Model Learning for Robot Control

Rezumatul tezei de doctorat

Autor:
BOTOND ATTILA BÓCSI

Conducător științific:
PROF. DR. HORIA F. POP

DEPARTAMENTUL DE MATEMATICĂ ȘI INFORMATICĂ,
UNIVERSITATEA BABES-BOLYAI, CLUJ-NAPOCA, 2012

Cuprinsul rezumatului

Cuprinsul tezei de doctorat

Lista publicațiilor

- B. Bócsi, L. Csató, B. Schölkopf, and J. Peters. Indirect robot model learning for tracking control. *Robotics and Autonomous Systems (submitted on 10 September 2012)*, 2012a. Trimis.
- B. Bócsi, P. Hennig, L. Csató, and J. Peters. Learning tracking control with forward models. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 259–264, St. Paul, MN, USA, 2012b
- B. Bócsi, D. Nguyen-Tuong, L. Csató, B. Schoelkopf, and J. Peters. Learning inverse kinematics with structured prediction. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 698–703, San Francisco, USA, 2011b
- H. Jakab, B. A. Bócsi, and L. Csató. Non-parametric value function approximation in robotics. In H. F. Pop, editor, *MACS2010: The 8th Joint Conference on Mathematics and Computer Science*, volume Selected Papers, pages 235–248, Komarno, Slovakia, 2011. Györ:NOVADAT
- B. Bócsi and L. Csató. Reinforcement learning algorithms in robotics. In M. Frentiu, H. F. Pop, and S. Motogna, editors, *KEPT-2011:Knowledge Engineering Principles and Techniques International Conference, Selected Papers.*, pages 131–143. Presa Universitara Clujeana, 2011a
- B. Bócsi, L. Csató, and Jan Peters. Structured output Gaussian processes. Technical report, Babes-Bolyai University, 2011a. URL `http://www.cs.ubbcluj.ro/~bboti/pubs/sogp_2011.pdf`
- B. A. Bócsi and L. Csató. Reinforcement learning algorithms in robotics. *Studia Universitatis Babes-Bolyai Series Informatica*, LVI(2):61–67, 2011b
- B. A. Bócsi, H. Jakab, and L. Csató. Nonparametric methods in robotics. In *Proceedings of the 8th Joint Conference on Mathematics and Computer Science (Abstract)*, page 8, Komarno, Slovakia, 2010

Grant de cercetare relat de teză

- Grant **PN-II-RU-TE-2011-3-0278** al ministerul educației: Nonparametric methods in machine learning: application to robotic and data analysis.

1 Introducere

În ultimele decenii, cercetarea legată de robotică a devenit din ce în ce mai populară. Motivul pricipal este utilizarea roboților într-o mare varietate de domenii. De exemplu, nivelul ridicat de automatizare în industrie este realizat prin intermediul unor roboți. Utilizarea roboților are mai multe beneficii asupra forței de muncă umană, de exemplu, acesta poate fi mai rapid, mai ieftin, și mai precis. În ciuda caracteristicilor atrăgătoare, robotica automată are dezavantaje serioase. În general, roboți industriali funcționează bine numai în medii unde nimic neașteptat nu se poate întâmpla. De obicei, oamenii nu au aceste limitări. Ca o soluție, trebuie să dezvoltăm roboți care funcționează bine și în medii necontrolate.

Din cauza limitărilor de mai sus, aplicarea roboților în medii stohastice a devenit o cerință nouă. De exemplu, atunci când terenul de execuție este necunoscut, mediul de lucru al robotului trebui să fie considerat stohastic. De asemenea modelarea stohastică a mediului este necesară atunci când se face interacțiune cu oamenii, fiindcă modelarea deterministică a comportamentului uman este mult prea complexă. Scopul este de a lăsa robotii să rezolve o clasă mai largă de probleme, fără a preprograma toate scenariile posibile. Putem afirma că într-un mediu care se află într-o schimbare continuă roboții trebuie să posede un comportament inteligent și adaptiv.

Controlul adaptiv al roboților a dus la o abordare relativ nouă a roboticii [Thrun et al., 2005]. Abordări și algoritmi au fost împrumutate din teoria de control [Liberzon, 2012; Sontag, 1998], inteligență artificială [Russell and Norvig, 2003], și instruire automată [Bishop, 2006]. În domeniul de control adaptiv al roboților sunt formulate mai multe probleme, cum ar fi localizare, planificare, sau control al mișcării [Thrun et al., 2005], dar domeniul a crescut într-o rată așa de uriașă, încât prezentarea exhaustivă a domeniului este dincolo de limitele acestei lucrări.

Această teză se concentrează pe controlul robotului la nivel scăzut. Mai precis, ne propunem să definim algoritmi care produc comenzi de motor la nivel scăzut, fără a atinge obiective la nivel înalt. Prin control la nivel scăzut ne referim la definirea directă a forței furnizate motoarelor. Ca urmare, experimentele realizate în acest domeniu par să fie relativ simple. Se poate crede că echilibrarea unui pol pe o mașină în mișcare sau urmărirea unei figuri predefinite cu un braț de robot sunt probleme simple. Cu toate acestea, definirea eficientă a unor soluții adaptive este dificilă. De exemplu, imaginați-vă că un braț de robot urmărește o figură predefinită (așa cum este prezentată în experimentele din secțiunea 4). Apoi, parametri robotului sunt schimbați (de exemplu, punem o greutate suplimentară pe braț), astfel, parametri robotului nu mai sunt adecvați pentru a atinge obiectivul dorit. Algoritmul trebuie (în mod explicit sau implicit) să-și recunoască faptul că parametri fizici s-au schimbat și să-și adapteze comportamentul la noile valori.

În această teză de doctorat ne vom concentra pe modul în care modelul robotul poate fi aproximat din date empirice. Întrucât algoritmul de control se bazează pe modelul robotului, algoritmul de control este aproximat de asemenea. Spunem că modelul robotului și algoritmul de control sunt *învățate* din date empirice. Învățarea modelelor de roboți din

date empirice are avantaje asupra soluțiilor analitice deoarece aceasta se adaptează în mod implicit la schimbările informațiilor senzoriale. Modelul robotul se bazează pe observații empirice, astfel, în cazul în care procesul de generare a datelor se schimbă, observațiile se schimbă, și de asemenea modelului robotului.

Pentru a învăța modelul robotului de date empirice, aplicăm metode de învățare automată. Când se aplică metode din acest domeniu, trebuie să luăm în considerare limitele și dezavantajele metodelor respective. De exemplu, o metodă de regresie poate dă predicții foarte precise, dar timpul de calcul poate fi prea mare pentru aplicații în robotică. O altă caracteristică este timpul de convergență: timpul pentru a colecta numărul necesar de observații nu poate depăși o limită rezonabilă de timp.

Rezumatul este organizat astfel: în secțiunea 2 introducem procesele gaussiene, deoarece le vom folosi extensiv pentru mai multe scopuri. Secțiunea 3 abordează tema de învățare automată în robotică dintr-un punt de vedere teoretic. Secțiunea 4 prezintă o abordare mai practică a temei de învățare în robotică: învățarea modelelor de cinematică inversă din date empirice. Secțiunea 5 extinde sistemul general al învățării modelelor de roboți. Metode din domeniul de învățare prin transfer sunt utilizate pentru îmbunătățirea învățării modelelor de cinematică inversă. Secțiunea 6 conclude rezumatul și prezintă direcții de cercetări viitoare.

Contribuții ale tezei

Teza are următoarele contribuții:

1. Am realizat un studiu comparativ al algoritmilor de învățare prin întărire (reinforcement learning – RL) în contextul de robot control. Am investigat cum funcționează metodele de RL în domenii multi-dimensionali cum este și domeniul robot control. Experimentele arată că modelarea directă a algoritmului de control este superioară față de abordările tradiționale. [Bócsi and Csató, 2011b] [Bócsi and Csató, 2011a].

2. Am folosit procesele gaussiene (Gaussian processes – GP) pentru aproximarea funcției de stare-acțiune din RL împreună cu algoritmul Q-learning. Rezultatele arată că modelul introdus converge la o politică mai precisă. [Jakab et al., 2011] [Bócsi et al., 2010].

3. Am propus o modelare indirectă a cinematicii inverse a roboților. Am folosit o funcție de energie comună a datelor de intrare și ieșire și am obținut predicții prin minimizarea locală a acestei funcții. Am propus trei diferite modalități pentru aproximarea funcției de energie folosind mașini cu suport vectorial (support vector machines), procese gaussiene și procese gaussiene pentru cinematica directă. Experimentele arată că o aproximare precisă a cinematicii inverse poate fi obținută folosind abordarea prezentată. Metoda poate fi folosită și pentru roboți nerigizi unde algoritmii tradiționali nu funcționează [Bócsi et al., 2011b], [Bócsi et al., 2012b], [Bócsi et al., 2011a], [Bócsi et al., 2012a]. Algoritmul numit *structured output Gaussian process* poate fi folosit și pentru a

rezolva probleme generale din domeniul învățare cu structuri în datele de ieșire [Bócsi et al., 2011a].

4. Învățarea modelelor de roboți a fost îmbunătățită folosind metode din domeniul de învățare prin transfer. Ideea principală este de a folosi informații acumulate din alte experimente. Datele adiționale sunt transformate astfel încât ele conțin informații utile pentru experimentul respectiv. Experimentele arată că un model mai precis de cinematică directă este obținut folosind datele adiționale. Rezultatele vor fi publicate în viitor.

2 Metode neparametrice și procese gaussiene

În ultimele decenii metodele neparametrice (de exemplu, mașini cu suport vectorial [Boser et al., 1992; Vapnik, 1999; Schölkopf and Smola, 2002], kernel principal component analysis [Schölkopf and Smola, 2002], kernel density estimation [Parzen, 1962], procese Dirichlet [Ferguson, 1973], procesele gaussiene [Rasmussen and Williams, 2005]) au fost folosite accentuat pe lângă metodele tradiționale, cum sunt principal component analysis [Lee and Verleysen, 2007] sau rețelele neuronale [Barber and Bishop, 1998].

Metodele parametrice tradiționale sunt definite printr-un număr fix de parametri (de exemplu ponderile rețelelor neuronale) și acești parametri sunt setați astfel încât o funcție de eroare este minimizată. Deoarece numărul de parametri este fix, complexitatea modelului nu este flexibilă. Metodele neparametrice nu sunt definite printr-un număr fix de parametri, ci parametri depind de date. Numărul adaptiv al parametrilor rezultă în modele mai flexibile.

Procesele gaussiene sunt modele neparametrice bayesiene care definesc o distribuție pe o funcție caracterizată printr-o valoare medie și o funcție de covarianță (sau kernel) $k(\cdot,\cdot)$ [Rasmussen and Williams, 2005]. Fiind dat datele de intrare $\mathbf{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m$, dorim să găsim o transformare $\mathbf{x} \to \mathbf{y}$ care explică datele cât mai bine. Ca soluție, distribuția unui punct de test $\mathbf{x}_*$ are o distribuție gaussiană cu funcție de valoare medie μ_* și varianță σ_*^2:

$$\begin{aligned} \mu_* &= \mathbf{k}_*^\top (\mathbf{K} + \sigma_0^2 \mathbf{I}_m)^{-1} \mathbf{y} \\ \sigma_*^2 &= k_{**} - \mathbf{k}_*^\top (\mathbf{K} + \sigma_0^2 \mathbf{I}_m)^{-1} \mathbf{k}_*, \end{aligned}$$

unde $\mathbf{K} \in \mathfrak{R}^{m\times m}$ cu $\mathbf{K}^{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, $\mathbf{k}_* \in \mathfrak{R}^{m\times 1}$ cu $\mathbf{k}_*^i = k(\mathbf{x}_i, \mathbf{x}_*)$, $k_{**} = k(\mathbf{x}_*, \mathbf{x}_*)$, $\mathbf{I}_m$ este matricea identică de dimensiune m și σ_0^2 este varianța zgomotului de măsurare.

Funcția de kernel – $k(\cdot,\cdot)$ din notațiile de mai sus – are un rol foarte important în mai multe metode de învățare automată. Menționăm kernelul exponențial cvadratic pentru că le va folosi extensiv în experimentele noastre:

$$k(\mathbf{x}_1, \mathbf{x}_2) = C \exp\left\{ -\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2\omega} \right\},$$

unde C este amplitudinea și ω este frecvența caracteristică. Parametrul $\boldsymbol{\omega}$ poate fi și un

vector. Astfel, putem defini o frecvență pentru fiecare dimensiune de intrare, *i.e.*, $k(\mathbf{x}_1, \mathbf{x}_2) = C \exp\left\{\sum_i (\mathbf{x}_1 - \mathbf{x}_2)_i^2 / 2\omega_i\right\}$.

Principalul dezavantaj al GP este complexitatea cubică de timp și spațiu în numărul datelor de intrare. Pentru a evita problema, diferite metode au fost propuse în literatură [Csató and Opper, 2002; Quiñonero Candela and Rasmussen, 2005; Lawrence et al., 2002; Snelson and Ghahramani, 2006; Titsias, 2009; Lázaro-Gredilla et al., 2010; Ranganathan et al., 2011; Snelson, 2006; Smola and Bartlett, 2001]. Noi adaptăm metoda propusă de Csató and Opper [2002] care poate fi folosită online.

Metoda este o aproximare a distribuției posterioare folosind un algoritm secvențial [Opper, 1998] în care combinăm probabilitatea unui punct de date cu un prior de GP care este posteriorul ultimului epată de aproximare.

Metoda de rarificare este o aproximare a posteriorului exact al procesului gaussian astfel încât discrepanța dintre posteriorul exact și posteriorul aproximat este minimizat. O caracteristică importantă a metodei folosite este că parametri procesului gaussian sunt actualizați și în cazul în care punctul nouă de intrare nu este inclus în mulțimea de bază. Astfel precizia aproximației este îmbunătățită chiar când mulțimea de bază nu se schimbă.

3 Învățare prin întărire

În domeniul de învățare automată, învățarea prin întărire (reinforcement learning – RL) este metoda cea mai apropiată pentru cerințele învățării în robotică [Sutton and Barto, 1998]. Motivul este că procesul de învățare este încorporat într-un mediu mai larg decât învățarea supravegheată sau nesupravegheată. În învățarea supravegheată, dorim să găsim o funcție dintre datele de intrare și ieșire, iar învățarea nesupravegheată caută structuri în date [Bishop, 2006]. În contrast, RL trebuie să ia în considerare și perspectiva de timp, deoarece toate predicțiile actuale vor afecta predicțiile viitoare.

Specific pentru RL este faza de explorare care lipsește la învățarea supravegheată și nesupravegheată. La începutul unui proces de învățare, facem *acțiuni* stohastice pentru a *explora lumea*. În momentul în care avem *informații suficiente* despre lume, acționăm astfel încât să obținem cât mai mult *feedback pozitiv*. Până în acest moment am folosit cuvintele intuitive *acțiune*, *explorare*, *lume*, *informație suficientă*, *feedback pozitiv*. Pentru a avea un algoritm matematic formulat, conceptele precedente trebuie formulate rigid prin procese de decizie Markov [Puterman, 1994].

Utilizarea metodelor din RL poate fi găsită în diverse domenii. De exemplu, strategiile învățate pentru jocuri de strategii (backgammon) nu numai concurează cu jucători umani ci și îl învinge. Cu toate acestea nu toate jocurile pot fi învățate în acest fel. De exemplu jocurile *șah* sau *go* sunt prea complexe și jucătorii umani joacă mult mai bine decât metodele RL.

În recent, metodele RL sunt folosite accentuate și în robotică [Peters et al., 2003; Peters and Schaal, 2008b; Bócsi and Csató, 2011a]. Prin folosirea metodelor RL, probleme dificile, cum sunt pole balancing [Deisenroth and Rasmussen, 2011], ball beam [Benbrahim et al.,

1992], sau problema mountain car [Rasmussen and Kuss, 2004] au fost rezolvate cu succes și bună precizie.

Avantajul principal al controlului bazat pe RL față de controlul analitic este că în primul caz nu trebuie să știm parametri ai robotului. Algoritmul adaptează comportamentul robotului la valorile actuale. Dacă parametri se schimbă în timp, algoritmul de control se schimă implicit.

În ce urmează, vom prezenta conceptele de RL prin procese de decizie Markov și vom face o analiză comparativă între algoritmii de RL în contextul de control de roboți.

Procese de decizie Markov

Formal, RL este definit în termeni de procese de decizie Markov (MDP) [Puterman, 1994]. Un astfel de proces consistă de un tuplu $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \pi)$ unde (1) $\mathcal{S}$ este spațiul de stări; (2) $\mathcal{A}$ este spațiul de acțiuni; (3) $\mathcal{P}^{a}_{ss'} : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathfrak{R}$, cu $\mathcal{P}^{a}_{ss'} = P(s'|s, a)$ este probabilitatea de tranziție; (4) $\mathcal{R}^{a}_{ss'} : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathfrak{R}$ este întărirea primită când acțiunea a este efectuată în starea s urmată de starea s'; (5) $\pi(s, a) : \mathcal{S} \times \mathcal{A} \to [0, 1]$, $\pi(s, a) = P(a|s)$ se numește politica care definește probabilitatea de a efectua acțiunea a în starea s. O traiectorie – sau episod – τ este o secvență de $(s_t, a_t, r_t) \in \mathcal{S} \times \mathcal{A} \times \mathcal{R}$ unde t este indexul pentru timp. Valorile a_{t+1}, s_{t+1}, r_{t+1} sunt obținute prin politica $\pi(s, a)$ și probabilitățile de transmisie.

Prin rezolvarea unui MDP înțelegem găsirea unei politici π' care minimizează întărirea discontată de-a lungul unei traiectorii generată de politica respectivă

$$\pi' = \arg\max_{\pi} \; E_{\pi}\left[\sum_{t} \gamma^{t} r_t\right],$$

unde $r_t \in \tau$, E_π este expectanța condiționată pe π, și $\gamma \in (0, 1]$ este un factor de discont. Obiectivul în RL este rezolvarea unui MDP care definește problema respectivă. Există mai mule abordări ale problemei, în cea ce urmează vom prezenta trei soluții din perspective diferite.

Algoritmi de învățare prin întărire

În continuare, vom prezenta abordările predominante din învățarea prin întărire.

1. **Metodele bazate pe funcții de valoare** [Sutton and Barto, 1998] modelează politica optimală într-un fel indirect cu ajutorul unor funcții de valoare. Aceste funcții măsoară utilitatea unor stări și acțiuni, iar bazată pe aceste valori, politica optimală alege acțiunea cu utilitatea cea mai mare. Odată ce funcția $Q(s, a)$ care exprimeă cât de utilă este acțiunea a în starea s este definită, politica optimală poate fi obținută prin alegerea acțiunii cea mai valoroase, *i.e.*, $\pi(s, a) \sim \text{argmax}_a\, Q(s, a)$. Există mai multe abordări ceea ce privește actualizarea funcției Q, de exemplu Q-learning [Sutton and Barto, 1998].

2. **Metodele de politică gradient** [Peters and Schaal, 2008b] modelează politica directă cu o funcție parametrică π_η, de exemplu rețele neuronale, și actualizează valorile parametrilor prin căutare de gradient [Snyman, 2005] a funcției de obiectiv $J(\eta)$: $J(\eta) = E_{\pi_\eta}\left[\sum_t \gamma^t r_t\right]$. Calculul gradientului $J(\eta)$ este intratabil. Pentru a aproxima acest gradient, diferite abordări au fost propuse: finite difference methods, vanilla policy gradient, natural policy gradient [Peters and Schaal, 2008b]

3. **Metodele evolutive** [Gomez and Miikkulainen, 2003] sunt algoritmi black-box de optimizare. Ei optimizează o funcție parametrică prin ținerea unei populații de parametri – numit individuali – și combinarea acesteia bazată pe valoarea funcției de obiectiv. În RL, individualii sunt politici și funcția de obiectiv este întărirea expectanță mediu [Moriarty et al., 1999].

Este demonstrat că metodele bazate pe funcții de valoare converg asimptotic la o politică optimală în cazul în care reprezentarea precisă a funcției este posibilă [Sutton and Barto, 1998]. Cânt spațiul de stare-acțiune este continuă această funcție trebuie aproximată.

Propunem o aproximare a funcției de valoare prin folosirea proceselor gaussiene și prezentăm algoritmul Q-learning corespunzător. Începem procesul de învățare cu o GP fără puncte de intrare. Apoi, GP-ul este actualizat în fiecare pas de Q-learning. Considerăm un episod $\tau = \{(s_t, a_t, r_t)\}$, iar la fiecare pas, actualizarea arată astfel:

$$q \leftarrow Q_{\text{pred}}(s_t, a_t) + \alpha\left[r_{t+1} + \gamma \max_a Q_{\text{pred}}(s_{t+1}, a) - Q_{\text{pred}}(s_t, a_t)\right],$$

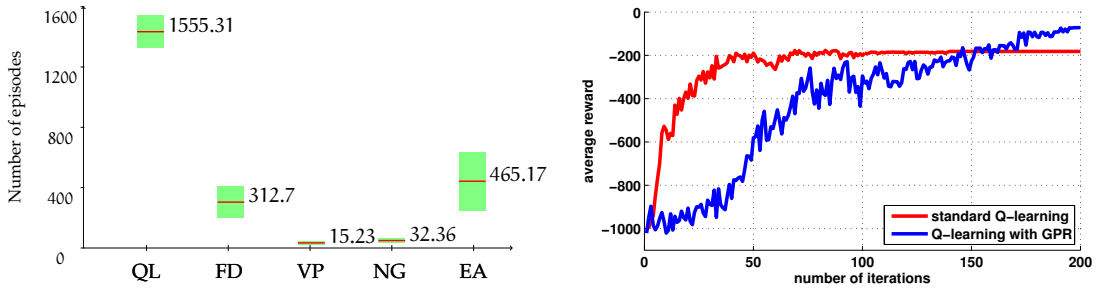
unde $\alpha \in [0, 1]$ este rata de învățare și $Q_{\text{pred}}(s, a)$ este predicția GP-ului actual pentru perechea de stare-acțiune (s, a). Procesul gaussian este actualizat cu data de intrare $((s_t, a_t), q)$.

Experimente

Prezentăm două experimente simulate pentru problemele pole balancing [Deisenroth and Rasmussen, 2011] și mountain car [Rasmussen and Kuss, 2004]. Accentuăm avantajele și dezavantajele metodelor prezentate în domeniul de robotică.

Pentru problema pole balancing, rezultatele sunt bazate pe 393 experimente și sunt demonstrate pe figura 1a, unde și varianța de convergența este prezentată. Ca măsură de performanță am folosit numărul mediu de episoade necesare pentru a găsi a politică bună. Din simulații se vede că metodele bazate pe căutare gradient produc rezultate superioare față de ceilalți algoritmi. Între acestea vanilla policy gradient obținea rezultatele cele mai bune. Metoda Q-learning este stabilă dar produce rezultatele cele mai slabe. Cauza este spațiul de stare continuu și multidimensională.

Bazată pe 400 de executări ale problemei mountain car, experimentele arată că algoritmul Q-learning converge mai repede dacă nu folosim aproximare cu GP – 61 de episoade au fost necesare pentru Q-learning și 146 pentru algoritmul cu extensia GP. Acest rezultat nu este surprinzător deoarece folosind o aproximare a funcției de valori pierdem precizie. Este mult mai important că algoritmul cu aproximare GP are ca rezultat o politică mai bună. Rezultatele sunt prezentate pe figura 1b.



(a) Pole balancing.

(b) Mountain car.

Figura 1: Rezultate ale experimentelor. Performanța la experimentul pole balcing era măsurată în numărul de episoade până la convergență, pentru Q-learning (QL), finite difference method (FD), vanilla policy gradient method (VP), natural gradient method (NG) și evolutionary algorithm (EA).

4 Învățarea modelelor de roboți pentru control de urmărire

În această secțiune, ne vom îndrepta atenția spre probleme mai realiste de robotică: controlul eficient al unui robot pentru control de urmărire (tracking control).

În general, controlul roboților este bazat pe modelul robotului. În cazul tradițional acest model este definit prin parametri fizici ai robotului. Bazat pe aceste valori configurația de articulație dorită (numit modelul cinematic) sau forțele dorite (numit modelul dinamic) au forme analitice de asemenea. Această abordare este folosită unde arhitectura robotului este fixă și condițiile externe nu se schimbă. În condițiile în care arhitectura robotului și mediul de execuție sunt fixe, modelele analitice rezultă în control precis și eficient.

În condiții diferite, soluțiile analitice au dezavantaje serioase. (1) Roboții moderni devin mult mai mult complecși și definirea modelului în funcție de toți parametri este prea complexă. Deși soluția analitică există, cerințele computaționale sunt prea scumpe. (2) Soluțiile analitice fac presupuneri lineare despre modelul robotului care rezultă în control ineficient și inexact. (3) Soluțiile analitice eșuează când valorile parametrilor nu sunt precise sau sunt necunocute. (4) Incorectitudine în modelul de robot poate fi cauzată și de zgomotul din datele senzoriale.

În această secțiune propunem o abordare adaptivă pentru a obține modelul robotului. Scopul nu este de a defini un model de control bazat pe structura fizică a robotului ci ca o funcție dintre datele de intrare senzoriale și datele de ieșire de control. Nu suntem interesați în reprezentarea actuală a acestei funcție ci numai datele de intrare și ieșire sunt relevante. Spațiul de funcție împreună cu datele empirice definesc modelul robotului. În următoarele, definim mai precis problema de control de urmărire.

Control de urmărire

De obicei, problema de control de urmărire (tracking control) este formulată în spațiul de efector (task-space) când efectorul robotului trebuie să-și urmeze o traiectorie predefinită.

Soluția problemei anterioare nu este unică. Pentru roboți redundanți, funcția din spațiul de efector la spațiul de articulații (joint space) nu este unică: pentru o poziție de efector există mai multe configurații de articulații care formează un spațiu neconvex de soluții [D'Souza et al., 2001]. În această secțiune, propunem un algoritm bazat pe învățarea automată care este capabil de a și controla roboți nerigizi unde soluțiile standarde nu funcționează.

Algoritmul de control de urmărire are trei pași: (1) aproximăm un model comun al coordinatelor de efector și al coordinatelor de articulații folosind tehnici de învățare automată. (2) aplicăm optimizare locală pentru a obține cinematica inversă notată cu f^{-1}; (3) bazată pe cinematica inversă, folosim un controlor de articulații la toate gradele de libertate (DoF) ale robotului pentru a obține forțele necesare. În continuare, prezentăm pasul unu și doi al algoritmului, deoarece pasul trei este semnificativ mai ușor de rezolvat [Nguyen-Tuong and Peters, 2010].

Învățarea indirectă a modelelor de roboți

Observația principală este că un model $E(\mathbf{x}, \theta)$ dintre datele de intrare și ieșire este bine definit iar predicții pot fi obținute prin minimizarea modelului la un punct de intrare dat, *i.e.*,

$$f^{-1}(\mathbf{x}) \quad \stackrel{\circ}{=} \quad \underset{\theta \in \Theta}{\operatorname{argmin}} \, E(\mathbf{x}, \theta), \tag{1}$$

unde $\mathbf{x}$ notează poziția de efector și θ notează poziția de articulații.

O întrebare importantă este cum să efectuăm minimizarea din ecuația (1). Întrebarea privește și problema de ne-unicitate a funcției de cinematică inversă: cum să efectuăm minimizarea în cazul în care o poziție de efector $\mathbf{x}^{\text{desired}}$ poate fi atinsă de mai multe articulații θ_1 și θ_2 (figura 2)? În acest caz, algoritmul trebuie să dea ca predicție θ_2 pentru a evita mișcări bruște. Acest comportament este favorabil pentru a obține traiectorii netede de articulații. Propunem a începe căutarea de gradient de la poziția curentă de articulație θ^{current} a robotului.

A doua întrebare importantă este cum să modelăm $E(\cdot, \cdot)$ pentru a obține un algoritm eficient care poate fi folosit în aplicații din lumea reală. Propunem trei abordări posibile, folosind *joint kernel support estimation*, *structured output Gaussian processes*, și o metodă bazată pe cinematica directă a robotului.

Folosind **Joint kernel support estimation** (JKSE), modelăm funcția de energie ca probabilitatea comună negativă a datelor $\mathbf{x}$ și θ, *i.e.*,

$$E(\mathbf{x}, \theta) \quad \stackrel{\circ}{=} \quad -p(\mathbf{x}, \theta). \tag{2}$$

JKSE modelează distribuția comună a datelor de intrare și ieșire ca un model log-linear al o funcției de feature [Lampert and Blaschko, 2009]. După simplificări, predicția pentru cinematica inversă arată astfel:

$$f^{-1}(\mathbf{x}) \quad = \quad \underset{\theta \in \Theta}{\operatorname{argmax}} \; \boldsymbol{w}^{\top} \phi(\mathbf{x}, \theta),$$

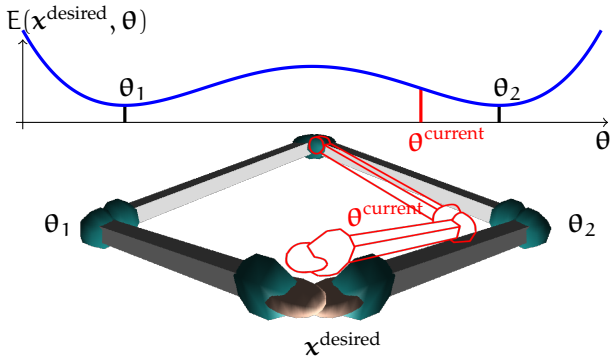


Figura 2: Ilutația schemei de predicție a funcției de cinematică inversă. În datele de intrare poziția $x^{desired}$ a fost atinsă de două configurații de articulație θ_1 și θ_2, deci $E(x^{desired}, \theta_1) = E(x^{desired}, \theta_2)$. Ca o predicție, algoritmul va alege θ_2 pentru că configurația curentă de articulație $\theta^{current}$ este mai aproape de θ_2.

unde w sunt parametri care generează distribuția $p(x, \theta)$ care explică datele $\mathbf{D} = \{(x_i, \theta_i)\}_{i=1}^m$ cel mai bine. Valorile parametrilor w sunt obținute folosind mașini cu suport vectorial cu o singură clasă (one-class support vector machines) [Schölkopf et al., 2001].

Folosind **structured output Gaussian processes** (SOGP), funcția de energie $E(\cdot, \cdot)$ este exprimată ca și valoarea posterioară de medie a unui GP negativă, *i.e.*,

$$E(x, \theta) \stackrel{\circ}{=} -\mu_{(x,\theta)}. \tag{3}$$

Datele de intrare la GP sunt datele comune de intrare și ieșire reprezentate de o funcție de feature $\phi(x, \theta)$ și datele de ieșire este $\mathbf{1}$ la toate datele. O astfel de mulțime de date poate conduce la overfitting. Pentru a evite overfitting, un prior puternic trebuie aplicat. În restul secției, folosim un prior zero pentru a păstra notațiile simple. Valoarea medie de posterior arată astfel:

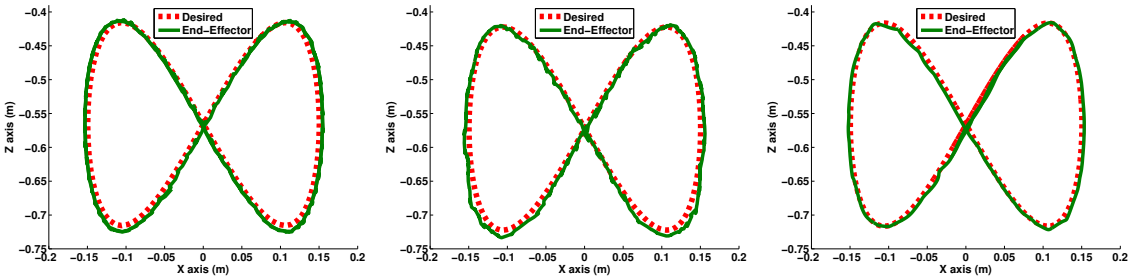
$$\mu_{(x,\theta)} = k_{(x,\theta)}^\top (\mathbf{K} + \sigma_0^2 \mathbf{I}_m)^{-1} \mathbf{1},$$

unde $\mathbf{K} \in \mathfrak{R}^{m \times m}$ cu $\mathbf{K}^{ij} = k((x_i, \theta_i), (x_j, \theta_j))$, $k_{(x,\theta)} \in \mathfrak{R}^{m \times 1}$ cu $k^i_{(x,\theta)} = k((x_i, \theta_i), (x, \theta))$, $k_{(x,\theta)(x,\theta)} = k((x, \theta), (x, \theta))$, $\mathbf{I}_m$ este matricea identică, σ_0^2 este varianța zgomotului de măsurare, și $\mathbf{1}$ este vectorul cu valori 1 de dimensiune m.

A treia abordare, numită "**forward Gaussian process modeling**" (FWGP), este bazată pe observația că modelul cinematic direct – notat cu $f(\cdot)$ – este mult mai ușor de modelat decât funcția inversă. Bazându-se pe această observație, construim funcția de energie astfel încât ea va depinde explicit pe cinematica directă. Odată ce știm funcția de cinematică directă, funcția de energie este definită ca și distanța Euclidiană dintre poziția dorită de efector și poziția anticipată de modelul cinematic direct, *i.e.*,

$$E(x, \theta) \stackrel{\circ}{=} \|x - f(\theta)\|^2. \tag{4}$$

Modelăm funcția de cinematică directă cu un GP. Fiind dat datele de instruire $\mathbf{D} = \{(x_i, \theta_i)\}_{i=1}^m$ cu datele de intrare θ_i și de ieșire x_i, predicția pentru o nouă θ are o distribuție



(a) JKSE. (b) SOGP. (c) FWGP.

Figura 3: Rezultatele urmăririi ale figurii de opt cu învățare offline. Se vede că o precizie bună a fost obținută cu fiecare model.

gaussiană cu valoare de medie μ_θ

$$\mu_\theta = \sum_{i=1}^{m} \alpha^i k(\theta, \theta_i) = \mathbf{k}_\theta^\top \boldsymbol{\alpha}, \tag{5}$$

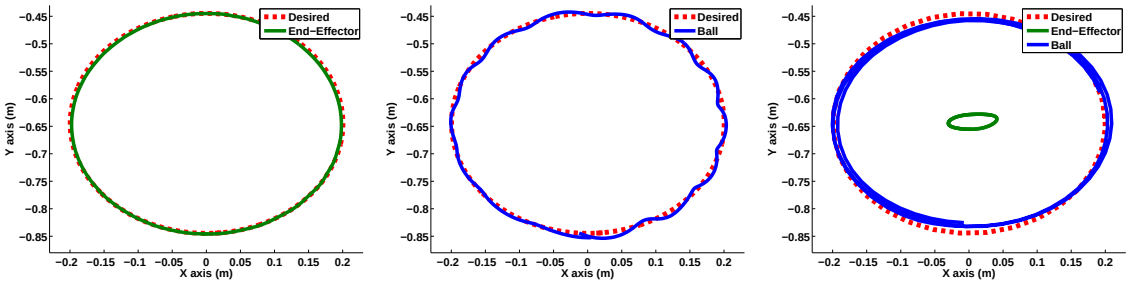
unde $k_{\theta\theta} = k(\theta, \theta)$, $\mathbf{k}_\theta \in \Re^{m\times 1}$ este un vector cu elemente $\mathbf{k}_\theta^i = k(\theta, \theta_i)$ și $\mathbf{K} \in \Re^{m\times m}$ este o matrice cu elemente $\mathbf{K}^{ij} = k(\theta_i, \theta_j)$. Funcția $k : \Re^n \times \Re^n \to \Re$ este un kernel și $\boldsymbol{\alpha} \in \Re^{m\times 1}$, unde $\boldsymbol{\alpha} = (\mathbf{K} + \mathbf{I}_m \sigma_0^2)^{-1}\mathbf{x}$ sunt parametri GP-ului. Valoare medie posterioară a procesului gaussian este folosită pentru predicția modelului cinematic direct din equația (5), *i.e.*, $f(\theta) = \mu_\theta$.

Gradientele pentru funcțiile de energie din ecuația (2), ecuația (3), și ecuația (4) au forme analitice, deci, căutarea gradient din ecuația (1) poate fi făcută eficient [Snyman, 2005].

Experimente

Prezentăm o evaluare empirică a metodelor prezentate pentru problema de control de urmărire. Algoritmul este aplicat pentru a învăța cinematica inversă a robotului Barrett WAM [Bócsi et al., 2011b] și pentru a urmări o figură de opt cu setări diferite. Rezultatele urmăririi sunt prezentate pe figura 3, unde se vede că o precizie bună a fost obținută. Experimentele arată că FWGP rezultă în modele mai precise decât JKSE sau SOGP care converg la precizia modelului analitic. Rezultatul este puțin surprinzător în lumina numărul punctelor pe care predicția era bazată. Modelul JKSE a fost bazat pe 8456 puncte, SOGP pe 200 puncte și FWGP pe 31 puncte.

Într-un experiment diferit, am modificat modelul simulat al robotului făcând-l mai complex prin fixarea unei mingi pe efectorul brațului de robot cu o coardă de 20 cm. Mișcarea oscilatoare a mingii a rezultat într-un sistem neliniar. Cu aceste setări, am făcut control de urmărire când poziția mingii a fost considerată în loc de poziția efectorului. Problema a fost de a urmări un circ cu o rază de 20 cm (figura 4a) pe planul orizontal. Experimentul a fost efectuat cu două setări diferite: (1) când punctul de destinație a mișcat încet (o rotație a fost făcută în 24 de secunde) și (2) când punctul de destinație a mișcat rapid (o rotație a fost făcută în 0.62 de secunde).



(a) Traiectoria efectorului cu mișcare încetă.

(b) Traiectoria mingii cu mișcare încetă.

(c) Traiectoria efectorului și al mingii cu mișcare rapidă.

Figura 4: Control de urmărire al unui circ cu un robot Barrett WAM simulat cu o minge fixată pe efector.

În primul caz, FWGP a învățat să-și miște efectorul deasupra circului dorit în timp ce mingea se mișca de-a lungul traiectoriei dorite. Viteza efectorului era aceeași ca și viteza mingii. Pentru a învăța un model care rezultă la precizia arătată pe figurile 4a și 4b aveam nevoie de patru minute de interacțiune. Modelul GP era bazat pe 20-25 de puncte. În al doilea caz, efectorul s-a mișcat cu o viteză mare în interiorul circului iar folosind forța centrifugală, mingea a mișcat pe de-a lungul traiectoriei dorite (figura 4c). După 20 de minute de interacțiune, modelul GP a fost bazat pe 13-15 puncte. Urmărirea circului cu efectorul robotului la viteza aceasta era imposibilă deoarece aș deteriora robotul.

Accentuăm că aceiași parametri au fost folosiți în amândouă experimente, deci comportamentul adaptiv depinde slab pe hyper-parametri modelului GP. În primul caz, FWGP a considerat mișcarea oscilatorie a mingii ca zgomot iar în al doilea caz, forța centrifugală a fost încorporată în modelul GP.

5 Învățarea modelelor de roboți folosind metode de transfer

Motivația acestei secțiuni vine din învățarea umană. O diferența fundamentală dintre învățarea umană și învățarea automată este că robotul nu are cunoștințe a-priori despre lume, în timp ce oamenii au experiențe din trecut. În acest context, deși problema țintă nu a fost efectuată de omul respectiv, abilitățile rezultate din experientele trecute va ușura învățarea problemei respective. În această secțiune considerăm scenariul când robotul folosește experiente trecute pentru a îmbunătății viteza învățării.

Scopul de a transfera cunoștințe dintre diferite probleme nu este nou. Observația a fost folosită în învățarea automată cu numele de *învățare prin transfer* (transfer learning) [Pan and Yang, 2010; Arnold et al., 2007; Pan et al., 2008; Taylor and Stone, 2009]. Învățarea prin transfer este bazată pe observația că o problemă de învățare automată poate fi îmbunătățită dacă cunoștințele disponibile din alte experimente pot fi refolosite. În robotică, învățarea prin transfer a fost folosită mai ales în învățarea prin întărire [Thrun and Mitchell, 1993].

Metode de învățare prin transfer în robotică

Scopul este de a îmbunătăți procesul de învățare al modelului de robot când presupunem că informații din experimente trecute pot fi folosite în formă de date adiționale. Definim o sarcină sursă (source task), o problemă care este deja rezolvată, și o sarcină țintă (target task), o problemă care este dificil de învățat. Considerăm problema când sarcina sursă are datele $\mathbf{D}^s = \{(\theta_i^s, \mathbf{x}_i^s)\}_{i=1}^N$ cu N puncte și dorim să învățăm o sarcină țintă cu date de învățare $\mathbf{D}^t = \{(\theta_i^t, \mathbf{x}_i^t)\}_{i=1}^K$ cu K puncte.

În această teză, dorim să îmbunătățim învățarea funcției de cinematică directă deci presupunem că datele de intrare sunt configurațiile de articulații ale robotului și datele de ieșire sunt pozițiile de efector ale robotului.

Ca un prim pas, reducem dimensiunea datelor la aceeași dimensiune. În experimentele noastre am folosit principal component analysis (PCA) ca metodă de reducere a dimensiunii. Prin aplicarea metodei PCA, presupunem o relație lineară dintre sub-spațiile cu dimensiune scăzută și înaltă. Mai întâi, centrăm datele, *i.e.*, scădem valoare medie, și le împărțim cu deviația standardă. Proiectarea arată astfel:

$$\begin{aligned} \mathbf{s} &= \mathbf{B}_s(\mathbf{d}^s - \boldsymbol{\mu}_s) \\ \mathbf{t} &= \mathbf{B}_t(\mathbf{d}^t - \boldsymbol{\mu}_t), \end{aligned}$$

unde $\mathbf{d}^s \in \mathbf{D}^s$ și $\mathbf{d}^t \in \mathbf{D}^t$ sunt datele sarcinii sursă și sarcinii țintă, iar $\mathbf{s} \in \boldsymbol{M}^s$ și $\mathbf{t} \in \boldsymbol{M}^t$ sunt punctele respective din varietatea cu dimensiune redusă. Valorile $\boldsymbol{\mu}_s = \mathbf{E}\{\mathbf{D}^s\}$ și $\boldsymbol{\mu}_t = \mathbf{E}\{\mathbf{D}^t\}$ sunt valorile medii ale datelor originale. Matricele $\mathbf{B}_s$ și $\mathbf{B}_t$ sunt matrice de transformare astfel încât varianțele mulțimilor $\boldsymbol{M}^s$ și $\boldsymbol{M}^t$ să fie maximizați. Pentru detalii consultă Lee and Verleysen [2007].

În pasul al doilea, modelăm funcția dintre cele două varietăți ca o proiecție lineară $f : \boldsymbol{M}^s \rightarrow \boldsymbol{M}^t$, cu

$$f(\mathbf{s}) = \boldsymbol{A}\mathbf{s}, \tag{6}$$

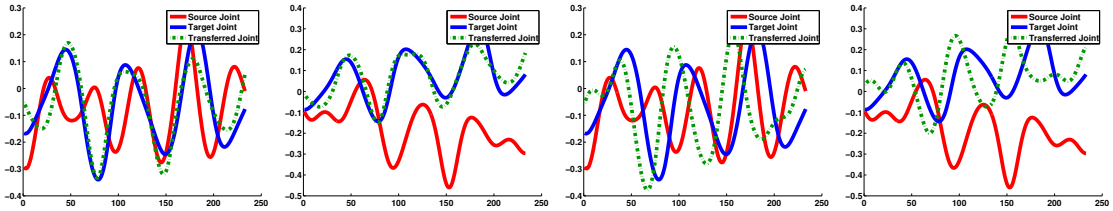
unde $\boldsymbol{A} \in \Re^{J \times J}$ este o matrice de transformare cu dimensiune J. Definim două scenarii de aliniere.

În primul scenariu, știm o corespondență directă dintre punctele mulțimilor. Prin corespondență directă înțelegem că $\mathbf{D}^s$ și $\mathbf{D}^t$ au același număr de puncte și punctele sunt împerecheate. Această setare poate fi folosită când aceeași sarcină a fost efectuată în spațiul sursă și în spațiul țintă.

În al doilea scenariu, nu există nici o corespondență dintre punctele mulțimilor $\mathbf{D}^s$ și $\mathbf{D}^t$. Este posibil că mulțimile nu au aceiași număr de puncte, *i.e.*, $|\mathbf{D}^s| \neq |\mathbf{D}^t|$.

- În primul caz – numit **aliniere prin corespondență directă** –, presupunem o funcție lineară și minimizăm eroarea transformației. Dorim să-l găsim valorile parametrilor $\boldsymbol{A}$ din equația (6) astfel încât expectanța de eroare să fie minimizată, *i.e.*,

$$\boldsymbol{A} = \underset{\boldsymbol{A}}{\operatorname{argmin}}\, \mathbf{E}\left\{(\mathbf{t} - \boldsymbol{A}\mathbf{s})^\top (\mathbf{t} - \boldsymbol{A}\mathbf{s})\right\},$$



(a) Joint 1. (b) Joint 3. (c) Joint 1. (d) Joint 3.

Figura 5: Rezultate pentru aliniere cu corespondență directă și aliniere dură.

unde $\mathbf{E}\{\cdot\}$ notează operatorul de expectanță. Soluția pentru matricea A are următoarea formă:

$$\mathbf{A} \quad = \quad \mathbf{\Sigma}_{ss}^{-1}\mathbf{\Sigma}_{ts}, \tag{7}$$

unde $\mathbf{\Sigma}_{ss}$, $\mathbf{\Sigma}_{tt}$, și $\mathbf{\Sigma}_{ts}$ sunt matrice de covarianță.

- În cazul al doilea – numit **aliniere dură** –, distanța dintre distribuțiile definite de punctele mulțimilor $\mathbf{M}^s$ și $\mathbf{M}^t$ este minimizată. În ce urmează presupunem că datele au o distribuție gaussiană, iar noi minimizăm distanța dintre două distribuții gaussiene $p(\mathbf{M}^s)$ și $p(\mathbf{M}^t)$. Definită ca și divergența Kullback-Leibler [Kullback, 1959], distanța dintre două gaussiene are formă analitică. Ca rezultat, o matrice $\mathbf{A}$ care minimizează această distanță este soluția ecuației următoare:

$$\mathbf{\Sigma}_{tt} \quad = \quad \mathbf{A}\mathbf{\Sigma}_{ss}\mathbf{A}^\top.$$

Această expresie este pătratică în $\mathbf{A}$ și nu are o soluție unică. O matrice $\mathbf{A}$ care este soluție a equației precedente poate fi obținută folosind decompoziția de valoare proprii a matricelor de covarianță [Trefethen and Bau, 1997]. Soluția propusa arată astfel:

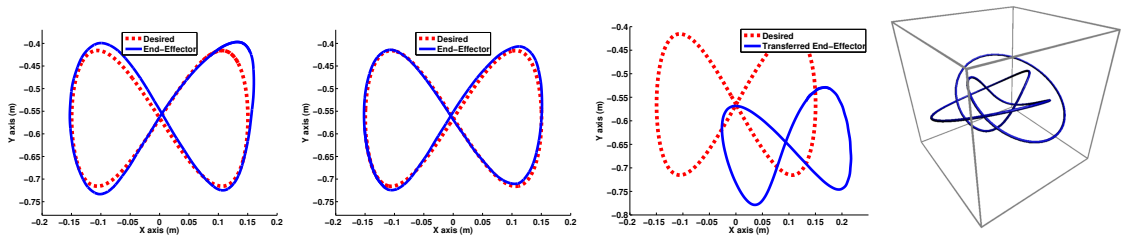
$$\mathbf{A} \quad = \quad \mathbf{U}_t\mathbf{\Lambda}_t^{1/2}\mathbf{\Lambda}_s^{-1/2}\mathbf{U}_s^\top,$$

unde $\mathbf{U}_s$ și $\mathbf{U}_t$ sunt matrice de rotație (*i.e.*, $\mathbf{U}_s\mathbf{U}_s^\top = \mathbf{I}$) cu vectori proprii ai $\mathbf{\Sigma}_{ss}$ și $\mathbf{\Sigma}_{tt}$, iar $\mathbf{\Lambda}_s$ și $\mathbf{\Lambda}_t$ sunt matrice diagonali cu valori proprii ai $\mathbf{\Sigma}_{ss}$ și $\mathbf{\Sigma}_{tt}$ respectiv.

Experimente

Am condus experimente pe roboți cu diferite arhitecturi pentru a accentua două proprietăți ale metodei: (1) pierderea de informație indusă de reducția de dimensiune nu este semnificantă și (2) puterea expresivă a funcției lineare este suficientă pentru a obține aliniere eficientă.

Experimentul a fost efectuat la un simulator al brațului Sarcos Master [Nguyen-Tuong et al., 2009b] cu opt grade de libertate și la un simulator al brațului Barrett WAM [Nguyen-Tuong et al., 2009b] cu șapte grade de libertate. Scupul în ambele probleme au fost urmărirea unui nod trefoil (figura 6d) cu efectorul roboților. Am folosit algoritmul de control analitic al roboților pentru a colecta date. După urmărirea figurii cu amândoi roboți pe timp de



(a) După transfer de date *fără* învățare adițională.

(b) După transfer de date *cu* învățare adițională.

(c) Transfer de la braț Master la braț de Barrett.

(d) Nod trefoil.

Figura 6: Rezultate de control de urmărire după trei secunde de interacțiune și folosirea metodei propuse.

un minut, aveam puncte cu corespondență directă. Am folosit metoda de aliniere cu corespondență directă pe aceste date. Figura 5a și figura 5b arată că după estimarea matricei $\mathbf{A}$ din equația (7), am transferat cu succes (verde) datele de la sarcina sursă (roșu) la sarcina țintă (albastru). Pentru a vedea câte informații pot fi păstrate cu metoda de aliniere dură am aplicat și metoda respectivă. Rezultatele sunt prezentate pe figura 5c și figura 5d. Se vede că transformarea nu este precisă dar valoarea media și varianța sunt păstrate, cum era de așteptat.

În experimentul următor, am transformat direct o figură de la brațul Master la brațul Barret prin folosirea matricei obținută din experimentul precedent. Am urmărit o figură de opt care a fost plasată în interiorul spațiului definit de nodul trefoil cu brațul Master. După transformarea coordinatelor de articulații și urmărirea acestei coordinate cu brațul Barrett, am obținut figura prezentată pe figura 6c. Prezentarea figurii dorite pe figura 6c este înșelătoare deoarece nu există o figura transformată *corect*. Am definit figura dorită ca și figura urmărită de controlerul analitic al brațului Barrett cu postura inițială asemănătoare cu cea folosită pentru nodul trefoil.

În experimentul final, am folosit aceleași arhitecturi de roboți pentru sarcina sursă și sarcina țintă ca și în experimentul anterior. Pentru sarcina sursă am folosit datele colectate în experimentul precedent. Sarcina țintă era de a accelera învățarea a funcției de cinematică directă a brațului Barrett. Modelul de cinematica directă a fost aproximat folosind procese gaussiene și acest model a fost folosit pentru control de urmărire. Fără a folosi metode de învățare prin transfer robotul are nevoie de timp între 20 de secunde și patru minute pentru a învăța acest model. După numai trei secunde de mișcări cvasi-stohastice am folosit metoda de aliniere dură. Am oprit procesul de învățare după trei secunde și am aplicat metoda. Figura 6a prezintă figura de opt obținută. Forma figurii de opt nu este perfectă, dar a fost obținută după numai trei secunde de interacțiune. Am repetat experimentul dar acum procesul de învățarea nu a fost oprit după trei secunde ci numai datele de la brațul Master au fost folosite. Figura 6b prezintă că dacă procesul de învățare nu este oprit o figura foarte precisă poate fi obținută.

6 Concluzii și cercetări în viitor

În această teză am adresat problema învățării modelelor de roboți în diferite contexte.

În primul rând am investigat metodele RL în domeniul robotică. Scopul era de a vedea cum funcționează metodele RL în acest domeniu, caracterizate cu spații de stare continuu și multidimensionali. Experimentele arată că aproximarea directă a politicii rezultă în control de robot mai precis. În continuare o extensie a algoritmului Q-learning a fost prezentată. Funcția de valoare stare-acțiune a fost modelată folosind GP. Experimentele arată că această extensie converge la o soluție mai bună decât algoritmul Q-learning standard .

Mai departe, am adresat o problemă mai specifică de învățare de modele de roboți, învățarea modelului de cinematică inversă. Noutatea algoritmului este că modelul de cinematică inversă este modelat indirect astfel un singur model poate conține mai multe soluții la o singură poziție de efector, iar soluția corectă este selectată pe baza principiului că mișcarea netedă a robotului este o cerință naturală. Experimentele conduse la brațul de robot Barrett arată că putem să obținem precizie bună cu algoritmul propus pentru probleme de control de urmărire care converg la precizia soluției analitice. Metoda propusă a fost capabilă de a controla un robot ne-rigid la care toate celelalte metode au eșuat.

Am propus o metodă de îmbunătățire a învățării de model de robot folosind paradigma din domeniul învățare prin transfer. Experimentele arată că modele precise de roboți au fost obținute folosind algoritmul propus cu o convergență mai rapidă.

Cercetări viitoare

Metodele propuse pot fi îmbunătățite în mai mule feluri. Extensia de GP a funcției de valoare stare-acțiune are dezavantaje serioase deja arătate în literatură. Aceste dezavantaje trebuie să fi atenuate. Folosirea faptului că modelul GP ne oferă o distribuție ca predicție, ar fi util să folosim și informația oferită de varianța posterioară și nu numai valoarea de medie.

Pentru învățarea modelului de cinematică inversă ar fi interesant de a încerca alte metode de învățare automată pentru a modela funcția de energie. Folosirea altor metode de căutare poate produce predicții mai rapide și mai precise. De asemenea este foarte important de a încerca metoda propusă la roboți cu mai multe grade de libertate.

Ideea de a folosi o funcție de energie și a obține predicții prin minimizarea acestei funcții poate fi folosită pentru alte modele de roboți, de exemplu, modele dinamice sau control în spațiul operațional.

Folosirea altor metode de învățarea prin transfer în contextul de robotică este o idee promițătoare. O altă direcție poate fi de a folosi informații din învățarea altor modele, de exemplu, cum ar fi învățarea modelului dinamic poate ajuta învățarea cinematică a robotului.

Bibliografie

M. Alvarez and N. D. Lawrence. Sparse convolved Gaussian processes for multi-output regression. In *Neural Information Processing Systems*, pages 57–64, 2008.

S.-I. Amari and H. Nagaoka. *Methods of Information Geometry (Translations of Mathematical Monographs)*. American Mathematical Society, 2001.

V. R. d. Angulo and C. Torras. Learning inverse kinematics via cross-point function decomposition. In *Proceedings of the International Conference on Artificial Neural Networks*, (ICANN 2002), pages 856–864, London, UK, UK, 2002. Springer-Verlag.

B. D. Argall, S. Chernova, M. Veloso, and B. Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57:469–483, 2009.

A. Arnold, R. Nallapati, and W. W. Cohen. A comparative study of methods for transductive transfer learning. In *Proc. IEEE Int. Conf. on Data Mining Workshops*, pages 77–82, 2007.

G. H. Bakir, T. Hofmann, B. Schölkopf, A. J. Smola, B. Taskar, and S. V. N. Vishwanathan, editors. *Predicting Structured Data*. Neural Information Processing. The MIT Press, 2007.

D. Barber. Bayesian methods for supervised neural networks. In *Handbook of Brain Theory and Neural Networks*. MIT Press, 2002.

D. Barber and C. M. Bishop. *Ensemble learning in Bayesian neural networks.*, pages 215–237. Springer-Verlag, Berlin, 1998.

H. Benbrahim, J. Doleac, J. Franklin, , and O. Selfridge. Real-time learning: A ball on a beam. In *Proceedings of the International Joint Conference on Neural Networks*, volume 1, pages 98–103, 1992.

D. P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 1995.

C. M. Bishop. *Pattern recognition and machine learning*. Springer, 1st edition, 2006.

B. Bócsi and L. Csató. Reinforcement learning algorithms in robotics. In M. Frentiu, H. F. Pop, and S. Motogna, editors, *KEPT-2011:Knowledge Engineering Principles and Techniques International Conference, Selected Papers.*, pages 131–143. Presa Universitara Clujeana, 2011a.

B. Bócsi, L. Csató, and Jan Peters. Structured output Gaussian processes. Technical report, Babes-Bolyai University, 2011a. URL http://www.cs.ubbcluj.ro/~bboti/pubs/sogp_2011.pdf.

B. Bócsi, D. Nguyen-Tuong, L. Csató, B. Schoelkopf, and J. Peters. Learning inverse kinematics with structured prediction. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 698–703, San Francisco, USA, 2011b.

B. Bócsi, L. Csató, B. Schölkopf, and J. Peters. Indirect robot model learning for tracking control. *Robotics and Autonomous Systems (submitted on 10 September 2012)*, 2012a.

B. Bócsi, P. Hennig, L. Csató, and J. Peters. Learning tracking control with forward models. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 259–264, St. Paul, MN, USA, 2012b.

B. A. Bócsi and L. Csató. Reinforcement learning algorithms in robotics. *Studia Universitatis Babes-Bolyai Series Informatica*, LVI(2):61–67, 2011b.

B. A. Bócsi, H. Jakab, and L. Csató. Nonparametric methods in robotics. In *Proceedings of the 8th Joint Conference on Mathematics and Computer Science (Abstract)*, page 8, Komarno, Slovakia, 2010.

E. V. Bonilla, K. M. Chai, and C. K. I. Williams. Multi-task Gaussian process prediction. In J. C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20 (NIPS)*. MIT Press, Cambridge, MA, 2008.

B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, COLT '92, pages 144–152, New York, NY, USA, 1992. ACM.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

G. Cauwenberghs and T. Poggio. Incremental and decremental support vector machine learning. In *Advances in Neural Information Processing Systems*, volume 13, 2001.

K. M. Chai, C. Williams, S. Klanke, and S. Vijayakumar. Multi-task Gaussian process learning of robot inverse dynamics. In *Advances in Neural Information Processing Systems (NIPS)*, pages 265–272, 2009.

C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Y.-S. Choi. Least squares one-class support vector machine. *Pattern Recognition Letters*, 30: 1236–1240, 2009.

J. J. Craig. *Introduction to Robotics: Mechanics and Control*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 1989.

R. Crites and A. Barto. Improving elevator performance using reinforcement learning. In *Neural Information Processing Systems*, pages 1017–1023. MIT Press, 1996.

L. Csató and M. Opper. Sparse on-line Gaussian processes. *Neural Computation*, 14(3):641–668, 2002.

W. Dai, G.-R. Xue, Q. Yang, and Y. Yu. Transferring naive Bayes classifiers for text classification. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, pages 540–545, 2007a.

W. Dai, Q. Yang, G. R. Xue, and Y. Yu. Boosting for transfer learning. In *Proceedings of the 24th international conference on Machine learning (ICML)*, pages 193–200. ACM, 2007b.

V. R. de Angulo and C. Torras. Learning inverse kinematics: reduced sampling through decomposition into virtual robots. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 38(6):1571–1577, 2008.

J. F. G. de Freitas. *Bayesian methods for neural networks*. PhD thesis, Trinity College. University of Cambridge, 1999.

B. de Kruif and T. de Vries. Pruning error minimization in least squares support vector machines. *IEEE Transactions on Neural Networks*, 14(3):696–702, 2003.

M. P. Deisenroth and C. E. Rasmussen. PILCO: A Model-Based and Data-Efficient Approach to Policy Search. In L. Getoor and T. Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

M. P. Deisenroth, C. E. Rasmussen, and J. Peters. Gaussian process dynamic programming. *Neurocomputing*, 72(7-9):1508–1524, 2009.

D. Demers and K. Kreutz-Delgado. Learning global direct inverse kinematics. In *Advances in Neural Information Processing Systems*, pages 589–595. Morgan Kaufmann, 1992.

R. Detry, D. Kraft, O. Kroemer, L. Bodenhagen, J. Peters, N. Krüger, and J. Piater. Learning Grasp Affordance Densities. *Paladyn Journal of Behavioral Robotics*, 2(1):1–17, 2011.

L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Applications of Mathematics. Springer, 1996.

A. D'Souza, S. Vijayakumar, and S. Schaal. Learning inverse kinematics. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*. Piscataway, NJ: IEEE, 2001.

A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing (Natural Computing Series)*. Springer, 2008.

Y. Engel, S. Mannor, and R. Meir. Bayes meets Bellman: the Gaussian process approach to temporal difference learning. In *Proceedings of the 20th International Conference on Machine Learning (ICML)*, pages 154–161, 2003.

Y. Engel, S. Mannor, and R. Meir. Reinforcement learning with Gaussian processes. In *Proceedings of the 20th International Conference on Machine Learning*, pages 201–208, 2005.

L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:594–611, 2006.

T. S. Ferguson. A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230, 1973.

M. Galassi, J. Davies, J. Theiler, B. Gough, G. Jungman, M. Booth, and F. Rossi. *Gnu Scientific Library: Reference Manual*, 2003. Software `http://www.gnu.org/software/gsl/manual/html_node/index.html`.

M. Ghavamzadeh and Y. Engel. Bayesian policy gradient algorithms. In *Advances in Neural Information Processing Systems 19*. MIT Press, 2007.

J. Ghosh and R. Ramamoorthi. *Bayesian Nonparametrics*. Springer Series in Statistics. Springer-Verlag, 2003.

J. Gittins, K. Glazebrook, and R. Weber. *Multi-Armed Bandit Allocation Indices*. John Wiley & Sons, 2011.

F. Gomez and R. Miikkulainen. Active guidance for a finless rocket using neuroevolution. *Genetic and Evolutionary Computation - GECCO 2003*, pages 213–213, 2003.

F. Gomez, J. Schmidhuber, and R. Miikkulainen. Accelerated neural evolution through cooperatively coevolved synapses. *Journal of Machine Learning Research*, 9:937–965, 2009.

D. Grollman. *Sparse Online Gaussian Process C++ Library*, 2012. Software http://cs.brown.edu/people/dang/code.shtml.

P. Hennig. Optimal reinforcement learning for Gaussian systems. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 325–333, 2011.

J. Huang, A. Gretton, B. Schölkopf, A. J. Smola, and K. M. Borgwardt. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2007.

H. Jakab and L. Csató. Reinforcement learning with guided policy search using Gaussian processes. In *International Joint Conference on Neural Networks (IJCNN)*, Brisbane, QLD, June 10-15 2012.

H. Jakab, B. A. Bócsi, and L. Csató. Non-parametric value function approximation in robotics. In H. F. Pop, editor, *MACS2010: The 8th Joint Conference on Mathematics and Computer Science*, volume Selected Papers, pages 235–248, Komarno, Slovakia, 2011. Györ-:NOVADAT.

E. Jaynes. *Probability Theory: The Logic of Science*. Cambridge University Press, 2003.

M. Johnson. PCFG models of linguistic tree representations. *Computatoinal Linguistics*, 24: 613–632, 1998.

M. I. Jordan and D. E. Rumelhart. Forward models: Supervised learning with a distal teacher. *Cognitive Science*, 16:307–354, 1992.

S. Kakade. A natural policy gradient. In *Neural Information Processing Systems*, pages 1531–1538, 2001.

M. Kalakrishnan, J. Buchli, P. Pastor, and S. Schaal. Learning locomotion over rough terrain using terrain templates. In *intelligent robots and systems, 2009. iros 2009. ieee/rsj international conference on*, pages 167–172, 2009.

R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

R. E. Kass and A. E. Raftery. Bayes factors. *Journal of the American Statistical Association*, 90 (430):773–795, 1995.

O. Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *Robotics and Automation, IEEE Journal of*, 3(1):43–53, 1987.

Z. Kira. Transferring embodied concepts between perceptually heterogeneous robots. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, pages 4650–4656, 2009.

J. Kober, B. J. Mohler, and J. Peters. Imitation and reinforcement learning for motor primitives with perceptual coupling. In *From Motor Learning to Interaction Learning in Robots*, pages 209–225. Springer, 2010.

J. R. Koza and J. P. Rice. Automatic programming of robots using genetic programming. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 194–201. The MIT Press, 1992.

S. Kullback. *Information Theory and Statistics*. Wiley, New York, 1959.

C. H. Lampert and M. B. Blaschko. Structured prediction by joint kernel support estimation. *Machine Learning*, 77:249–269, 2009.

P. Laskov, C. Gehl, S. Krüger, and K.-R. Müller. Incremental support vector learning: Analysis, implementation and applications. *J. Mach. Learn. Res.*, 7:1909–1936, 2006.

N. D. Lawrence, M. Seeger, and R. Herbrich. Fast sparse Gaussian process methods: The informative vector machine. In *Neural Information Processing Systems*, pages 609–616. MIT Press, 2002.

M. Lázaro-Gredilla, J. Quiñonero Candela, C. E. Rasmussen, and A. R. Figueiras-Vidal. Sparse spectrum Gaussian process regression. *Journal of Machine Learning Research*, 99: 1865–1881, 2010.

Y. Lecun, S. Chopra, R. Hadsell, F. J. Huang, G. Bakir, T. Hofman, B. Schoelkopf, A. Smola, and B. T. (eds). A tutorial on energy-based learning. In *Predicting Structured Data*. MIT Press, 2006.

J. Lee and M. Verleysen. *Nonlinear Dimensionality Reduction*. Springer, 2007.

D. Liberzon. *Calculus of Variations and Optimal Control Theory: a Concise Introduction*. Princeton University Press, 2012.

A. Liegeois. Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Transactions on Systems, Man, and Cybernetics*, 7(12):842–868, 1977.

D. J. C. Mackay. Bayesian methods for backpropagation networks. In E. Domany, J. L. van Hemmen, and K. Schulten, editors, *Models of Neural Networks III*, chapter 6, pages 211–254. Springer, 1994.

C. D. Manning and H. Schütze. *Foundations of statistical natural language processing*. MIT Press, Cambridge, MA, USA, 1999.

Z. Marx, M. T. Rosenstein, L. P. Kaelbling, and T. G. Dietterich. Transfer learning with an ensemble of background tasks. In *Advances in Neural Information Processing Systems*. MIT Press, 2005.

A. McCallum and C. Sutton. An introduction to conditional random fields for relational learning. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning*. MIT Press, 2006.

F. S. Melo and M. I. Ribeiro. Q-learning with linear function approximation. In *Proceedings of the 20th Annual Conference on Learning Theory*, pages 308–322. Springer-Verlag, 2007.

D. E. Moriarty, A. C. Schultz, and J. J. Grefenstette. Evolutionary algorithms for reinforcement learning. *Journal of Artificial Intelligence Research*, 11:241–276, 1999.

J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal. Operational Space Control: A Theoretical and Empirical Comparison. *Int. J. Rob. Res.*, 27(6):737–757, 2008.

R. Neal. Regression and classification using Gaussian process priors (with discussion). *Bayesian Statistics*, 6:475–501, 1999.

K. Neumann, M. Rolf, J. J. Steil, and M. Gienger. Learning inverse kinematics for pose-constraint bi-manual movements. In *Proceedings of the 11th international conference on Simulation of adaptive behavior: from animals to animats*, SAB'10, pages 478–488, 2010.

D. Nguyen-Tuong and J. Peters. Using model knowledge for learning inverse dynamics. In

Proceedings of the IEEE International Conference on Robotics and Automation, pages 2677–2682, 2010.

D. Nguyen-Tuong, M. W. Seeger, and J. Peters. Model learning with local Gaussian process regression. *Advanced Robotics*, 23(15):2015–2034, 2009a.

D. Nguyen-Tuong, M. W. Seeger, and J. Peters. Model learning with local Gaussian process regression. *Advanced Robotics*, 23(15):2015–2034, 2009b.

D. Nguyen-Tuong, M. W. Seeger, and J. Peters. Real-time local GP model learning. In *From Motor Learning to Interaction Learning in Robots*, pages 193–207. Springer, 2010.

S. Nolfi and D. Parisi. Learning to adapt to changing environments in evolving neural networks. In *Adaptive Behavior*, pages 75–98, 1997.

M. Opper. *A Bayesian approach to on-line learning*, pages 363–378. Cambridge University Press, 1998.

E. Oyama and S. Tachi. Modular neural net system for inverse kinematics learning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3239 – 3246, 2000.

E. Oyama, N. Y. Chong, A. Agah, T. Maeda, and S. Tachi. Inverse kinematics learning by modular architecture neural networks with performance prediction networks. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1006–1012, 2001.

S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

S. J. Pan, J. T. Kwok, and Q. Yang. Transfer learning via dimensionality reduction. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, 2008.

E. Parzen. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.

J. Peters and S. Schaal. Learning to control in operational space. *International Journal of Robotics Research*, 27(2):197–212, 2008a.

J. Peters and S. Schaal. Reinforcement learning of motor skills with policy gradients. *Neural Networks*, 21(4):682–697, 2008b.

J. Peters, S. Vijayakumar, and S. Schaal. Reinforcement Learning for Humanoid Robotics. In *Conference on Humanoid Robots*, 2003.

F. M. Phelps and J. H. Hunter. An analytical solution of the inverted pendulum. *American Journal of Physics*, 33, Issue 4:285, 1965.

F. Pourboghrat. Neural networks for learning inverse-kinematics of redundant manipulators. In *Proceedings of the 32nd Midwest Symposium on Circuits and Systems*, volume 2, pages 760–762, 1989.

M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley-Interscience, 1994.

J. Quiñonero Candela and C. E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.

R. Raina, A. Y. Ng, and D. Koller. Constructing informative priors using transfer learning. In *Proceedings of the 23rd international conference on Machine learning*, ICML '06, pages 713–720, New York, NY, USA, 2006. ACM.

A. Ranganathan, M.-H. Yang, and J. Ho. Online sparse Gaussian process regression and its applications. *IEEE Transactions on Image Processing*, 20(2):391–404, 2011.

C. E. Rasmussen and M. Kuss. Gaussian processes in reinforcement learning. In *Advances in Neural Information Processing Systems (NIPS)*, pages 751–759. MIT Press, 2004.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

R. F. Reinhart and J. J. Steil. Recurrent neural associative learning of forward and inverse kinematics for movement generation of the redundant pa-10 robot. In *Proceedings of the 2008 ECSIS Symposium on Learning and Adaptive Behaviors for Robotic Systems*, LAB-RS '08, pages 35–40, Washington, DC, USA, 2008. IEEE Computer Society.

R. F. Reinhart and J. J. Steil. Reaching movement generation with a recurrent neural network based on learning inverse kinematics for the humanoid robot icub. In *IEEE CONF. HUMANOID ROBOTICS*, 2009.

C. P. Robert and G. Casella. *Monte Carlo Methods*. Springer, second edition, 2004.

M. Rolf, J. J. Steil, and M. Gienger. Goal babbling permits direct learning of inverse kinematics. *IEEE Trans. Autonomous Mental Development*, 2(3):216 – 229, 2010.

S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Ed., 2003.

C. Salaun, V. Padois, and O. Sigaud. Learning forward models for the operational space control of redundant robots. In O. Sigaud and J. Peters, editors, *From Motor Learning to Interaction Learning in Robots*, volume 264, pages 169–192. Springer, 2010.

A. L. Samuel. Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 3(3):210–229, 1959.

S. Schaal, D. Sternad, and C. G. Atkeson. One-handed juggling: A dynamical approach to a rhythmic movement task. *Journal of Motor Behavior*, pages 165–183, 1996.

B. Schlkopf, A. J. Smola, and K. R. Müller. Kernel principal component analysis. *Advances in kernel methods: support vector learning*, pages 327–352, 1999.

B. Schölkopf and A. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT-Press, Cambridge, MA, 2002.

B. Schölkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computations*, 13:1443–1471, 2001.

L. Sciavicco and B. Siciliano. *Modelling and Control of Robot Manipulators (Advanced Textbooks in Control and Signal Processing)*. Advanced textbooks in control and signal processing. Springer, 2nd edition, 2005.

J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY, USA, 2004.

B. Siciliano. Kinematic control of redundant robot manipulators: A tutorial. *Journal of Intelligent and Robotic Systems*, 3(3):201–212, 1990.

S. P. Singh. Transfer of learning by composing solutions of elemental sequential tasks. *Mach.*

Learn., 8(3-4):323–339, 1992.

B. Skinner. *About Behaviorism*. Knopf Doubleday Publishing Group, 2011.

V. Smidl and A. Quinn. On Bayesian principal component analysis. *Computational Statistics and Data Analysis*, 51(9):4101–4123, 2007.

A. J. Smola and P. Bartlett. Sparse greedy Gaussian process regression. In *Neural Information Processing Systems*, pages 619–625. MIT Press, 2001.

E. Snelson. Local and global sparse Gaussian process approximations. *Artificial Intelligence and Statistics*, 11, 2006.

E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems*, pages 1257–1264. MIT press, 2006.

J. A. Snyman. *Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms*, volume 97 of *Applied Optimization*. Springer-Verlag New York, 2005.

E. Sontag. *Mathematical Control Theory: Deterministic Finite Dimensional Systems*. Texts in Applied Mathematics. Springer, 1998.

M. Spong and M. Vidyasagar. *Robot Dynamics And Control*. Wiley India Pvt. Ltd., 2008.

L. Steels and M. Hild, editors. *Language Grounding in Robots*. Springer, New York, 2012.

G. Sun and B. Scassellati. Reaching through learned forward model. In *Proceedings of the IEEE-RAS/RSJ International Conference on Humanoid Robots*, Santa Monica, 2004.

R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, 1998.

J. A. K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9:293–300, 1999.

B. Taskar, C. Guestrin, and D. Koller. Max-margin Markov networks. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, 2004.

M. E. Taylor and P. Stone. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(1):1633–1685, 2009.

G. Tesauro. Temporal difference learning and TD-gammon. *Commun. ACM*, 38(3):58–68, 1995.

G. Tevatia and S. Schaal. Inverse kinematics for humanoid robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 294–299, 2000.

S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents series)*. Intelligent robotics and autonomous agents. The MIT Press, 2005.

S. B. Thrun and T. M. Mitchell. Lifelong robot learning. Technical Report IAI-TR-93-7, Robotics and Autonomous Systems, 1993.

M. E. Tipping. *Bayesian Inference: An Introduction to Principles and Practice in Machine Learning*, volume 3176, chapter 3, pages 41–62. Springer Berlin Heidelberg, 2004.

M. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *the 12th International Conference on Artificial Intelligence and Statistics*, volume 5, 2009.

L. Trefethen and D. Bau. *Numerical linear algebra*. Miscellaneous Books. Society for Industrial and Applied Mathematics, 1997.

V. Tresp. A Bayesian committee machine. *Neural Comput.*, 12:2719–2741, 2000.

I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6:1453–1484, 2005.

V. Vapnik. *The Nature of Statistical Learning Theory*. Statistics for Engineering and Information Science. Springer, 1999.

S. Vijayakumar, A. D'Souza, and S. Schaal. Incremental online learning in high dimensions. *Neural Computation*, 17:2602–2634, 2005.

A. von Twickel, M. Hild, T. Siedel, V. Patel, and F. Pasemann. Neural control of a modular multi-legged walking machine: Simulation and hardware. *Robotics and Autonomous Systems*, 60(2):227 – 241, 2012.

G. Wahba. *Spline Models for Observational Data*. CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial and Applied Mathematics, 1990.

C. Wang and S. Mahadevan. Manifold alignment using Procrustes analysis. In *Proceedings of the 25th international conference on Machine learning*, pages 1120–1127. ACM New York, NY, USA, 2008.

J. Weston, O. Chapelle, A. Elisseeff, B. Schölkopf, and V. Vapnik. Kernel dependency estimation. In *Neural Information Processing Systems*, pages 873–880, 2002.

C. Williams and M. Seeger. Using the nyström method to speed up kernel machines. In *Neural Information Processing Systems*, pages 682–688. MIT Press, 2001.

R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. In *Machine Learning*, pages 229–256, 1992.