:

1. gradul de succes – măsura performanței;
2. percepția completă de la începutul unei acțiuni, până la momentul finalizării acțiunii;
3. ce știe despre mediul înconjurător;
4. ce acțiuni poate efectua.

Agentul rațional ideal trebuie să efectueze fiecare acțiune prin maximizarea performanțelor sale, pe baza datelor furnizate de sistemul de percepție. Aparent el respectă „*rațiunea*” cu care este înzestrat, dar el nu știe să se orienteze dual (stânga/dreapta, sus/jos). Raționale prima etapă este de evaluarea riscurilor - este de fapt un alt element de construcție al unui *agent inteligent*. Apoi trebuie să „*caute*” pe baza algoritmului de maximizare a performanței. Toate se subordonează „*analizei*” care are implementat un contor de timp.

În [11] se prezintă formatul ideal al unui agent inteligent:

1. biologic – asumarea elementelor din mediu;
2. comunicare;
3. arhitectură, planificarea task-urilor și control;
4. localizare, mapare[[17]](#footnote-17) și explorare;
5. transport și manipulare obiecte;
6. coordonarea mișcării;
7. roboți reconfigurabili;
8. învățare – deep learning, machine learning;
9. probleme distribuite adiționale pentru sisteme de roboți mobili autonomi.

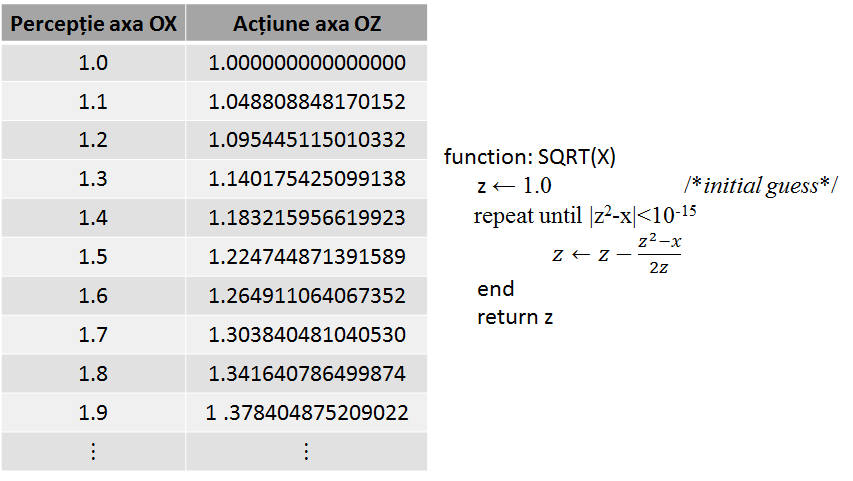


Fig. 1‑11 Maparea ideală și rezolvarea ei prin aplicarea metodei celor mai mici pătrate.

Inițiatorul mediului de testare al mediului Inteligenței Artificiale, Alan Turing[[18]](#footnote-18), a considerat că agenții reali și artificiali sunt identici, chiar dacă în lucrare consideră că agenții artificiali trbuie să dețină un algoritm (H&S) care să îi permită imitarea comportamentului uman.

Datele interne din structura unui agent inteligent se actualizează pe măsură ce informațiile care provin din sistemul de percepție ajund să fie interpretate de elementele cognitive ale agentului.

Agenții primiesc informația în mod secvențial, respectiv, informație după informație, astfel încât la intrare există adusă doar o singură percepție. Care este secvența pe care o va primi agentul ține de proiectant.

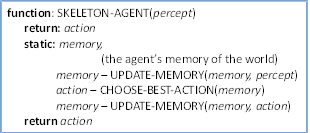


Fig. 1‑12 Structura unui agent inteligent.

În continuare, măsurarea performanței din Fig. 1‑13 se face aplicând din exterior un mod de comparare al unui comportament. Se scriu astfel datele într-un tabel de căutare (Fig. 1‑24) , tabel care funcționează prin menținerea, în memorie, a întregii secvențe de percepție cu ajutorul unui indice, care conține acțiunea corespunzătoare pentru toate secvențele posibile provenite din sistemul de precepție. Tabelul descrie de fapt valorile de risc:

1. numărul de intrări necesare pentru jocul de șah - – care pot fi asociate unei competiții normale, iar dacă se ia în calcul complexitatea jocului (game-tree complexity), din punct de vedere combinatoric, numărul estimat este în jur de (Shannon number[[19]](#footnote-19));
2. timpul pentru construirea tablei de joc;
3. faptul că agentul nu are autonomie deloc, deoarece calculul celor mai bune acțiuni este în întregime încorporat, în cazul în care mediul va suferi schimbări, într-un mod neașteptat, agentul va fi pus în dificultate și nu va putea învinge un competitor uman;
4. agentul dispune de un mecanism tip machine-learning, astfel încât, să aibă o anumită autonomie, însă timpul de învățare al tuturor mișcărilor posibile ar fi prohibit.

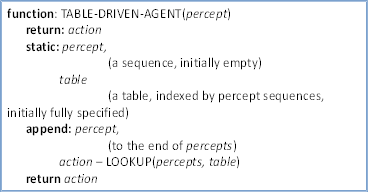


Fig. 1‑13 Tabel de căutare predifinit pentru un anumit agent.

Pentru a putea rezolva problema de mai înainte s-a apelat la noțiunea de „*Structură de Control Ierarhică – Hierarchical Control[[20]](#footnote-20)*”. Agentul este împărțit în senzori și un sistem de percepție complex, care trimite informațiile către un controler din care vor pleca informațiile pentru sistemul de acționare. Această structură este deficitară, întrucât este foarte lentă. De exemplu în cazul unui robot care trebuie să parcurgă un labirint ar trebui să dureze un timp foarte mare.

De asemenea capacitatea de a descrie mediul înconjurător este limitată.

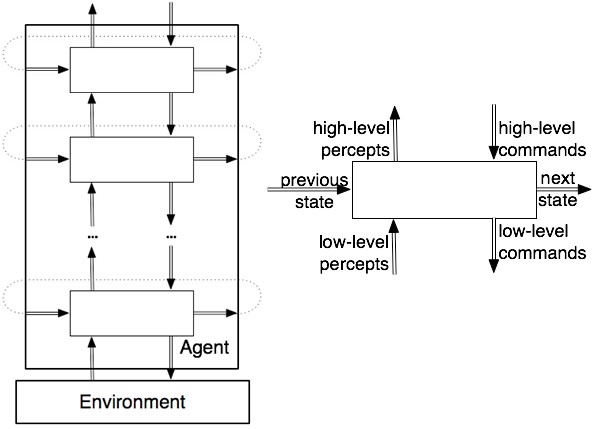


Fig. 1‑14 Structura idealizată a unui agent cu control ierarhic.

Arhitectura alternativă este o structură ierarhică de controlere (Fig. 1‑14), fiecare layer (strat) vede layerele de mai jos, ca pe un corp virtual care provine din sistemeul de percepție și care apoi trimite informațiile către sistemul de acționare. Straturile de nivel inferior sunt capabile să ruleze mult mai rapid și oferă o vedere mai simplă către straturile superioare, concomitent cu ascunderea informațiilor neesențiale.

Există trei tipuri de intrări pentru fiecare strat, la fiecare dată:

* caracteristicile stării de certitudine, denumite valori sau caracteristici anterioare;
* caracteristicile reprezentând precepțiile din stratul ierarhic de mai jos;
* caracteristicile reprezentând comenzile din stratul ierarhic de mai sus.

Există trei tipuri de ieșiri din fiecare strat, la fiecare dată:

* precepțiile de nivel superior pentru stratul de mai sus;
* comenzile de nivel inferior pentru stratul de dedesubt;
* următoarele valori pentru caracteristicile stării de certitudine.

Pentru a pune în aplicare un controler, fiecare intrare de la un strat trebuie să obțină valoarea sa de undeva. Fiecare comandă de intrare sau de precepție trebuie să fie conectată la o ieșire a unui alt strat. Analiza informațiilor în layerele superioare utilizează metode discrete și robuste, pe când în layerle inferioare analiza este de continuu și robust. Controlerul care are implementate ambele sisteme de analiză discret/robust și continuu/robust se numesc controelere hibride.

Raționamentele cantitaive folosește un aparat matematic ce utilizează analiza numerică prin metode integro-diferențiale. Raționamentele calitative folosec în plus *logica*. Raționamentul calitativ determină legile cantitative ce trebuie aplicate. Valorile discrete pot lua forme de reprezentare diverse: obiective, comenzi cu termeni fuzzy, motivații calitative

În cazul unui robot care se deplasează obiectivele, comenzile și motivațiile sunt date de spațiul, viteza, accelerația și obstacolele ce pot apare în traseul robotului. Bracarea roților poate fi gestionată ca fiind o acțiune instantanee, dar care trebuie ajustată permanent în funcție de timp, destinație, hartă și traiectorie. Senzorii de poziție oferă coordonatele (prin suprapunere GPS și cel triaxial), dacă este echipat cu ambele sisteme de poziționare. Dacă nu, la întâlnirea unui obstacol va efectua un viral la stânga și apoi la dreapta (spațiu/timp). Altfel spus harta este o secvență de locații citite într-o anumită ordine prestabilită (Fig. 1‑15).

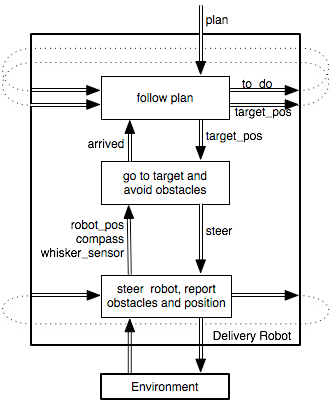


Fig. 1‑15 Descompunerea structura idealizate a unui robot în agenți supuși unui control ierarhic.

În interiorul unui layer avem caracteristici ce sunt funcții ale altor carcateristici (cinematică, dinamică, curent consumat, absorbit, sisteme termodinamice, hidraulice etc.) și ale intrărilor provenite de la alte layere. Dacă informațiile ajung într-un centru de prelucrare (Fig. 1‑16) pentru a se lua o decizie privind ocolirea unor obstacole, atunci elementele efectoare (motoarele electrice) vor primi comenzile de modificare a turației astfel încât dinamica robotului să fie modificată.

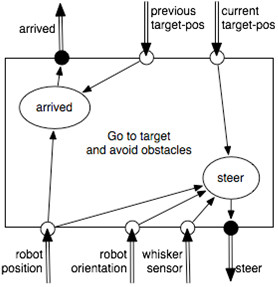


Fig. 1‑16 Layerul de mijloc în cadrul structurii ierarhice.

Acest layer nu menține starea de certitudine internă, astfel încât, funcția de tranziție să fie o mulțime nevidă. Funcția de comandă va specifica direcția robotului în funcție de intrările sale. Faptul că robotul atinge punctul stabilit, atribuie o valoare (care este dată de funcția de mișcare) ținând cont de valoarea anterioară și de o valoare prag predefinită :

*arrived ←distance(previous\_target\_pos,robot\_pos)<threshold*

*if whisker\_sensor=on*

*then steer←left*

*else if straight\_ahead(robot\_pos,robot\_dir,current\_target\_pos)*

*then steer←straight*

*else if left\_of(robot\_position,robot\_dir,current\_target\_pos)*

*then steer←left*

*else steer←right*

*end if*

*if arrived and not empty(to\_do)*

*then*

*target\_pos' ←coordinates(head(to\_do))*

*to\_do'←tail(to\_do)*

*end if*

dacă:

* *arrived ←* înseamnă atribuire, distanța este distanța euclidiană, iar pragul este o distanță în unitățile corespunzătoare;
* *straight\_ahead (robot\_pos, robot\_dir, current\_target\_pos)* este adevărat atunci când robotul este la *robot\_pos*;
* *robot\_dir* reprezintă direcția când poziția este țintă actuală;
* *current\_target\_pos* este stabilită ca direcție în sensul longitudinal al robotului (valoare prag);
* *left\_of* este funcția de testare în cazul în care obiectivul este spre stânga robotului.

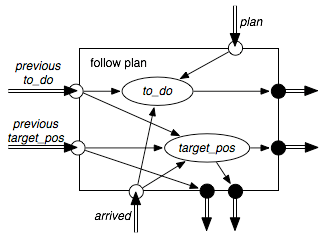


Fig. 1‑17 Layerul de deasupra în cadrul structurii ierarhice.

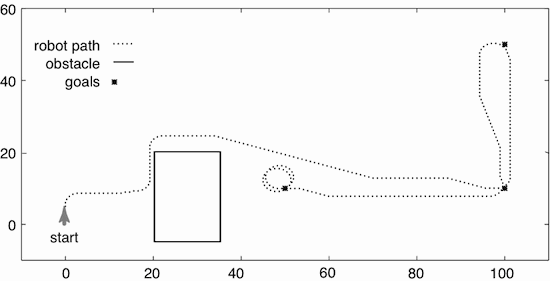


Fig. 1‑18 Simulare a deplasării robotului.

BIBLIOGRAFIE – cap. 1

|  |  |
| --- | --- |
| [1] | Stuart J. Russell și Peter Norvig, Artificial Intelligence a Modern Approach, New Jersey 07632: Prentice Hall, Englewood Cliffs, 1995. |
| [2] | Nicolas Andreff și Brahim Tamadazte, „Laser Steering using Virtual Trifocal Visual Servoing,” FEMTO-ST Institute, UMR CNRS 6174 - UFC / ENSMM / UTBM., 24 rue Alain Savary, 25000 Besan¸con, France. |
| [3] | C. Copoț, „Tehnici de control pentru sistemele servoing vizuale,” Politehnium, Iași, 2011. |
| [4] | T. Schmits și A. Visser, „An Omnidirectional Camera Simulation for the USARSim World.,” în *RoboCup 2008: Robot Soccer World Cup XII [papers from the 12th annual RoboCup International Symposium, July 15-18, 2008]*, Suzhou, China, July 15-18, 2008. |
| [5] | M. Brown și D. G. Lowe, „Recognising Panoramas,” în *Proceedings Ninth IEEE International Conference on Computer Vision*, Nice, France, october, 2003. |
| [6] | L. Tony și B. Lars, „Scale-Space Theory in Computer Vision,” Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994. |
| [7] | P. Lynne E., „Distributed Intelligence: Overview of the Field and its Application in Multi-Robot Systems,” în *AAAI Fall Symposium on Regarding the Intelligence in Distributed Intelligent Systems, 2007.*, Alicante, march 2008. |
| [8] | D. A. Schoenwald, „AUVs: In Space, Air, Water, and on the Ground,” în *Autonomous Unnmed Vehicles*, Albuquerque, NM 87185, U.S.A., P.O. Box 5800, MS 1004, , December 2000. |
| [9] | I. Nuță, „Contribuţii la dezvoltarea și perfecționarea tehnicii de intervenție în situații de urgență.,” Military Technical Academy, București, 2015. |
| [10] | D. L. Poole și A. K. Mackworth, Artificial Intelligence: Foundations of Computational Agents, 2nd Edition, 2 ed., Cambridge: Cambridge University Press, 2017. |
| [11] | L. E. Parker, „Curent State of the Art in Distributed Autonomous Mobile Robotics,” Springer-Verlag, Tokyo, 2000. |
| [12] | G. J.C., Dimensional Analysis, Liverpool: Springer-Verlag London, 2011. |
| [13] | B. I. d. P. e. Mesures, „The International System of Units (SI),” în *The International System of Units (SI)*, 8th edition ed., O. I. d. l. C. d. Metre, Ed., Paris, 1, Boulevard Ney, 75018: STEDI MEDIA, 2006, p. 88. |
| [14] | F. I., „http://www.scribd.com/doc/59227370/27/Metodele-analizei-dimensionale,” Editura ALMA MATER, Bacău, 2007. [Interactiv]. [Accesat 23 10 2016]. |

ANEXE – cap. 1

**No table of figures entries found.**

FIGURI – cap. 1

[Fig. 1‑1 Rețea neuronală specifică unei ființe umane. 11](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592626)

[Fig. 1‑2 Reprezentarea experimentului Turing 12](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592627)

[Fig. 1‑3 Google autonomous vehicle 14](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592628)

[Fig. 1‑4 Domenii de bază ale IA 15](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592629)

[Fig. 1‑5 Arnold Denker în competiție cu robotul HITECH 17](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592630)

[Fig. 1‑6 Controlerul robotului: un PLC master și două PLC slave 18](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592631)

[Fig. 1‑7 Schema funcțională a unei camere video omnidirecționale 19](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592632)

[Fig. 1‑8 Principiul functional al MAISART. 20](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592633)

[Fig. 1‑9 Labirintul circular folosit pentru demonstrarea tehnologiei MASIART. 20](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592634)

[Fig. 1‑10 Interacțiunea între un agent și mediul real 22](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592635)

[Fig. 1‑11 Maparea ideală și rezolvarea ei prin aplicarea metodei celor mai mici pătrate. 24](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592636)

[Fig. 1‑12 Structura unui agent inteligent. 24](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592637)

[Fig. 1‑13 Tabel de căutare predifinit pentru un anumit agent. 25](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592638)

[Fig. 1‑14 Structura idealizată a unui agent cu control ierarhic. 25](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592639)

[Fig. 1‑15 Descompunerea structura idealizate a unui robot în agenți supuși unui control ierarhic. 26](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592640)

[Fig. 1‑16 Layerul de mijloc în cadrul structurii ierarhice. 27](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592641)

[Fig. 1‑17 Layerul de deasupra în cadrul structurii ierarhice. 28](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592642)

[Fig. 1‑18 Simulare a deplasării robotului. 28](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592643)

[Fig. 1‑10 Schema funcțională a unui system servoing 29](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592644)

[Fig. 1‑11 Principiul funcției de autocorelație 30](file:///F:\DICIPLINE%20INGINERIE\Discipline%20RO\010_Inteligenta%20Artificiala\IA_1_Suport%20curs\T_01_Introducere.%20Agenti%20inteligenti\IA_cap_01.docx#_Toc528592645)

TABELE – cap. 1

**No table of figures entries found.**

NOTAȚII ȘI SIMBOLURI – cap. 1

Mărimile , simbolurile și unitățile de măsură au fost reprezentate respectând [17], [18], [19].

|  |  |  |  |
| --- | --- | --- | --- |
| Mărimea | Simbol | Unitatea de măsură în <SI> | Domeniu |
| Acceleraţie | a, g | m/s2 | mecanică |
| Amplitudinea unui semnal | A | V, A | semnale electrice |
| Căldură specifică | c | m2/(s2·K) | transfer termic |
| Coeficient de conductibilitate a temperaturii prin convecţie sau radiaţie | α | W/(m2·K) = kg/(s3·K) | transfer termic |
| Coeficient de conductivitate termică | λ | W/(m·K) = (kg·m)/(s3·K) | transfer termic |
| Coeficient termic de dilatare volumetrică | β | 1/T | transfer termic |
| Coficientul de frecare termic pe unitatea de suprafaţă | τ | N/m2 = kg/(m·s2) | transfer termic |
| Conductibilitatea termică specifică sau coeficient de conductibilitate total | αT | W/(m2·K) = kg/(s3·K) | transfer termic |
| Debitul masic |  | kg/s | transfer termic |
| Densitate | ρ | kg/m3 | termodinamică |
| Energia internă | U | J = kg·m2/s2 | transfer termic |
| Energie termică | Q | J = kg·m2/s2 | transfer termic |
| Entalpie | H | J = kg·m2/s2 | transfer termic |
| Faza inițială | φ0 | rad | semnale electrice |
| Flux termic | q | W = kg·m2/s3 | transfer termic |
| Forţă | F | N = kg·m/s2 | mecanică |
| Frecvența semnalului | f | Hz | semnale electrice |
| Intensitatea curentului electric | I | A | semnale electrice |
| Lungime | l, x | m | mecanică |
| Masă | m | kg | mecanică |
| Perioada semnalului | T | s | semnale electrice |
| Polul unui SA | p | - | sisteme automate |
| Presiune | p | N/m2 = kg/m·s2 | termodinamică |
| Pulsația unui semnal | ω0 | rad/s | semnale electrice |
| Rezistenţa termică | R | K/W = (K·s2)/(kg·m2) | termodinamică |
| Temperatură | T | K | termodinamică |
| Tensiune electrică | U | V | semnale electrice |
| Tensiune specifică | σ | N/m2 = kg/(m·s2) | hidraulică |
| Tensiune superficială | σS | N/m = kg/s2 | hidraulică |
| Timp | t | s |  |
| Vâscozitatea cinematică | νc =μd/ρ | m2/s | hidraulică |
| Vâscozitatea dinamică | μd | kg/(m·s) | hidraulică |
| Viteză | v | m/s | mecanică |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

ABREVIERI – cap. 1

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
| **ABS** | Anti-lock Braking System - Sistem antiblocare a roților |
| **ADC** | Analog to Digital Conversion A → D |
| **AO (op-AMP)** | Amplificator Operațional |
| **BC** | Bloc de Calcul |
| **BI** | Bloc de Identificare |
| **BMP** | Bloc de Modificare a Parametrilor |
| **CM** | Common-mode |
| **CP** | Criteriu de Perfromanță |
| **DAC** | Digital to Analog Converter D → A |
| **DOF** | Degrees of Freedom |
| **EC** | Element de Comparație |
| **EE** | Element de Execuție |
| **ELCM** | Element (sistem) Liniar și Continuu Monovariabil |
| **EMC** | Compatibilitate electromagnetică |
| **EMI** | Interferență electromagnetică |
| **EPS** | Electronic Stability Control – controlul electronic al stabilității |
| **FT** | Funcție de Transfer |
| **H** | Funcție de Transfer |
| **IC** | Integrated Circuit |
| **IA** | Inteligență Artificială |
| **LAN** | Local Area network |
| **LDO** | Low drop out |
| **LID** | Automatic Language Identification |
| **MIMO** | Multi-Input Multi-Output sisteme multivariabile |
| ***p*** | Polul sistemului automat |
| **P** | Regulator automat Proporțional |
| **PCB** | Plăci cu circuite integrate |
| **PD** | Regulator automat Proporțional-Derivativ |
| **PI** | Regulator automat Proporțional-Integral |
| **PID** | Regulator automat Proporțional-Integral-Derivativ |
| **PSF** | Punct Static de Funcționare |
| **RA** | Regulator Automat |
| **RAA** | Regulator Automat Adaptiv |
| **RF** | Radio Frecvență |
| **SA** | Sistem Automat |
| **SAL** | Sistem Automat Liniar |
| **SALC** | Sistem Automat Liniar și Continuu |
| **SAN** | Sistem Automat Neliniar |
| **SFint** | Spațiul Funcțiilor de Intrare |
| **SFout** | Spațiul Funcțiilor de Ieșire |
| **SISO** | Single-Input Single-Output sisteme monovariabile |
| **SRA** | Sistem Reglare Automată |
| **SFI** | Spațiul Funcțiilor de Intrare |
| ***tt*** | *Durata regimului tranzitoriu* |
| **u** | Vectorul Mărimilor de Intrare |
| **U** | Spațiul Funcțiilor de Intrare |
| **UMVS** | Unmanned Maritime Vehicle Systems |
| **VL** | Variabilă Longitudinală - |
| **VT** | Variabilă Transversală - |
| **x** | Vectorul de Stare |
| **X** | Spațiul Stărilor |
| **y** | Vectorul Mărimilor de Ieșire |
| **Y** | Spațiul Funcțiilor de Ieșire |
| **** | Laplacian |
|  |  |
|  |  |
|  |  |

1. <http://statweb.stanford.edu/~jtaylo/courses/stats202/svms.html> [↑](#footnote-ref-1)
2. <http://www.psych.utoronto.ca/users/reingold/courses/ai/turing.html> [↑](#footnote-ref-2)
3. <https://en.wikipedia.org/wiki/Dartmouth_Conferences> ;

   <http://www-formal.stanford.edu/jmc/history/dartmouth/dartmouth.html> [↑](#footnote-ref-3)
4. <http://www.egemin-automation.com/en/automation/material-handling-automation_ha-solutions/agv-systems> [↑](#footnote-ref-4)
5. <http://www-formal.stanford.edu/jmc/whatisai/node1.html> [↑](#footnote-ref-5)
6. <http://www.aaai.org/ojs/index.php/aimagazine/article/view/739/657> [↑](#footnote-ref-6)
7. <http://www.ace.tuiasi.ro/users/103/2011-Copot%20Cosmin.pdf> [↑](#footnote-ref-7)
8. <http://www.mitsubishielectric.com/news/2018/pdf/0214-b.pdf> [↑](#footnote-ref-8)
9. Mitsubishi Electric's AI creates the State-of-the-ART in technology [↑](#footnote-ref-9)
10. <https://en.wikipedia.org/wiki/Moravec's_paradox> [↑](#footnote-ref-10)
11. distanța dintre planul de mișcare a obiectelor și cameră este constant [↑](#footnote-ref-11)
12. <https://en.wikipedia.org/wiki/Stigmergy> [↑](#footnote-ref-12)
13. <https://dexonline.ro/intrare/omniscien%C8%9B%C4%83/39032> - calitatea de a ști tot [↑](#footnote-ref-13)
14. <https://merlinstore.ro/jucarii/257-joc-cu-bile-de-sticla-standard-solitaire.html> [↑](#footnote-ref-14)
15. <http://www.netinform.ro/blog/2016/03/10/inteligenta-artificiala-il-invinge-pe-campionul-jocului-go/> [↑](#footnote-ref-15)
16. <http://www.scritub.com/sociologie/psihologie/Strategii-de-rezolvare-a-probl2332213815.php> [↑](#footnote-ref-16)
17. <https://dexonline.ro/intrare/mapare/216699> - cartografiere [↑](#footnote-ref-17)
18. <http://www.turing.org.uk/scrapbook/test.html> [↑](#footnote-ref-18)
19. <https://en.wikipedia.org/wiki/Claude_Shannon> [↑](#footnote-ref-19)
20. <http://artint.info/html/ArtInt_37.html> [↑](#footnote-ref-20)