

SISTEM ROBOT EDUCAȚIONAL DE NAVIGARE

Mihaela M. Oprea

Nicolae Aldea

Universitatea Petrol - Gaze din Ploiești

Rezumat: Testarea eficienței diferiților algoritmi de navigare, de învățare automată, aplicați în domeniul roboticii mobile implică primul pas în utilizarea unor sisteme robot de tip educațional, urmând ca într-o etapă ulterioară algoritmi să fie testați pe sisteme robot comerciale. Lucrarea prezintă un sistem robot educațional, ROBO, destinat navigării într-un mediu interior necunoscut, care este utilizat cu precădere pentru testarea algoritmilor specifici inteligenței artificiale.

Cuvinte cheie: navigare, robot mobil, algoritmi de navigare.

1. Introducere

Navigarea roboților mobili, într-un mediu dinamic și incert, reprezintă o direcție prioritară de cercetare în domeniul roboticii [1]. Pentru a naviga în siguranță într-un mediu interior necunoscut, robotul mobil trebuie să urmărească o anumită cale, numită și traiectorie și în același timp să rezolve situațiile noi care pot apărea. De exemplu, robotul trebuie să fie capabil să evite obstacolele din mediu, să rezolve erorile senzorilor (de exemplu, ale senzorului sonar sau erorile de odometrie). În timpul urmăririi căii, robotul deliberază și utilizează o politică de mapare a percepției mediului în acțiuni astfel încât să aibă posibilitatea de a reacționa la schimbările din mediul său de lucru. Planificarea traiectoriei reprezintă o etapă importantă în navigarea robotului și determină modalitatea în care robotul se poate mișca eficient, între două locații date. Incertitudinea indusă de senzori complică planificarea traiectoriei întrucât robotul poate să nu fie capabil să urmărească cu exactitate traiectoria determinată. Un mediu real de lucru este inaccesibil (de exemplu, datorită imperfecțiunii senzorilor), nedeterminist (de exemplu, bateriile se consumă/descarcă, roțile alunecă), nonepisodic (de exemplu, efectele unei acțiuni se schimbă în timp), dinamic și continuu. Reprezentarea mediului de lucru se poate realiza fie geometric, fie topologic. Reprezentarea geometrică este adecvată mediilor interioare, în timp ce reprezentarea topologică este adecvată mediilor exterioare de lucru. Testarea sistemelor robot mobile se bazează pe noțiunea de autonomie, adică abilitatea unui robot de a îndeplini în siguranță misiunea primită (fără intervenția omului). Autonomia robotului este implementată cu ajutorul tehnicilor de inteligență artificială [2], [3].

În ultimul deceniu, utilizarea roboților mobili s-a extins din domeniul academic și de cercetare în cel comercial și industrial. Dintre clasele de roboți mobili utilizați pe scară largă la ora actuală menționăm: Nomad [4], Khepera [5] și Pioneer [6]. În plus, au apărut și o serie de roboți jucărie, cum sunt Palm Pilot Robot [7] și I-Cybie [8]. Acești roboți realizează diferite task-uri în aplicații din cele mai diverse. Robotul Nomad utilizează rețelele neuronale artificiale atât în percepția mediului, cât și în navigare, simulând reacțiile creierului uman și încercând o adaptare continuă la mediu. Khepera II este un robot educațional de dimensiuni mai mici, modularizat la nivel hardware și software, cu o bibliotecă eficientă pentru aplicații on-board, destinat experimentării algoritmilor de planificare a traiectoriei, de evitare a obstacolelor etc. Pioneer 3-DX este dotat cu un calculator de bord, care urmărește în timp real deplasarea robotului și este destinat aplicațiilor de cartografiere, teleoperare, localizare, recunoaștere, vedere artificială etc. Palm Pilot Robot a fost dezvoltat la Universitatea Carnegie Mellon și este un robot autonom tip jucărie de dimensiuni mici, care permite testarea diferiților algoritmi de navigare existenți. I-Cybie este un robot inteligent interactiv, practic un câine cibernetic, care simulează un câine real și care are o serie de abilități: recunoaște vocea stăpânului și răspunde la câteva comenzi vocale ale acestuia, poate naviga în siguranță, evitând obstacolele.

O soluție economică, implementabilă în mediul academic și de cercetare, o reprezintă dezvoltarea unor sisteme robot de tip educațional [9], care să permită utilizarea lor ca un pas intermediar între simulatoare și sistemele reale [10], [11]. La ora actuală, există o astfel de tendință, de a construi sisteme robot educaționale din componente oferite pe piață de diferite firme specializate în comerțul cu jucării (de exemplu, Lego). Lucrarea prezintă un sistem robot educațional de navigare, ROBO, care a fost dezvoltat în cadrul Catedrei de Informatică a Universității Petrol - Gaze din Ploiești [12] cu scopul de a fi utilizat la experimentarea diferiților algoritmi de navigare și de învățare automată, prezentați la cursurile de inteligență artificială și sisteme multiagent. Soluția adoptată a fost una simplă și economică atât din punct de vedere al resurselor fizice utilizate (robot, senzori), cât și software (C++ Builder).

Lucrarea cuprinde patru secțiuni. După secțiunea introductivă, în secțiunea a 2-a este descrisă problema navigării unui robot mobil și sunt prezentați, pe scurt, principalii algoritmi de navigare.

Secțiunea a 3-a prezintă sistemul robot autonom ROBO, destinat navigării într-un mediu interior (de exemplu, etajul unei clădiri). Ultima secțiune concluzionează lucrarea.

2. Navigarea roboților mobili

Navigarea constă în localizarea robotului (stabilirea poziției în cadrul unui sistem de referință), planificarea traiectoriei (stabilirea drumului ce va fi parcurs de robot astfel încât să fie rezolvată și problema ocolirii obstacolelor și problema urmării peretelui) și crearea / interpretarea unei hărți a mediului de navigare. Figura 1 prezintă schematic planificarea traiectoriei. Astfel, problema navigării constă în stabilirea unei traiectorii între poziția curentă (S_i) și poziția destinație (S_f) la care trebuie să ajungă robotul, astfel încât să fie evitate obstacolele statice și dinamice din mediul de lucru [13]. Harta este o reprezentare internă a mediului de navigare. Robotul își va monitoriza continuu senzorii și va verifica poziția sa în raport cu referința.

Sintetizând, problema găsirii unei traiectorii este echivalentă cu o problemă de optimizare care are două restricții:

1. evită coliziunile cu obstacolele statice și dinamice din mediul de lucru,
2. minimizează lungimea drumului ce va fi parcurs între S_i și S_f .

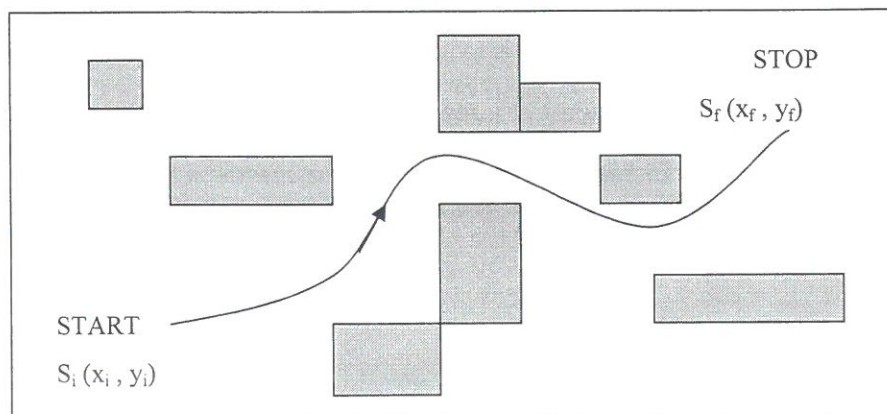


Figura 1. Planificarea traiectoriei

Planificarea traiectoriei se poate realiza cu ajutorul metodelor locale, respectiv globale. Metodele locale se bazează pe informațiile primite de la senzori, nu utilizează cunoștințe a priori, nu găsesc întotdeauna drumul optim, pot genera blocaje în forme de tip U. Metodele globale presupun existența unui model complet al mediului de lucru și, în consecință, nu sunt potrivite mediilor necunoscute și dinamice. Avantajul major al metodelor locale este timpul mic de răspuns, ele fiind adecvate aplicațiilor ce necesită răspuns în timp real. Metodele locale sunt aplicate roboților mobili, care navighează exclusiv pe baza informațiilor primite de la senzori.

Principalele tipuri de senzori care permit roboților mobili să realizeze o navigare autonomă sunt senzorii de proximitate (cu infraroșu, cu ultrasunete, cu laser), senzorii tactili (de coliziune), senzorii de verticalitate, senzorii de orientare (GPS, camerele video) [14]. Navigarea unui robot mobil trebuie să ia în calcul o serie de erori posibile, de înaintare, de poziționare, de direcție. De exemplu, în cazul navigării în linie dreaptă, paralel cu peretele, eroarea de înaintare este aceea în care robotul este prea aproape de peretele din față, în acest caz soluția fiind dată de senzorii cu infraroșu montați față-spate (poziționați perpendicular față de perete). Eroarea de poziționare poate apărea după o curbă, în timp ce eroarea de direcție poate conduce la o navigare a robotului către perete, și nu paralel cu peretele, în cazul urmării peretelui. O soluție este oferită de o tabelă cu acțiuni de corecție, specifice fiecărei situații.

Din clasa algoritmilor de navigare, care se bazează pe metode de căutare pentru găsirea traiectoriei de navigare, fac parte algoritmii euristici: Best-First (BF) [15], A^* [16], D^* [17] și algoritmul Bellman [18].

Rezolvarea problemei navigării unui robot mobil se poate reduce la căutarea unei soluții (un drum între pozițiile de start și destinație) într-un spațiu de căutare (arbore, respectiv graf ȘI-SAU). Astfel, soluțiile date de algoritmii euristici pot fi preluate în navigarea roboților pentru planificarea traiectoriei.

Algoritmii euristici utilizează funcții euristice, $f(n)$, care permit găsirea rapidă a unei soluții, de obicei suboptimală, n fiind nodul curent (starea curentă, de exemplu, poziția curentă a robotului) în care se evaluează f .

Căutarea informată de tip Best-First este o strategie de căutare generală, în care nodurile de cost minim (în conformitate cu o anumită măsură) sunt primele expandate, adică selectate pentru generarea traiectoriei. Funcția de evaluare euristică $f(n)$ depinde de nodul curent n , de starea finală (destinația robotului), de informația acumulată de strategia de căutare până în acel nod și de alte cunoștințe specifice navigării roboților mobili.

În cazul algoritmului A*, funcția f are formă aditivă:

$$f(n)=g(n) + h(n)$$

Funcția g măsoară distanța parcursă de la poziția de start la poziția curentă (starea n), iar funcția h estimează distanța care trebuie parcursă de la poziția curentă la poziția destinație (starea finală). Nodurile care au valoarea minimă a funcției f sunt alese cu prioritate. Algoritmul A* poate găsi soluția optimă în anumite condiții impuse funcției euristice, care se referă la admisibilitate, monotonicitate și grad de informare. În cazul navigării într-un mediu dinamic, se poate utiliza o variantă a algoritmului A*, numită D*, introdusă de către Stentz în [17]. Algoritmul Bellman [18] are la bază programarea dinamică. Este utilizată o matrice Bellman de direcție, care va ghida procesul de selecție a pozițiilor următoare în care robotul se va deplasa în drumul său către poziția destinație.

3. Sistemul de navigare autonomă ROBO

Sistemul de navigare ROBO este alcătuit dintr-o mașină de teren în miniatură (figura 2), capabilă să se deplaseze pe suprafețe solide plane și să depășească obstacole ca pragul unei uși sau să urce o rampă puțin înclinată.

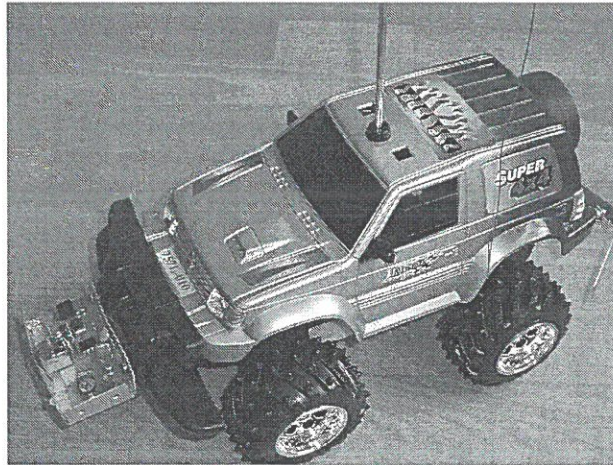


Figura 2. Robotul ROBO

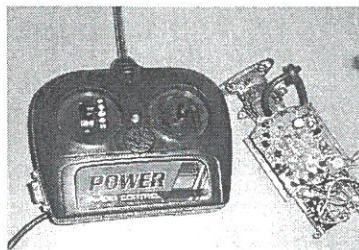


Figura 3. Partea fixă a sistemului robot ROBO

Dimensiunile robotului sunt: 35 cm lungime, 18 cm lățime, 60 cm înălțime, 8 cm diametrul roților și 1 Kg greutate. Robotul este dotat cu senzori de proximitate cu infraroșu, senzori de coliziune, două motoare

de curent continuu, comandate prin choppere (fiecare realizând deplasarea pe o direcție în ambele sensuri). Comunicarea cu calculatorul în ambele direcții se realizează cu ajutorul a două sisteme de transmisie de unde radio, care lucrează pe frecvențe diferite (27 MHz și 49 MHz). Sistemul utilizează 2 baterii de 9 V și 11 baterii de 1,5 V, în situația în care nu se dorește legarea robotului de nici un reper fix, sau se poate utiliza un transformator de 9 V curent continuu și 5 baterii de 1,5 V.

Din punct de vedere fizic, sistemul este alcătuit din două părți, partea mobilă și partea fixă. Partea mobilă a sistemului robot este alcătuită, la rândul ei, din partea de emisie a subsistemului de achiziție, partea de recepție a subsistemului de comandă și din partea de execuție reprezentată de cele două motoare de curent continuu. Partea fixă a sistemului robot este prezentată în figura 3. Aceasta este în legătură fizică cu calculatorul printr-un cablu conectat la portul de imprimantă și este alcătuită din partea de recepție pentru subsistemul de achiziție și din partea de emisie a subsistemului de comandă.

Controlul robotului permite următoarele moduri de lucru:

- *modul de navigare manuală* – în acest mod, robotul poate fi comandat de un operator uman cu ajutorul telecomenzii sau prin intermediul interfeței, prin apăsarea unor butoane;
- *modul de navigare hoinar* – în acest mod, robotul se plimbă aleatoriu în mediul de lucru, reacționând la obstacolele din mediu;
- *modul de navigare planificată* – în acest mod, robotului i se dă o poziție finală, în care trebuie să ajungă, iar ROBO ocolește obstacolele în încercarea de a găsi o traiectorie de navigare, dacă aceasta există.

Arhitectura sistemului robot (figura 4) este alcătuită din mașina de teren în miniatură și trei subsisteme: subsistemul de calcul, subsistemul de comandă și subsistemul senzorial și de achiziție.

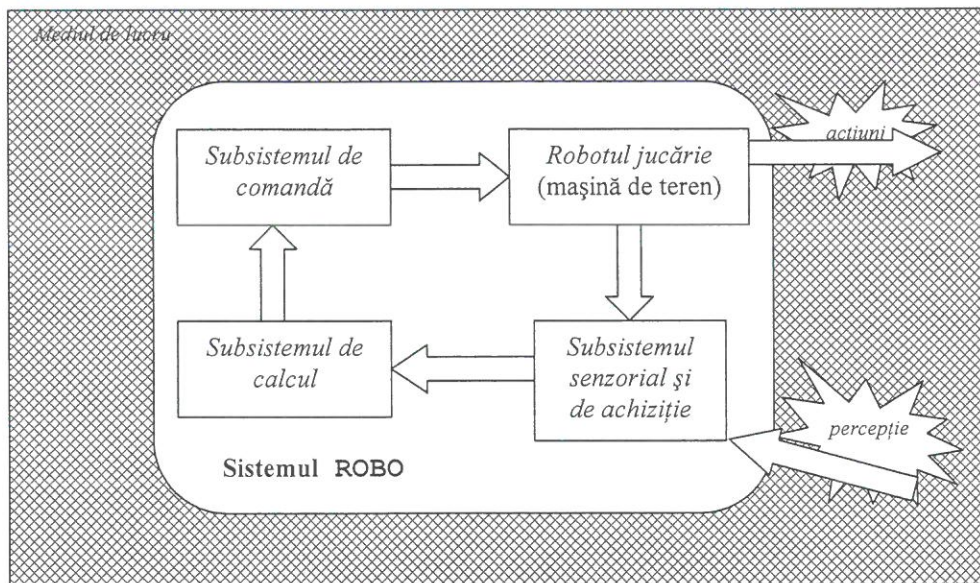


Figura 4. Arhitectura sistemului ROBO

Subsistemul de calcul este alcătuit din calculator și aplicația software creată în C++ Builder 5.0. Pentru comunicarea cu portul paralel s-a utilizat biblioteca "axdrv.h". Subsistemul de comandă realizează transmiterea unor comenzi de la calculator către robot și execuția lor printr-o conversie *curent – unde radio – curent – deplasare*. Subsistemul senzorial și de achiziție realizează detectarea obstacolelor din traiectoria robotului precum și achiziția acestor date de către calculator. Partea de achiziție este realizată prin intermediul unui sistem de emisie – recepție de unde radio pe 49 MHz. Sistemul de emisie a undelor radio este montat pe robot și acesta emite numai atunci când este comandat de către sistemul senzorial. Subsistemul senzorial este alcătuit din doi senzori de coliziune și un senzor de proximitate cu unde infraroșii. Cei doi senzori de coliziune sunt montați în spatele robotului pentru a detecta eventualele coliziuni stânga – spate sau dreapta – spate. Aceștia vor genera un semnal numai în cazul în care robotul va lovi ceva în încercarea de a da cu spatele. Senzorul cu infraroșu este montat în fața robotului fiind capabil să sesizeze un obstacol aflat la distanțe de până la 13 cm. Cu acest tip de senzor, pot apărea o serie

4. Concluzii

Sistemul ROBO a fost proiectat și implementat în scopuri pedagogice, fiind un sistem robot educațional de navigare. El este destinat experimentării diferiților algoritmi de navigare, de învățare automată (de tip învățare prin răsplată, rețele neuronale artificiale cu reacție înainte, algoritmi genetici) prezentați la cursurile de inteligență artificială și sisteme multiagent. Sistemul senzorial al robotului ROBO cuprinde un senzor de proximitate cu unde infraroșii și doi senzori de coliziune. Sistemul robot poate fi extins pentru diferite aplicații. De exemplu, se poate monta un sonar pentru a crește raza senzorială și, astfel, sistemul ROBO poate fi utilizat ca sistem de pază și alarmă, care patrulează la etajul unei instituții.

Bibliografie

1. **LATOMBE, J.-C.:** Robot Motion Planning, Kluwer Academic Publishers, 1995.
2. **MURPHY, R. R.:** Introduction to AI Robotics, MIT Press, 2000.
3. **PEARL, J.:** Probabilistic Reasoning in Intelligent Systems, Palo Alto, Morgan Kaufmann, 1988.
4. <http://www.21stcentury.co.uk/robotics/nomad.asp>
5. <http://www.k-team.com>
6. <http://www.activrobots.com/>
7. <http://www.21stcentury.co.uk/robotics/palmpilotrobot.asp>
8. <http://www.21stcentury.co.uk/robotics/icybie.asp>
9. **BORANGIU, Th., A. HOSSU:** Sisteme educaționale în robotică, Editura Tehnică, București, 1991.
10. **NEVES, M. C., E. OLIVEIRA:** A Control Architecture for an Autonomous Mobile Robot. În: Proceedings of Autonomous Agents 97, Marina Del Rey, California, USA, 1997, pp. 193-200.
11. **SIMMONS, R., R. GOODWIN, K. Z. HAIGH, S. KOENIG, J. O'SULLIVAN:** A Layered Architecture for Office Delivery Robots. În: Proceedings of Autonomous Agents 97, Marina Del Rey, California, USA, 1997, pp. 245-252.
12. **ALDEA, N.:** Sistem robot autonom de navigare, lucrare de licență, Universitatea Petrol-Gaze din Ploiești, Catedra de Informatică, 2004.
13. **OPREA, M.:** Path Planning in a Robot Navigation System. În: Proceedings of the 16th International Joint Conference on Artificial Intelligence IJCAI99 – workshop PLAN-2, Stockholm, Sweden, 1999, pp. 65-70.
14. **EVERETT, H. R.:** Sensors for Mobile Robots: Theory and Application, A. K. Peters Ltd, 1995.
15. **RUSSEL, S., P. NORVIG:** Artificial Intelligence – A Modern Approach, Prentice-Hall International, Inc., New Jersey, 1995.
16. **STOUT, B.:** The Basics of A* for Path Planning. În: Game Programming Gems, 2000, pp. 254-263.
17. **STENTZ, A.:** The Focussed D* Algorithm for Real-Time Replanning. În: Proceedings of the International Joint Conference on Artificial Intelligence – IJCAI95, 1995, pp. 1652-1659.
18. **BELLMAN, R.:** On a Routing Problem, Quarterly of Applied Mathematics, 16(1), 1958, pp. 87-90.