

UNIVERZA NA PRIMORSKEM  
FAKULTETA ZA MATEMATIKO, NARAVOSLOVJE IN  
INFORMACIJSKE TEHNOLOGIJE

Zaključna naloga  
**Strojno učenje iz interakcije**  
Machine learning from interaction

Ime in priimek: Rok Breulj

Študijski program: Računalništvo in informatika

Mentor: doc. dr. Peter Rogelj

Koper, Avgust 2014

## Ključna dokumentacijska informacija

Ime in PRIIMEK:

Naslov zaključne naloge:

Kraj:

Leto:

Število listov:

Število slik:

Število tabel:

Število prilog:

Število strani prilog:

Število referenc:

Mentor:

Somentor:

Ključne besede:

Math. Subj. Class. (2010):

### **Izvleček:**

Izvleček predstavlja kratek, a jedrnat prikaz vsebine naloge. V največ 250 besedah nakažemo problem, metode, rezultate, ključne ugotovitve in njihov pomen.

## Key words documentation

Name and SURNAME:

Title of final project paper:

Place:

Year:

Number of pages:

Number of figures:

Number of tables:

Number of appendices:

Number of appendix pages:

Number of references:

Mentor:

Co-Mentor:

Keywords:

Math. Subj. Class. (2010):

**Abstract:**

## Zahvala

# Kazalo vsebine

<b>Kazalo tabel</b>	<b>vi</b>
<b>Kazalo slik</b>	<b>vii</b>
<b>Seznam kratic</b>	<b>viii</b>
<b>1 Uvod</b>	<b>1</b>
1.1 Psihologija . . . . .	3
1.1.1 Klasično pogojevanje . . . . .	3
1.1.2 Operantno pogojevanje . . . . .	4
<b>2 Problem okrepitvenega učenja</b>	<b>8</b>
2.1 Elementi okrepitvenega učenja . . . . .	8
2.2 Končen Markov proces odločanja . . . . .	11
2.3 Diskretno okrepitveno učenje . . . . .	12
2.4 Raziskovanje in izkoriščanje . . . . .	13
2.4.1 $\epsilon$ -požrešna izbira dejanj . . . . .	14
<b>3 Tabularne rešitve</b>	<b>15</b>
3.1 Dinamično programiranje . . . . .	15
3.2 Monte Carlo metode . . . . .	16
3.3 Učenje na podlagi časovne razlike - TD(0) . . . . .	16
3.4 Združitev metod - TD( $\lambda$ ) . . . . .	16
<b>4 Posploševanje in funkcijska aproksimacija</b>	<b>17</b>
4.1 Umetne nevronske mreže . . . . .	17
4.2 Metode gradient descent . . . . .	17
<b>5 Namizna igra Hex</b>	<b>18</b>
5.1 Ozadje . . . . .	18
5.2 Učenje . . . . .	18
5.3 Rezultati . . . . .	18

<b>6 Zaključek</b>	<b>19</b>
--------------------	-----------

<b>Literatura</b>	<b>21</b>
-------------------	-----------

# Kazalo tabel

1.1	Vpliv pozitivne in negativne okrepite in kaznovanja na vedenje. . . . .	6
-----	---	---

# Kazalo slik

1.1	Klasično pogojevanje piščalke namesto hrane za slinjenje pri psu [5]. . .	4
2.1	Interakcija med učencem in njegovim okoljem [15]. . . . .	9



# Seznam kratic

*tj.* to je

*npr.* na primer

# 1 Uvod

Učimo se skozi naše celotno življenje. Eden izmed osnovnih načinov učenja temelji na podlagi interakcije z okoljem. V računalništvu pogosto radi odidemo po tej eksperimentalni poti, posebej ko verjamemo, da smo blizu rešitvi. Vendar pa se ni potrebno zazreti tako daleč, kot je računalništvo. Že kot otroci, ko mahamo z rokami in gledamo naokoli, nimamo izrecnega učitelja, imamo pa neposredno senzomotorično povezavo z našo okolico. S svojim vedenjem vplivamo na okolje in naša čutila izkoriščamo za pridobitev ogromne količine podatkov o vzrokih in učinkih, o posledicah dejanj in načinih, kako doseči cilje. Skozi naše življenje predstavljajo tovrstne interakcije velik vir znanja o našem okolju in o nas samih. Ko se učimo voziti avto ali pogovarjati, se zavedamo kako se okolje odziva na naša dejanja in iščemo način kako vplivati na rezultat z našim vedenjem.

Čeprav ni ene same standardne definicije inteligence, lahko različne predlagane definicije med seboj primerjamo in hitro najdemo precejšnje podobnosti. V veliko primerih, definicije inteligence vsebuje idejo, da se mora posameznik, ki je inteligen, znati prilagoditi okoljem, s katerimi se poprej še nikoli ni srečal, in v njih doseči cilje [21]. Za inteligentno vedenje očitno torej potrebujemo način, da ovrednotimo in razvrstimo nove položaje. Da se posameznik lahko uči in prilagaja svoje vedenje, mora znati upoštevati tudi informacije iz okolja in iz njih sklepati. Okrepiteveno učenje (angl. reinforcement learning) predstavlja teorijo o učenju povečanja nagrade na voljo v okolici in tako neposredno povečanje možnosti prilagoditve in preživetja. Nekatere naloge so preveč zapletene, da bi se jih opisalo v statičnem računalniškem programu, kar je danes pogost postopek. Način za dinamično učenje in razvijanje programa je pri nekaterih nalogah torej potreben.

Praktično vse aktivnosti živali, podjetij in računalniških programov vključujejo niz dejanj za doseg cilja. Tako pri vožnji z avtomobilom na delo kot tudi med pripravo jutranje kave obstaja cilj in zaporedje dejanj za uspešno opravljen cilj. Prilagodljiv krmilni sistem, ki se zna učiti izvajati takšne sekvenčne naloge odločanja lahko najde vlogo v številnih domenah, kot so krmiljenje proizvodnega procesa, avtonomnih vozilih, letalstvu in pripomočkih za invalide. V pametnih sistemih, ki delujejo v dinamičnih okoljih resničnega sveta, kjer se ni mogoče zanašati na obvladljive pogoje in kjer vladata negotovost ter časovne omejitve, ima lahko odločanje na podlagi okrepitevnega učenja

bistvene prednosti pred ostalimi vrstami učenja.

Področje okrepitevenega učenja sega v zelo različne discipline in je močno povezano s teorijo krmiljenja (angl. control theory), psihologijo in nevroznanostjo. Teorija krmiljenja pripomore k reševanju problema z analitičnega oziroma matematičnega vidika, medtem ko se psihologija in nevroznanost za odgovore zgledujeta po bioloških procesih. Veliko temeljnih smernic je izpeljanih iz psihologije vedenja in učenja; teorijah, ki se tičejo nagrajevanja in pogojevanja dejanj. Algoritmični pristopi so izpeljani pod podobnimi načeli kot ljudje in živali oblikujemo vedenja glede na odzive iz okolice.

Zamisel o gradnji inteligentni strojev sega v daljno preteklost; o tem so razmišljali že Egipčani. Čez leta se je razvilo veliko teorij, ampak šele z začetkom sodobnega računalnika pred 60-imi leti sta se umetna inteligenca in strojno učenje razvila v samostojno znanstveno področje [2]. Leta 1948 je Claude Elwood Shannon [4] napisal predlog za šahovski program, Arthur Samuel [1] pa je leta 1959 razvil računalniški program, ki se je naučil igrati namizno igro dama z igranjem proti samemu sebi. V zadnjih letih so se raziskave osredotočale bolj na posnemanje bioloških modelov, da bi izdelale programe, ki rešujejo probleme in razmišljajo kot ljudje. Nevrološke mreže (angl. neural networks), pri katerih gre za zelo poenostavljen model možganov, so bile uspešno uporabljene v vrsti aplikacijah. Po formalizaciji Samuelevega pristopa in oblikovanja učenja na podlagi časovne razlike  $\lambda$  Richarda Suttona [14] je Richard Tesauro [10] leta 1992 razvil računalniškega igralca za igro Backgammon, ki je tekmoval proti najboljšim človeškim igralcem na svetu. Čeprav je Tesaurova združitev pristopa okrepitevenega učenja in nevroloških mrež pretresla področje umetne inteligence in Backgammonske skupnosti, ni bilo veliko drugih uspehov v namiznih igrah [9, 17, 18]. Prenos Tesaurove rešitve v največje namizne igre na področju umetne inteligence – šah in Go – niso uspele; rezultati so bili slabši, kot pa so jih dosegale konvencionalne metode. Poleg namiznih iger se je okrepiteveno učenje uporabljalo tudi v problemih robotike, razporejanja, dinamičnih dodelitev in optimizacije [15].

V nadaljevanju naloge je v razdelku 1.1 pregledan izvor okrepitevenega učenja iz vedenjske plati. Temu sledi pogled s stališča umetne inteligence in inženirstva, razišče pa se tudi računski pristop do učenja iz interakcije. V razdelku ?? je matematično opredeljen celotni problem okrepitevenega učenja, rešitve pa so predstavljene v razdelku ??. Primerjajo se različni algoritmi, njihove povezave in kombinacije. Ker so cilji in pravila pri abstraktnih namiznih igrah jasno opredeljeni, s čimer se poenostavi model in simulacija, in poleg tega predstavljajo zahteven in hkrati zanimiv problem, so priročno testno okolje za študijo tovrstnega učenja. V razdelku ?? so rešitve iz okrepitevenega učenja uporabljene v namizni igri Hex in na koncu v razdelku ?? razpravljeni rezultati skupaj s pogledom na prihodnost.

## 1.1 Psihologija

Okrepiteveno učenje ima korenine v psihologiji učenja živali, iz kjer izvira tudi samo ime. Posebej se nanaša na klasično pogojevanje (angl. classical conditioning) in operantno pogojevanje (angl. operant conditioning).

### 1.1.1 Klasično pogojevanje

Klasično pogojevanje (imenovana tudi Pavlovo pogojevanje) je učenje prek povezav oz. asociacij.

V začetku 20. stoletja je ruski psiholog Ivan Pavlov (1849-1936) med preučevanjem prebavnega sistema psov odkril vedenjski fenomen [11]: psi so se začeli sliniti takoj, ko so laboratorijski tehniki, ki so jih hranili, vstopili v sobo, čeprav psi še niso dobili hrane. Pavlov je spoznal, da so se psi začeli sliniti, ker so vedeli, da bodo dobili hrano; povezali so prihod tehnikov s hranjenjem.

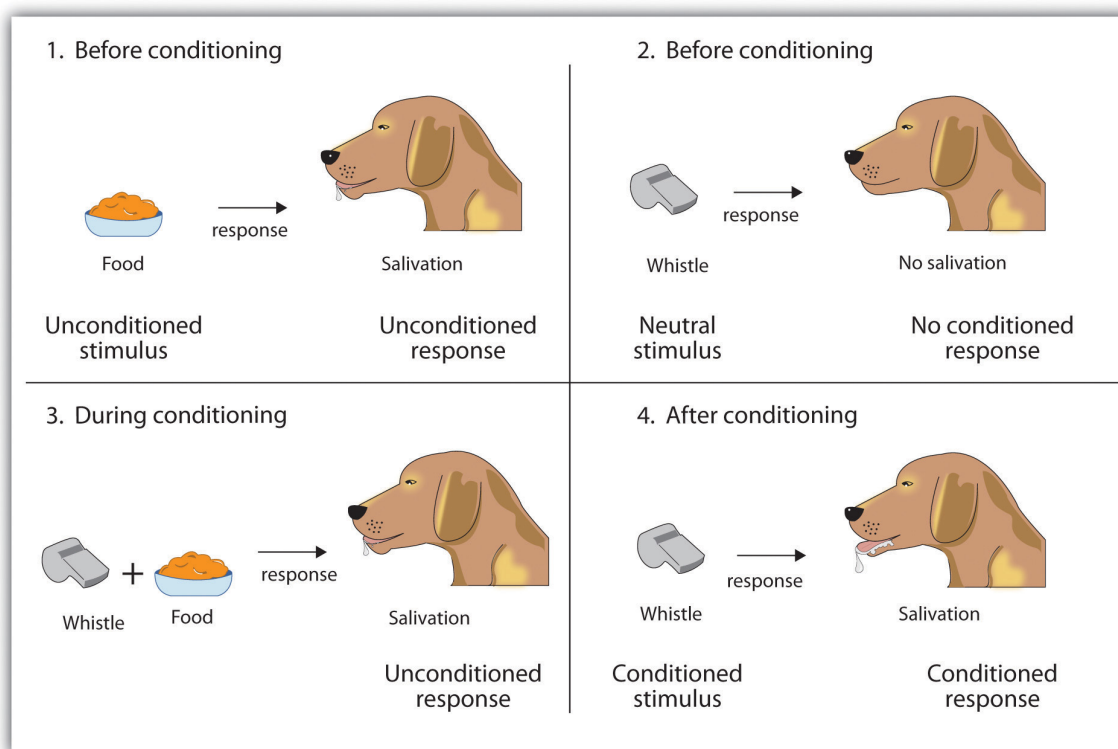
S svojo ekipo je začel proces raziskovati bolj podrobno. Opravi je vrsto eksperimentov, pri katerih so bili psi izpostavljeni zvoku, tik preden so dobili hrano. Sistematično je nadzoroval časovno razliko med pojavom zvoka in hrano ter zabeležil količino sline pri pseh. Najprej so se psi slinili samo, ko so hrano videli ali zavohali. Po večkrat predstavljenem zvoku skupaj s hrano pa so se psi začeli sliniti takoj, ko so zaslišali zvok. Zvok so se namreč naučili povezati s hrano, ki mu je sledila.

Pavlov je odkril temeljni asociativni proces imenovan klasično pogojevanje. Gre za učenje, pri katerem postane nevtralna spodbuda (na primer: zvok) povezana s spodbudo, ki vedenje sproži sama po sebi (na primer: hrana). Ko se povezava enkrat nauči, poprej nevtralna spodbuda zadošča za pojav vedenja, ki je v večji meri enakovredno (Pavlov je opazil razliko v sestavi sline [12, 16, 20]).

Prihod tehnikov oz. zvok je Pavlov imenoval pogojena spodbuda (angl. conditioned stimulus CS), ker je njen učinek odvisen od povezave s hrano. Hrano je imenoval nepogojena spodbuda (angl. unconditioned stimulus US), ker njen učinek ni odvisen od predhodnih izkušenj. Podobno gre pri pogojenem odzivu (angl. conditioned response CR) za odziv pogojene spodbude CS in pri nepogojenem odzivu (angl. unconditioned response UR) za odziv nepogojene spodbude US. Pavlov je odkril, da je krajši razmak med zvokom in prikazom hrane povzročil močnejše in hitrejše učenje pogojenega odziva CR psa [23].

Pogojevanje je evolucijsko koristno, ker omogoča organizmom razviti pričakovanja, ki jim pomagajo v dobrih in slabih okoliščinah. Razvidno je na primeru živali, ki zavoha novo hrano, jo poje in posledično zbolijo. Če se žival zna naučiti povezave vonja (CS) s hrano (US), se bo znala izogibati določeni hrani že po vonju.

Klasično pogojevanje obravnava samo problem napovedovanja, ker odziv živali ne



Slika 1.1: Klasično pogojevanje piščalke namesto hrane za slinjenje pri psu [5].

vpliva na eksperiment, oziroma na splošno ne vpliva na okolje. Učenje na podlagi časovne razlike (angl. temporal difference learning), ki je opisano pozneje v razdelku 3.3, je bilo prvotno povezano predvsem s klasičnim pogojevanjem in problemom napovedovanja, kjer pogojena spodbuda (CS), ki je vezana na poznejšo nepogojeno spodbudo (US), povzroči potrebo po ovrednotenju časovne razlike vrednostne funkcije. Cilj izračuna je zagotoviti, da postane pogojena spodbuda (CS) po učenju napovednik nepogojene spodbude (US). Osnutek na temo algoritmičnih pristopov do eksperimentov klasičnega pogojevanja sta sestavila Belkenius in Morén [3].

Čeprav je bilo učenje na podlagi časovne razlike sprva namenjeno reševanju problema napovedovanja, se uporablja tudi za reševanje problema optimalnega krmiljenja (glej razdelek ??) [15].

### 1.1.2 Operantno pogojevanje

V klasičnem pogojevanju se organizem nauči povezati nove spodbude z naravnim biološkim odzivom, kot je slinjenje ali strah. Organizem sam se ne nauči ničesar novega, ampak se v prisotnosti novega signala začne vesti na že obstoječi način. Po drugi strani pa gre pri operantnem pogojevanju za učenje, ki se zgodi glede na posledice vedenja in lahko vsebuje nova dejanja. Med operantno pogojevanje sodi primer, ko se pes na

ukaz vsede, ker je v preteklosti za to dejanje dobil pohvalo. Za operantno pogojevanje gre tudi, ko nasilnež v šoli grozi sošolcem, ker lahko tako doseže svoje cilje, ali ko otrok domov prinese dobre ocene, ker so mu starši v nasprotnem primeru zagrozili s kaznijo. Pri operantnem pogojevanju se organizem uči iz posledic svojih dejanj.

Psiholog Edward L. Thorndike (1874-1949) je bil prvi, ki je sistematično preučil operantno pogojevanje. Izdelal je škatlo, katero je bilo mogoče odpreti samo po rešitvi preproste uganke. Vanjo je spustil mačko in opazoval njeno vedenje. Sprva so mačke praskale in grizle naključno, sčasoma pa so slučajno potisnile na ročico in odprle vrata, za katerimi je stala nagrada – ostanki ribe. Ko je bila mačka naslednjič zaprta v škatlo, je poizkusila manjše število neučinkovitih dejanj, preden se je uspešno osvobodila. Po več poizkusih se je mačka naučila skoraj takoj pravilno odzvati. [7]

Opazovanje tovrstnih sprememb v mačjem vedenju je Thorndikeju pomagalo razviti njegov zakon o učinku, pri katerem gre za princip, da se odzivi, ki v določeni situaciji navadno pripeljejo do prijetnega izida, bolj verjetno pojavijo ponovno v podobni situaciji, medtem ko je za odzive, ki tipično pripeljejo do neprijetnega izida, manj verjetno, da se ponovno pojavijo v tej situaciji. [8]

Vedenjski psiholog B. F. Skinner (1904-1990) je omenjene ideje razširil in jih povezal v bolj dovršen sistem, ki opredeljuje operantno pogojevanje. Zasnoval je operantne komore (tako imenovane Skinner škatle) za sistemsko preučevanje učenja, pri katerih gre za zaprto strukturoz dovolj prostora za manjšo žival, v kateri je slednja s pritiskom na palico ali gumb prišla do nagrade v obliki vode ali hrane. Struktura je poleg tega vsebovala tudi napravo za grafični zapis odzivov živali. [5]

Najosnovnejši eksperiment je Skinner izvedel na način, ki je zelo podoben Thorndikejevem poizkusu z mačkami. V škatlo je spustil podgano, katera se je sprva odzvala po pričakovanjih - hitela je naokrog, vohljala in praskala po tleh ter stenah. Čez čas je slučajno naletela na gumb in ga pritisnila ter s tem prišla do koščka hrane. Naslednjič je že potrebovala manj časa in z vsakim novim poizkusom je hitreje pritisnila na gumb. Kmalu je pritiskala na gumb, čim je lahko jedla hrano. Kot pravi zakon o učinku, se je podgana naučila ponavljati dejanje, ki ji je pomagalo priti do hrane, in prenehala z dejanjem, ki so se bila izkazala za neuspešna. [5]

Skinner je preučeval, kako živali spreminjajo svoje vedenje v odvisnosti od okrepitve (angl. reinforcement) in kaznovanja (angl. punishment). Določil je izraze, ki razlagajo proces operantnega učenja (glej tabelo 1.1). Okrepitev je opredelil kot dogodek, ki utrdi ali zviša verjetnost nekega vedenja, kaznovanje pa je označil za dogodek, ki oslabi ali zniža verjetnost nekega vedenja. Uporabil je še izraza pozitivno in negativno za opredeliti, če je spodbuda predstavljena ali odvzeta. Pozitivna okrepitev torej utrdi odziv s predstavitvijo nečesa prijetnega in negativna okrepitev utrdi odziv z znižanjem ali odvzemom nečesa neprijetnega. Pohvala otroka za opravljeno domačo

nalogo tako na primer sodi v pozitivno okrepitev, medtem ko predstavlja jemanje aspirina za zniževanje glavobola negativno okrepitev. V obeh primerih okrepitev zviša verjetnost, da se vedenje v prihodnosti ponovi. [5]

Izraz	Opis	Izid	Primer
Pozitivna okrepitev	Predstavljena ali povečana prijetna spodbuda	Vedenje je utrjeno	Otrok dobi slaščico, potem ko pospravi sobo
Negativna okrepitev	Zmanjšana ali odvzeta neprijetna spodbuda	Vedenje je utrjeno	Starši se prenehajo pritoževati, potem ko otrok pospravi sobo
Pozitivno kaznovanje	Predstavljena ali povečana neprijetna spodbuda	Vedenje je oslabiljeno	Učenec dobi dodatno domačo nalogo, potem ko nagaja v razredu
Negativno kaznovanje	Zmanjšana ali odvzeta prijetna spodbuda	Vedenje je oslabiljeno	Otrok izgubi privilegij računalnika, potem ko pride pozno domov

Tabela 1.1: Vpliv pozitivne in negativne okrepitve in kaznovanja na vedenje.

Čeprav je razlika med okrepitvijo (povišanje verjetnosti vedenja) in kaznovanjem (znižanje verjetnosti vedenja) navadno jasna, je v nekaterih primerih težko določiti, ali gre za pozitivno ali negativno. V vročem poletnem dnevu je lahko svež veter zaznan kot pozitivna okrepitev (ker prinese hladnejši zrak) ali pa negativna okrepitev (ker odvzame vroč zrak). V nekaterih primerih je lahko okrepitev hkrati pozitivna in negativna. Za odvisnika, jemanje drog hkrati prinese užitek (pozitivna okrepitev) in odstrani neprijetne simptome umika (negativna okrepitev). [5]

Pomembno se je tudi zavedati, da okrepitev in kaznovanje nista zgolj nasprotna pojma. Spreminjanje vedenja s pomočjo pozitivne okrepitve je skoraj vedno učinkovite od kaznovanja, ker pozitivna okrepitev pri osebi ali živali izboljša razpoloženje in pripomore k vzpostavitvi pozitivnega razmerja z osebo, ki predstavlja okrepitev. Med vrste pozitivne okrepitve, ki so učinkovite v vsakdanjem življenju, sodijo izrečene pohvale in odobritve, podelitev statusa in prestiža ter neposredno denarno izplačilo. Pri kaznovanju pa je po drugi strani bolj verjetno, da bodo spremembe vedenja samo začasne, ker temelji na prisili in vzpostavi negativno ter nasprotujoče razmerje z osebo, ki predstavlja kazen. Ko se oseba, ki kazen predstavi, iz okolja umakne, se neželeno vedenje najverjetneje vrne. [5]

Operantno pogojevanje je metoda učenja, ki se uporablja pri treniranju živali za izvajanje različnih trikov. V filmih in na predstavah so živali, od psov do konjev in

delfinov, naučene dejanj z uporabo pozitivnih okrepitev – skačejo čez ovire, se vrtijo, pomagajo osebi pri vsakdanjih opravilih in izvajajo podobna zanje neobičajna dejanja. [5]

Velikokrat se pri učenju hkrati izvajata klasično in operantno pogojevanje. Učitelji imajo s seboj napravo, ki proizvede določen zvok. Učenje se začne z nagrajevanjem zelenega enostavnega dejanja s hrano (operantno pogojevanje) in hkrati z vzpostavljanjem povezave med hrano in zvokom (klasično pogojevanje). Hrana se tako lahko po intervalih izpusti in pred dejanjem se doda še zvočni ukaz, na katerega želimo vezati učeno dejanje (klasično pogojevanje). Tako z nagrado hrane povežemo samo zvočni ukaz, ki je predstavljen pred dejanjem. Kompleksnejša dejanja se naučijo postopoma iz enostavnejših z nadaljno povezavo spodbud, kar se imenuje proces oblikovanja. [5]

Spodbude, ki so organizmu naravno zadovoljive, kot na primer hrana, voda in preprečevanje bolečine, se imenujejo primarne spodbude, medtem ko gre pri sekundarni spodbudi za nevtralni dogodek, ki je s primarno spodbudo povezan s pomočjo klasičnega pogojevanja. Primer sekundarne spodbude je povezava zvoka s primarno spodbudo, kakršna je hrana. Dodaten primer vsakdanje sekundarne spodbude je denar. Denar ni zaželen sam po sebi, temveč zaradi primarnih spodbud oziroma stvari, katere se z njim lahko kupijo. [5]

Tudi domači ljubljenci se obnašanja naučijo na podlagi teh konceptov; ne samo na ukaz, ampak tudi samega vedenja na povodcu, do tujcev itd. S to metodo se živali lahko nauči celo razlikovati med podobnimi vzorci, s čimer lahko znanstveniki na živalih preverjajo sposobnost učenja. Porter in Neuringer [6] sta golobe na primer naučila, da so razlikovali med različnimi vrstami glasbe, Watanabe, Sakamoto in Wakita [22] pa so jih naučili razlikovati med različnimi stili umetnosti.

Operantno pogojevanje se od klasičnega razlikuje v tem, da spremeni vedenje do okolja. Ne obravnava več samo problema napovedovanja, ampak širši problem krmljenja.



## 2 Problem okrepitvenega učenja

*Okrepitveno učenje* (angl. *reinforcement learning*) po [15] je učenje kaj narediti, kako izbirati dejanje, da povečamo številčni nagrajevalni signal. Učencu niso nikoli predstavljena pravilna ali optimalna dejanja kot pri večini oblik strojnega učenja. Katera dejanja prinesejo največjo nagrado mora sam odkriti s poizkušanjem. Skozi interakcijo z okoljem se uči posledic svojih dejanj. V najbolj zanimivih in težavnih primerih imajo dejanja vpliv ne le na takojšnjo nagrado ampak tudi na naslednji položaj in posledične nagrade. Te dve karakteristiki, iskanje s poizkušanjem in zamudne nagrade, so dve najpomembnejši lastnosti okrepitvenega učenja.

Okrepitveno učenje se razlikuje od nadzorovanega učenja (angl. *supervised learning*) v tem, da nima izobraženega zunanega nadzornika, ki predloži učencu primere in rezultate. Nadzorovano učenje je pomemben tip učenja vendar ni primerno za učenje iz interakcije. V interaktivnih problemih je velikokrat nepraktično pridobiti primere želenega vedenja, ki so pravilni in hkrati predstavljajo vsa stanja v katerih mora učenec delovati. V neznanem okolju, kjer bi si lahko predstavljali, da je učenje najbolj koristno, se mora učenec učiti iz svojih izkušenj.

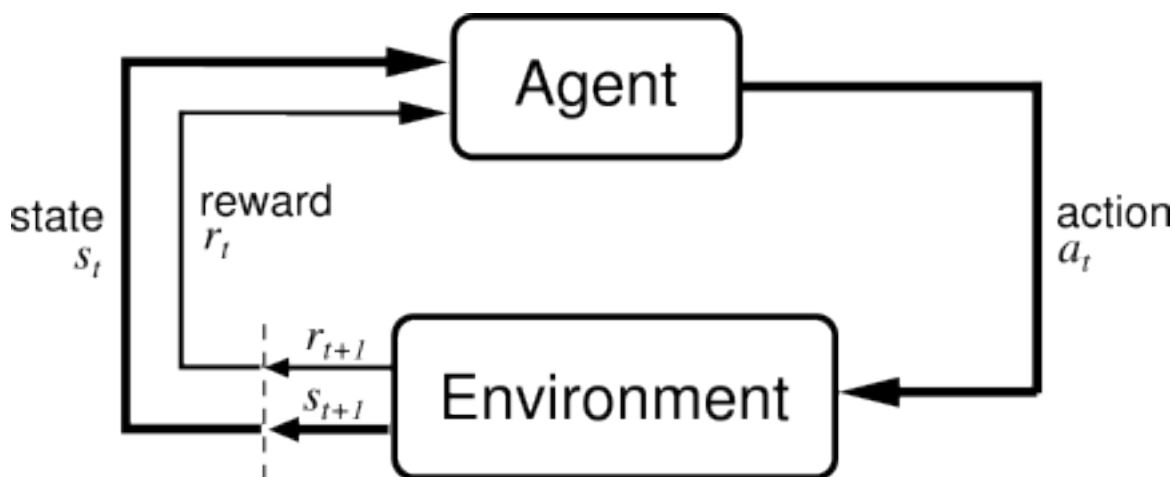
To poglavje formalno definira dele okrepitvenega učenja ter določi predpostavke potrebne za opis rešitev v sledečih poglavjih.

### 2.1 Elementi okrepitvenega učenja

Cilj algoritmov okrepitvenega učenja je optimizirati vedenje *učenca* (angl. *agent*). Učenec je tisti, ki se skozi interakcijo odloča o *dejanjih* (angl. *action*) za rešitev zadane *naloge* (angl. *task*).

Učenec se s svojimi dejanji vede na *okolje* (angl. *environment*). Vse kar učenec ni zmožen poljubno spremeniti se smatra kot izven učenca in pripada okolju. Učenec in okolje neprestano vplivata na drug drugega; učenec izbira dejanja in okolje se odziva na ta dejanja s predstavitvijo novih *stanj* (angl. *state*) učencu. Okolje, ob prehodih na nova stanja, oddaja tudi številčne *nagrade* (angl. *reward*), katere učenec poizkuša povečati skozi čas. Natančneje, učenec in okolje so v interakciji v vsakem koraku diskretnega zaporedja časa  $t = 0, 1, 2, 3, \dots$ . V vsakem koraku učenec prejme predstavitev stanja okolja,  $s_t \in S$ , kjer je  $S$  množica vseh možnih stanj. Glede na stanje se odloči za

dejanje,  $a_t \in A_s$ , iz množice možnih dejanj. En časovni korak pozneje prejme učenec, kot posledica svojega dejanja, številčno nagrado,  $r_{t+1} \in R$ , in se znajde v novem stanju,  $s_{t+1}$ . Slika 2.1 prikazuje opisan potek interakcije med učencem in okoljem. Takšna opredelitev elementov okrepitvenega učenja ustreza številnim težavam. Ni nujno, da časovni koraki predstavljajo fiksne intervale v resničnem času, lahko se nanašajo na poljubne zaporedne faze odločanja oziroma delovanja. Dejanja so lahko v zelo nizkem nivoju, na primer napetosti, ki krmilijo roko robota, ali pa odločitve v visokem nivoju, na primer v katero šolo se vpisati ali pa kakšno hrano pripraviti za večerjo. Stanja so lahko tudi v zelo različnih predstavitev, od nizko-nivojskih odčitkov senzorjev do abstraktnih simboličnih opisov sob. V splošnem so lahko dejanja katerekoli odločitve o katerih se želimo učiti in stanja karkoli, ki nam lahko pomaga pri teh odločitvah. Edini cilj učenca je povečati prejete nagrade čez čas.



Slika 2.1: Interakcija med učencem in njegovim okoljem [15].

*Politiko* (angl. *policy*)  $\pi$  imenujemo pravilo po katerem se učenec odloča katero dejanje izvesti v vsakem od stanj. Z drugimi besedami: politika preslikuje stanja v dejanja. Sama po sebi zadostuje za popolno definirano vedenje. V času  $t$  predstavlja  $\pi_t(a_t|s_t)$  verjetnost, da učenec izvede dejanje  $a_t$  v stanju  $s_t$ . V psihologiji koncept politike ustreza povezavam spodbud z odzivi. V splošnem so politike lahko stohastične.

*Nagrajevalna funkcija* (angl. *reward function*) opredeljuje cilje v nalogi okrepitvenega učenja, saj povečava nagrad čez čas predstavlja edini cilj učenca. V grobem, nagrajevalna funkcija stanja okolja preslikuje v realno število,  $r_t$ , nagrado, ki predstavlja trenutno zaželenost stanja. Pozitivne nagrade spodbujajo obiske stanj, negativne pa jih odvrčajo. Oddane nagrade predstavljajo, kako dobro se učenec vede v okolju; nagrade definirajo dobre in slabe dogodke. Uporaba nagrajevalnega signala za formalizacijo ideje cilja je eno izmed najbolj značilnih lastnosti okrepitvenega učenja. Čeprav se takšno formuliranje ciljev morda prvotno zdi omejujoče, se je v praksi izkazalo za

zelo fleksibilno in primerno. Številne nagrade so pogosto preprosto definirane kot  $+1$  ali  $-1$ . Kritično je, da nagrade točno določajo zadan cilj. Nagrajevalni signal ni primerno mesto za podeljevanje učenca s predhodnim znanjem kako doseči cilj. Pri primeru učenca, ki igra šah, je pravilno nagrajevanje samo v primeru zmage, ne pa za dosego podciljev, kot je zavzemanje nasprotnikovih figur. Če nagrajujemo takšne podcilje, se zna zgoditi, da učenec najde pot, kako doseči podcilje brez, da doseže končen cilj. V primeru šaha, bi lahko našel način zavzemanja nasprotnikovih figur tudi na račun izgubljenih igr. Nagrajevalna funkcija je način komuniciranja z robotom kaj naj doseže, ne pa kako naj to doseže. V biološkem sistemu bi lahko nagrade identificirali kot užitek ali bolečina. V psihologiji so to okrepitve ali kaznovanje. Nagrajevalna funkcija je nujno del okolja in je učenec ne sme biti zmožen spremeniti. V splošnem so nagrajevalne funkcije lahko stohastične.

Med tem ko nagrajevalna funkcija označuje kaj je dobro v takojšnjem smislu, *vrednostna funkcija* (angl. *value function*)  $V$  določa kaj je dobro na dolgi rok. Vrednostna funkcija izraža pričakovano nagrado iz obiska stanja, hkrati takojšnjo in dolgoročno; to je, izraža skupno količino nagrade, ki jo učenec lahko predvideva nabrati čez čas z začetkom v določenem stanju. Vrednosti upoštevajo stanja, ki verjetno sledijo, in nagrade iz teh stanj. Vrednostna funkcija  $V$  preslikuje stanja  $s$  v vrednosti  $v$ . Najpomembnejši del skoraj vseh algoritmov okrepitvenega učenja je učinkovito ocenjevanje vrednosti. Tudi v vsakdanjem življenju velikokrat ocenjujemo in napovedujemo dolgoročno vrednost situacij, kar nam predstavlja nemajhen izziv. Stanja, ki imajo visoko takojšnjo nagrado so lahko dolgoročno slaba in imajo nižjo vrednost kot alternativna. Slaščica, na primer, ima kratkoročen užitek, ampak velikokrat ni najboljša izbira prehrane kar se tiče našega telesnega zdravja. Obratno je tudi mogoče, stanje ima lahko zelo nizko takojšnjo nagrado, ampak se skozi prihodnost izkaže za najboljšo izbiro. Naša pot do službe je lahko hitrejša, če ne izberemo očitno najkrajšo pot, ampak upoštevamo promet do katerega nas zadana pot pripelje. Tudi živali se pri operantnem učenju učijo vrednosti in ne le takojšnjih nagrad. Hitro se naučijo, da dolgoročno prejmejo nagrade hitreje, če se pravilno vedejo, kot pa če ne. Nagrade so primarne med tem, ko so vrednosti sekundarne. Brez nagrad ne bi bilo vrednosti, ampak vedemo se glede na ocene vrednosti. Razlika je tudi v tem, da so nagrade oddane iz okolja, vrednosti pa moramo neprestano ocenjevati učenci sami. Vrednostna funkcija določa politiko vedenja, saj s svojim vedenjem želimo povečati nagrade katere vrednostna funkcija opisuje.

*Vrednostna funkcija dejanj* (angl. *action-value function*)  $Q$  je enakovredna vrednostni funkciji, s to razliko, da stanja  $s$  slika direktno v dejanja  $a$ . Vrednostna funkcija  $V$  določa vrednost  $v$  stanju  $s$  ( $V(s) = v$ ) med tem, ko vrednostna funkcija dejanj določa vrednost  $v$  dejanju  $a$  ( $Q(a) = v$ ).

Nekateri algoritmi znajo uporabijo tudi model okolja, pri drugih je pa opuščen.

## 2.2 Končen Markov proces odločanja

V okrepitevnem učenju se učenec vede na podlagi signala iz okolja, ki ga imenujemo tudi stanje okolja. V tem razdelku je opisano kaj je zahtevano od signala stanja in kakšne informacije je smiselno od signala pričakovati.

Po eni strani lahko signal stanja pove veliko več, kot samo trenutne meritve. Stanja so lahko predstavljena z močno obdelanimi originalnimi meritvami, ali pa s kompleksnimi strukturami, ki so zgrajena skozi čas. Če slišimo odgovor “da” se znajdemo v zelo različnih stanjih odvisno od predhodnega vprašanja, ki ga ne slišimo več.

Po drugi strani pa ne smemo predpostavljati, da nam signal stanja zna povedati vse o okolju, ali celo vse kar nam bi prišlo prav za odločanje. Če igramo igro s kartami ne smemo predvidevati, da bomo izvedeli kaj imajo drugi igralci v rokah ali pa katera je naslednja karta na vrhu kupa. Če učenec odgovori na telefon, ne smemo predpostavljati, da ve kdo ga kliče vnaprej. V obeh primerih obstajajo skrite informacije stanja, ki jih učenec ne more vedeti, ker jih ni nikoli prejel.

Kar bi radi, idealno, je signal stanja, ki povzame vse uporabne predhodne informacije. Za to je ponavadi potrebna več kot samo trenutna informacija, ampak nikoli več kot celotna preteklost vseh prejetih informacij. Signal stanja, ki zadrži vse uporabne predhodne informacije, pravimo, da je *Markov*, oziroma, da ima *Markovo lastnost*. Na primer, pri igri štiri v vrsto je trenutna konfiguracija vseh polj Markovo stanje, ker povzame vse kar je pomembno o poteku igre. Čeprav je veliko informacije o poteku igre izgubljeno, je vse pomembno še vedno na voljo.

Pri končnem številu stanj in nagrad, je v splošnem dinamika okolja definirana samo s popolno porazdelitvijo verjetnosti

$$Pr\{r_{t+1} = r, s_{t+1} = s' | s_0, a_0, r_1, \dots, s_{t-1}, a_{t-1}, r_t, s_t, a_t\} \quad (2.1)$$

na odziv okolja v času  $t + 1$ , na dejanje v času  $t$  in za vse vrednosti  $r, s'$  in prejšnjih dogodkov  $s_0, a_0, r_1, \dots, s_{t-1}, a_{t-1}, r_t, s_t, a_t$ . Če ima signal stanja *Markovo lastnost*, pa je odziv okolja v času  $t + 1$  odvisen samo od stanja in dejanja v času  $t$  in lahko dinamiko okolja definiramo z določitvijo le porazdelitve verjetnosti

$$Pr\{r_{t+1} = r, s_{t+1} = s' | s_t, a_t\}, \quad (2.2)$$

za vse  $r, s', s_t$  in  $a_t$ . Z drugimi besedami, signal stanja ima *Markovo lastnost*, in je *Markovo stanje*, če in samo če je (2.1) enako (2.2) za vse  $s', r$ , in preteklosti  $s_0, a_0, r_1, \dots, s_{t-1}, a_{t-1}, r_t, s_t, a_t$ . V tem primeru pravimo, da ima celotno okolje in naloga Markovo lastnost.

Če ima okolje Markovo lastnost, lahko iz enostavne dinamike predhodnega stanja (2.2) napovedujemo naslednje stanje in naslednjo nagrado za trenutno stanje in dejanje. V tem okolju lahko napovedujemo vsa stanja in nagrade v prihodnosti enako dobro kot bi lahko s popolno preteklostjo do trenutnega časa. Sledi enako, da je najboljša politika izbire dejanj enako dobra v Markovem stanju kot najboljša politika izbire dejanj s trenutnim stanjem in popolno zgodovino.

Čeprav Markova lastnost velikokrat ne drži popolnoma v nalogah okrepitvenega učenja, je vseeno zelo primerno razmišljati o stanju v okrepitvenem učenju kot približnemu Markovemu stanju. Omogoča nam razmišljati o odločitvah in vrednostih na podlagi trenutnega stanja.

Naloga okrepitvenega učenja, ki zadovoljuje Markovo lastnost imenujemo *Markov proces odločanja* (angl. *MDP – Markov decision process*). Če je prostor stanj in dejanj končno, potem to imenujemo *končen Markov proces odločanja* (angl. *finite MDP*).

Končen MDP je torej popolnoma definiran s:

- končno množico dosegljivih stanj  $S$ ,
- končno množico izvedljivih dejanj  $A$ ,
- prehodno funkcijo, definirano na vseh stanjih iz  $S$  in za vsa dejanja iz  $A$ , ki je za prehod v stanje  $s' \in S$  odvisna samo od trenutnega stanja  $s \in S$  in dejanja  $a \in A$ :

$$P(s'|s, a) = \Pr\{s_{t+1} = s' | s_t = s, a_t = a\}, \quad (2.3)$$

- nagrajevalno funkcijo, ki je, posledično od prehodne funkcije, tudi odvisna samo od trenutnega stanja in trenutnega dejanja:

$$R(s, a, s') = E[r_{t+1} | s_t = s, a_t = a, s_{t+1} = s']. \quad (2.4)$$

## 2.3 Diskretno okrepitveno učenje

Ta razdelek povzame okrepitveno učenje [15] v diskretnem primeru, to je, v primeru, ko je prostor stanj okolja diskreten in končen in čas je razdeljen v diskretne korake.

Politika  $\pi$  slika stanje v dejanje, kot omenjeno v razdelku 2.1. Za končne MDP lahko definiramo tudi *optimalno politiko* (angl. *optimal policy*)  $\pi^*$ . Naj bosta  $\pi$  in  $\pi^*$  politiki in  $V^\pi$  vrednostna funkcija politike  $\pi$  ter  $V^{\pi^*}$  vrednostna funkcija politike  $\pi^*$ . Politika  $\pi^*$  je optimalna, če ima vrednostno funkcijo  $V^{\pi^*}$  z naslednjo lastnostjo

$$V^{\pi^*}(s) \geq V^\pi(s), \forall s, \quad (2.5)$$

za vse možne politike  $\pi$ .

Kar je bilo do sedaj navedeno kot pričakovana dolgoročna nagrada je pogosto imenovan *pričakovan donos* (angl. *expected return*). Formalna definicija donosa nekega stanja v času  $t$  za poslednje nagrade  $r_{t+1}, r_{t+2}, r_{t+3} \dots$  je

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots = \sum_{k=0}^{T-t-1} \gamma^k r_{t+k+1}, \quad (2.6)$$

kjer je  $0 \leq \gamma \leq 1$ . Donos je vsota vseh nadaljnjih nagrad, ki jih pričakujemo po času  $t$  do končnega stanja v času  $T$ . V končnem stanju definiramo, da je prehod možen samo v isto stanje in nagrada ob prehodu vedno ničelna. S tem lahko poenotimo *epizodične* in *neskončne naloge* z uvedbo konstante  $\gamma$ , ki predstavlja *faktor popuščanja* (angl. *discount factor*), s tem, da je lahko  $T = \infty$  ali  $\gamma = 1$ , ampak ne oba hkrati. Pri neskončnih nalogah, ki jih ne moremo razdeliti na epizode, je  $T = \infty$ , saj se nikoli ne končajo, hkrati pa mora biti  $\gamma < 1$ , drugače lahko donos postane neskončen. Faktor popuščanja določa koliko želimo upoštevati prihodnje nagrade. Z  $\gamma = 0$  se začne učenec brigati samo za trenutno nagrado. Pri epizodičnih nalogah, kot so igre, je navadno  $\gamma = 1$ .

Za končne MDP lahko *vrednost stanja*  $s$  po politiki  $\pi$ , t.i. *vrednostno funkcijo*  $V^\pi(s)$ , definiramo kot pričakovan donos iz stanja  $s$  z nadaljnjim upoštevanjem politike  $\pi$ , formalno:

$$V^\pi(s) = E_\pi[R_t | s_t = s] = E_\pi \left[ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| s_t = s \right], \quad (2.7)$$

kjer  $E_\pi[\cdot]$  predstavlja pričakovano vrednostjo, če učenec sledi politiki  $\pi$ .

Podobno lahko *vrednost dejanja*  $a$  v stanju  $s$  po politiki  $\pi$ , t.i. *vrednostno funkcijo* dejanj  $Q^\pi(s, a)$ , definiramo kot pričakovan donos iz stanja  $s$  ob dejanju  $a$  in nadaljnjim upoštevanjem politike  $\pi$ , formalno:

$$Q^\pi(s, a) = E_\pi[R_t | s_t = s, a_t = a] = E_\pi \left[ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| s_t = s, a_t = a \right]. \quad (2.8)$$

## 2.4 Raziskovanje in izkoriščanje

Eden od izzivov okrepitvenega učenja, ki jih ne najdemo v ostalih oblikah strojnega učenja, je kompromis med raziskovanjem (angl. *exploration*) in izkoriščanjem (angl. *exploitation*). Med učenjem, ko učenec uporablja približek optimalne vrednostne funkcije za svoje vedenje, mu to omogoča pridobiti največjo znano nagrado, ampak nikjer ni zagotovil, da je ta znana politika tudi v splošnem najboljša. Boljša rešitev bi mogoče lahko bila najdena, če bi učenec imel dovoljenje raziskovati dejanja, ki jih še ni poizkusil.

Spodaj opisana  $\epsilon$ -požrešna izbira dejanj je zelo preprosta in v praksi pogosto uporabljena. Obstaja še veliko drugih metod, nekaterih bolj kompleksnih od drugih. Več primerov je v delu Thrun [19] ter Sutton in Barto [15].

### 2.4.1 $\epsilon$ -požrešna izbira dejanj

Ena izmed najbolj enostavnih pristopov k izbiri dejanja za ravnovesje med raziskovanjem in izkoriščanjem je uvod parametra  $\epsilon$ , ki določi verjetnost izbire naključnega dejanja. Na vsakem koraku po tej metodi učenec izbere naključno dejanje z verjetnostjo  $\epsilon$  in požrešno dejanje z verjetnostjo  $1 - \epsilon$ .

Velikokrat je koristno izbrati veliko naključnih dejanj ob začetku učenja in nato kot učenje napreduje znižati pogostost naključnih dejanj. S tem v fazi največjega učenja čimbolj raziščemo prostor stanj. Znižanje vrednosti  $\epsilon$  logično zniža stopnjo raziskovanja in zviša stopnjo izkoriščanja. Težava pri  *$\epsilon$ -požrešni metodi izbiranja dejanj* (angl.  *$\epsilon$ -greedy action selection*) je, da ne obstaja preprostega načina za izbiro vrednosti  $\epsilon$ . V veliko primerih je težavno izbrati kdaj povečati ali znižati število naključnih dejanj, ki naj jih učenec izbere.

## 3 Tabularne rešitve

V tem poglavju so opisani ključni algoritmi okrepitvenega učenja v svojih enostavnih oblikah – ko je prostor stanj in dejanj dovolj majhen za predstavitev ocenjene vrednostne funkcije s poljem ali tabelo. V teh primerih znajo algoritmi najti optimalno vrednostno funkcijo in optimalno politiko vedenja. To je v nasprotju z aproksimacijskimi metodami opisanimi v naslednjem poglavju, ki znajo najti le približne rešitve, ampak jih lahko uporabimo učinkovito na veliko večjih problemih.

### 3.1 Dinamično programiranje

Leta 1957 je Bellman [REF] objavil ključen članek, ko je uvedel izraz *dinamično programiranje* (angl. *DP – dynamic programming*), ki se nanaša na zbirko algoritmov, ki jih lahko uporabimo za izračunati optimalno politiko za MDP. Algoritmi dinamičnega programiranja potrebujejo popoln in natančen model okolja za delovanje. Vsi se nanašajo na Bellmanovo enačbo

$$\begin{aligned}
 V_{\pi}(s) &= E_{\pi}[R_t | s_t = s] \\
 &= E_{\pi} \left[ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| s_t = s \right] \\
 &= E_{\pi} \left[ r_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \middle| s_t = s \right] \\
 &= \sum_a \pi(a|s) \sum_{s'} P(s'|s, a) \left[ R(s, a, s') + \gamma E_{\pi} \left[ \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \middle| s_{t+1} = s' \right] \right] \\
 &= \sum_a \pi(a|s) \sum_{s'} P(s'|s, a) \left[ R(s, a, s') + \gamma V_{\pi}(s') \right].
 \end{aligned} \tag{3.1}$$

Bellmanova enačba pravi, da je možno izračunati vrednost vseh stanj  $s$ , to je,  $V(s)$ , z rekurzivnimi koraki čez vsa stanja ki so *povezana* s stanjem  $s$ . Izraz *povezana* je tukaj definiran kot: dva stanja  $s$  in  $s'$  sta povezana če in samo če je  $P(s'|s, a) > 0$  pri vseh  $a \in A$ . Postopek je ponovljen za vsak  $s'$  povezan s  $s$  dokler ne pridemo do končnega stanja. Verjetnost  $P(s'|s, a)$  določa koliko ima izraz  $R(s, a, s') + \gamma V_{\pi}(s')$  vpliv na vrednost stanja  $s$ . Pomembno se je zavedati, da potrebujemo prehodno funkcijo  $P$  za izračun enačbe. Če je  $P$  znan se enačba razširi na sistem enačb, ki ga lahko rešimo enostaven način.



Bellman je opozoril na težavo, ki jo je imenoval *prekletstvo dimenzionalnosti* (angl. *the curse of dimensionality*), to je število izračunov potrebnih za rešitev sistema enačb se večja eksponentno z številom vhodnih dimenzij. To je dobro znana težava v področju strojnega učenja. Težava je bolj v tem, da se prostor stanj večja eksponentno s številom vhodnih dimenzij, kot pa v Bellmanovem pristopu k rešitvi MDPjev.

*Teorem izboljšanja politike* (angl. *policy improvement theorem*) navaja, da ko spremenimo pot politike na pot, ki je po vrednostni funkciji boljša, dobimo izboljšano politiko. Ta lastnost je uporabljena v metodah dinamičnega programiranja tako kot tudi v veliki meri drugih algoritmov okrepitvenega učenja.

## 3.2 Monte Carlo metode

$$V(s_t) \leftarrow V(s_t) + \alpha[R_t - V(s_t)] \quad (3.2)$$

## 3.3 Učenje na podlagi časovne razlike - TD(0)

$$V(s_t) \leftarrow V(s_t) + \alpha[R_{t+1} + \gamma V(s_{t+1}) - V(s_t)] \quad (3.3)$$

## 3.4 Združitev metod - TD( $\lambda$ )

## 4 Posploševanje in funkcijska aproksimacija

### 4.1 Umetne nevronske mreže

### 4.2 Metode gradient descent

## 5 Namizna igra Hex

### 5.1 Ozadje

### 5.2 Učenje

### 5.3 Rezultati

## 6 Zaključek

Iz okrepitvenega učenja so se razvili solidni matematični temelji in impresivne aplikacije. Računska študija okrepitvenega učenja je sedaj obsežna, z aktivnimi raziskovalci na raznolikih disciplinah kot so psihologija, teorija krmiljenja (angl. control theory), operacijske raziskave (angl. operations research), umetna inteligenca in nevroznanost. Posebej pomembne so zveze z optimalnim nadzorom in dinamičnim programiranjem. Celoten problem učenja iz interakcije za dosego ciljev ni še zdaleč rešen, vendar se je naše razumevanje na tem področju bistveno izboljšalo. Sedaj lahko postavimo sestavne ideje kot so učenje na podlagi časovne razlike, dinamično programiranje in funkcijske aproksimacije skladno s celotnim problemom.

Eden večjih trendov katerih je okrepitveno učenje deležno je večji stik med umetno inteligenco in ostalimi inženirskimi disciplinami. Nedolgo nazaj se je umetno inteligenco smatralo kot popolnoma ločeno od teorije nadzora in statistiko [15]. Imelo je opravka z logiko in simboli, ne pa s števili. Umetna inteligenca so bili obširni LISP programi, ne linearna algebra, diferencialne enačbe ali statistika. V zadnjih desetletjih se je ta pogled spremenil. Moderni raziskovalci umetne inteligence sprejemajo statistične in nadzorne algoritme kot pomembne konkurenčne metode ali pa enostavno kot orodja. Prej prezrta področja med umetno inteligenco in konvencionalnega inženirstva so sedaj med najbolj aktivnimi, vključno z nevronskimi mrežami, pametnim nadzorom in okrepitvenim učenjem. V okrepitvenem učenju se ideje optimalne teorije nadzora in stohastične aproksimacije razširijo za nasloviti širše in bolj ambiciozne cilje umetne inteligence.

Ray Kurzweil, inventor in futurist, je, v svoji nefiktivni knjigi “The Singularity Is Near: When Humans Transcend Biology” [13], opisal svoj zakon o pospeševanju donosov, ki napoveduje eksponentno povečanje v tehnologijah kot so računalništvo, genetika, nanotehnologija, robotika in umetna inteligenca. Predvideva, da bo do okrog leta 2020 obstajal računalnik za tisoč ameriških dolarjev, ki bo imel računsko zmoglost posnemati človeško inteligenco. Po tem, pričakuje, da bo tehnologija optičnega zajemanja človeških možganov pripomogla k učinkovitemu modelu človeške inteligence do okrog 2025. Ta dva elementa bosta omogočala računalnikom opraviti Turingov preizkus do leta 2029. In do zgodnjih 2030 bo količina nebiološkega računanja prekoračilo zmoglost vse žive biološke inteligence človeštva. Končno eksponentno povečanje v

računski zmognosti bo privedlo do dogodka Singularnosti – močno in moteče preoblikovanje v človeški sposobnosti – leta 2045.

## Literatura

- [1] A. L. Samuel, *Some Studies In Machine Learning Using the Game of Checkers*, IBM Journal on Research and Development, 1959. (*Citirano na strani 2.*)
- [2] A. Persson, *Using Temporal Difference Methods In Combination With Artificial Neural Networks to Solve Strategic Control Problems*, KTH Numerical Analysis and Computer Science, Royal Institute of Technology, Stockholm, Sweden, 2004. (*Citirano na strani 2.*)
- [3] C. Balkenius, J. Morén, *Computational Models of Classical Conditioning: A Comparative Study*, From animals to animats 5: proceedings of the fifth international conference on simulation of adaptive behavior. MIT Press/Bradford Books: Cambridge, MA, 1998. (*Citirano na strani 4.*)
- [4] C. E. Shannon, *A Mathematical Theory of Communication*, Bell Sys. Tech. Journal, vol. 27, 1948. (*Citirano na strani 2.*)
- [5] C. Stangor, *Introduction to Psychology*, MIT Press, Cambridge, MA, 2011. (*Citirano na straneh vii, 4, 5, 6 in 7.*)
- [6] D. Porter, A. Neuringer, *Music Discrimination By Pigeons*, Journal of Experimental Psychology: Animal Behavior Processes 10.2, 1984. (*Citirano na strani 7.*)
- [7] E. L. Thorndike, *Animal Intelligence: An Experimental Study of the Associative Processes In Animals*, Psychological Monographs: General and Applied 2.4, 1898. (*Citirano na strani 5.*)
- [8] E. L. Thorndike, *Animal Intelligence: Experimental Studies*, The Journal of Nervous and Mental Disease 39.5, 1912. (*Citirano na strani 5.*)
- [9] G. Markkula, *Playing risk aversive go on a large board using local neural network position evaluation functions*, Department of physical resource theory and complex systems group, Chalmers university of technology Göteborg, 2004. (*Citirano na strani 2.*)
- [10] G. Tesauro, *Practical issues in temporal difference learning*, Machine Learning 4, 1992. (*Citirano na strani 2.*)

- [11] I. Pavlov, *Conditioned Reflexes*, Courier Dover Publications, 2003. (*Citirano na strani 3.*)
- [12] R. A. Rescorla, *Pavlovian Conditioning: It's Not What You Think It Is*, American Psychologist, 1988. (*Citirano na strani 3.*)
- [13] R. Kurzweil, *The Singularity Is Near: When Humans Transcend Biology*, Penguin, 2005. (*Citirano na strani 19.*)
- [14] R. S. Sutton, *Learning to predict by the methods of temporal difference*, Machine Learning 3, 1988. (*Citirano na strani 2.*)
- [15] R. S. Sutton, A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, MA, 1998. (*Citirano na straneh vii, 2, 4, 8, 9, 12, 14 in 19.*)
- [16] M. E. Bouton, *Learning and Behavior: A Contemporary Synthesis*, Sinauer Associates, 2007. (*Citirano na strani 3.*)
- [17] M. Ito, T. Yoshioka, S. Ishii, *Strategy acquisition for the game "Othello" based on reinforcement learning*, Technical report, Nara Institute of Science and Technology, 1998. (*Citirano na strani 2.*)
- [18] N. N. Schraudolf, P. Dayan, T. Sejnowski, *Learning to evaluate go positions via temporal difference methods*, Vol. 62 of Studies in fuzziness and soft computing, Springer Verlag, 2001. (*Citirano na strani 2.*)
- [19] S. B. Thrun, *The Role of Exploration in Learning Control*, Department of Computer Science, Carnegie-Mellon University, 1992. (*Citirano na strani 14.*)
- [20] S. J. Shettleworth, *Cognition, Evolution and Behavior*, Oxford University Press, 2009. (*Citirano na strani 3.*)
- [21] S. Legg, M. Hutter, *A Collection of Definitions of Intelligence*, Frontiers in Artificial Intelligence and Applications, 2007. (*Citirano na strani 1.*)
- [22] S. Watanabe, J. Sakamoto, M. Wakita, *Pigeons' Discrimination of Paintings by Monet and Picasso*, Journal of the experimental analysis of behavior 63.2, 1995. (*Citirano na strani 7.*)
- [23] T. L. Brink, *Psychology: A Student Friendly Approach*, San Bernardino Community College, 2008. (*Citirano na strani 3.*)
- [24] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, M. Riedmiller, *Playing Atari with Deep Reinforcement Learning*, arXiv preprint arXiv:1312.5602, 2013. (*Ni citirano.*)