**RĪGAS TEHNISKĀ UNIVERSITĀTE**

Datorzinātnes un informācijas tehnoloģijas fakultāte

Lietišķo datorsistēmu institūts

**Domeniks Delvers**

bakalaura akadēmisko, pirmā līmeņa studiju programmas

students, stud. apl. nr. 201RDB298

**VAIRĀKU AĢENTU**

**PĀRKLĀJUMA CEĻA MEKLĒŠANA**

**BAKALAURA DARBS**

Zinātniskais vadītājs Doktors, Profesors

Agris Ņikitenko

RĪGA 2022

**RĪGAS TEHNISKĀ UNIVERSITĀTE**

**DATORZINĀTNES UN INFORMĀCIJAS TEHNOLOĢIJAS FAKULTĀTE**

<institūta nosaukums>

<struktūrvienības nosaukums>

**<Noslēguma darba no saraksta: bakalaura darbs, maģistra darbs, diplomprojekts, kvalifikācijas darbs> izpildes lapa**

Noslēguma darba autors:

students(-e) <vārds, uzvārds>\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(paraksts, datums)

Noslēguma darbs ieteikts aizstāvēšanai:

Zinātniskais vadītājs:

<zinātniskais grāds, amats, vārds, uzvārds> \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(paraksts, datums)

Anotācija

Putekļsūcēj-robots, simulācija, ceļa plānošanas algoritms

Mūsdienās mājas tīrīšanas roboti paliek arvien izplatītāki, un īpaši – putekļsūcēj-roboti. Viena no īpašībām, kas raksturo robotu ir tā tīrīšanas ātrums, tāpēc ražotāji mēģina izstrādāt pēc iespējas efektīvākus algoritmus, lai robots spētu iztīrīt visu dzīvokļa platumu pēc iespējas mazākā laikā. Šajā darbā tiek apskatīti vairāki tādi algoritmi, kā arī simulēti virtuālajā vidē un salīdzināti dažādos apstākļos.

Darbā ir X lappušu, Y attēlu, Z tabulu, W pielikumu un Q izmantoto avotu skaits.

Abstract

Nowadays, home cleaning robots are becoming more and more popular, particularly vacuum-cleaning robots. One of the characteristics that defines a robot is its cleaning speed, so manufacturers are trying to develop as efficient algorithms as possible so that the robot can clean the entire width of the apartment in the shortest possible time. In this work, several such algorithms are examined and simulated in a virtual environment and compared under different conditions.

Satura rādītājs

[Anotācija 2](#_Toc132117968)

[Abstract 3](#_Toc132117969)

[Satura rādītājs 4](#_Toc132117970)

[Ievads 6](#_Toc132117971)

[1. Literatūras pārskats 8](#_Toc132117972)

[1.1. Tradicionālas pārklājuma meklēšanas tehnikas 8](#_Toc132117973)

[1.1.1. Šūnu dekompozīcija 8](#_Toc132117974)

[1.1.2. Bustrofedona dekompozīcija 8](#_Toc132117975)

[1.1.3. Režģa bāzētās metodes 9](#_Toc132117976)

[1.1.4. Aptveroša koka metodes 9](#_Toc132117977)

[1.2. Apmācība ar pastiprinājumu vienīga aģenta pārklājuma ceļa plānošanai 9](#_Toc132117978)

[1.2.1. Q – apmācība 10](#_Toc132117979)

[1.2.2. Dziļie Q – tīkli 10](#_Toc132117980)

[1.2.3. Aktiera – kritiķa metodes 10](#_Toc132117981)

[1.2.4. Monte Carlo koka pārmeklēšana 11](#_Toc132117982)

[1.3. Apmācība ar pastiprinājumu vairāku aģentu pārklājuma ceļa meklēšanai 11](#_Toc132117983)

[1.3.1. Neatkarīgi mācekļi 12](#_Toc132117984)

[1.3.2. Kopīgu darbību mācekļi 12](#_Toc132117985)

[1.3.3. Centralizēta apmācība ar decentralizētu izpildi 12](#_Toc132117986)

[1.3.4. Komunikācijas pamatotas pieejas 12](#_Toc132117987)

[2. ProBlēmas Nostādne 13](#_Toc132117988)

[2.1. Vairāku aģentu pārklājuma ceļa plānošanas formalizēšana iekštelpu vidēs 13](#_Toc132117989)

[2.1.1. Aģenta un vides modelis 13](#_Toc132117990)

[2.1.2. Mērķi un ierobežojumi 13](#_Toc132117991)

[2.2. Izaicinājumi vairāku aģentu pārklājuma ceļa plānošanā 14](#_Toc132117992)

[2.2.1. Mērogojamība 14](#_Toc132117993)

[2.2.2. Koordinācija 14](#_Toc132117994)

[2.2.3. Neskaidrība un dinamiskas vides 14](#_Toc132117995)

[2.2.4. Komunikācijas ierobežojumi 14](#_Toc132117996)

[3. Pastiprināšanas mācīšanas pārskats 16](#_Toc132117997)

[3.1. Ievads pastiprināšanas mācīšanās 16](#_Toc132117998)

[3.1.1. Galvenie jēdzieni 16](#_Toc132117999)

[3.1.2. Markova lēmumu procesi (MLP) 16](#_Toc132118000)

[3.2. Vienīgā aģenta pastiprināšanas mācīšanas algoritmi 16](#_Toc132118001)

[3.2.1. Uz modeli balstītas un no modeļa brīvas metodes 16](#_Toc132118002)

[3.2.2. Uz vērtībām balstītas metodes 17](#_Toc132118003)

[3.2.3. Uz politikām balstītas metodes 17](#_Toc132118004)

[3.2.4. Aktiera – kritiķa metodes 17](#_Toc132118005)

[3.3. Izpētes un izmantošanas kompromiss 18](#_Toc132118006)

[3.3.1. Epsilon – *greedy* 18](#_Toc132118007)

[3.3.2. Augšējas uzticības robeža 18](#_Toc132118008)

[3.3.3. Tompsona izlase 18](#_Toc132118009)

[4. Vairāku aģentu pastiprināšanas mācīšanas 19](#_Toc132118010)

[4.1. Vairāku aģentu pastiprināšanas mācīšanas pārskats 19](#_Toc132118011)

[4.1.1. Problēmas vairāku aģentu pastiprināšanas mācībās 19](#_Toc132118012)

[4.1.2. Kooperatīvie, konkurējošie un jaukti scenāriji 19](#_Toc132118013)

[4.2. Vairāku aģentu mācīšanas *frameworks* 19](#_Toc132118014)

[4.2.1. Neatkarīga Q-mācīšanās (IQL) 19](#_Toc132118015)

[4.2.2. Kopīgas darbības mācīšanās (JAL) 20](#_Toc132118016)

[4.2.3. Koordinētā pastiprināšanās mācīšanās (CRL) 20](#_Toc132118017)

[4.3. Komunikācija vairāku aģentu pastiprināšanas mācīšanās 20](#_Toc132118018)

[4.3.1. Ziņu nodošanas pieejas 20](#_Toc132118019)

[4.3.2. Diferencējamā starpaģentu mācīšanās (DIAL) 20](#_Toc132118020)

[4.3.3. Komunikācijas protokoli un arhitektūras 20](#_Toc132118021)

[4.4. Centralizēta apmācība ar decentralizētu izpildi 21](#_Toc132118022)

[4.4.1. Pretfaktiskie vairāku aģentu politikas gradienti (COMA) 21](#_Toc132118023)

[4.4.2. QMIX un VDN 21](#_Toc132118024)

[5. Iekštelpas vides procedurāla ģenerēšana 22](#_Toc132118025)

[5.1. Vides skeleta veidošana 22](#_Toc132118026)

[5.2. Durvju izvietojums 23](#_Toc132118027)

[5.3. Mēbeļu un šķēršļu ģenerēšanā 24](#_Toc132118028)

[Atsauksmes 26](#_Toc132118029)

Ievads

Pēdējos gados ir palielinājusies interese pret pilna pārklājuma ceļa meklēšanas algoritmu attīstīšanu robotizēto sistēmu un mākslīgā intelekta nozarēs. Iespēja efektīvi izbraukt un pārklāt dažādas kompleksas, dinamiskas vides ir ļoti svarīga vairākiem uzdevumiem: glābšanas operācijām, drošības uzraudzībai, kartēšanai, vides monitoringam un īpaši tīrīšanai. Mājas tīrīšanas robotu tirgū jau no 2014 gada ir novērojama stabila izaugsme. Palielinoties interesei pēc gudrās mājas tehnoloģijas un automatizācijas, pieprasījums pēc mājas tīrīšanas robotiem arvien aug. Īpaša tirgus izaugsme tika novērota COVID – 19 pandēmijas laikā. Sakarā ar pandēmiju daudzi cilvēki bija spiesti pavadīt vairāk laika mājās, tāpēc palielinājās interese pēc produktiem, kas ļauj saturēt dzīvokļus tīrākus. Saskaņā ar “*Research and Markets*” pētījumu, starp 2021. gadu un 2026 gadu, var sagaidīt 20% salikto gada pieauguma tempu (*Compound Annual Growth Rate*). Zinātniskais raksts pamato tik strauju izaugsmi ar palielinātu pieprasījumu pēc inovatīviem mājas tīrīšanas veidiem, kā arī gudrās mājas tehnoloģijas izplatību. Mājas tīrīšanas robotu tirgū ir daudz konkurējošo ražotāju, piemēram: “*iRobot*”, “*Dyson*”, “*LG Electronics*”, “*Ecovacs* *Robotics*” un “*Samsung* *Electronics*”. Augot pieprasījumam pēc mājas tīrīšanas robotiem, ražotāji cenšas uzlabot savu robotu efektivitāti, lai piesaistītu vairāk klientu.

Īpaši būtiski robotu efektivitāti ietekmē tā programmatūra: navigācijas algoritms, kartēšanas algoritms un ceļa plānošanas algoritms. Lielākā daļā tieši no ceļa plānošanas algoritma ir atkarīgs, cik ātri un kvalitatīvi robots paveiks savu uzdevumu. Algoritmam ir jānodrošina pilns telpas pārklājums, šķēršļu apbraukšana, kā arī jāatrod ceļš ar minimālo garumu un mazāku pagriezienu skaitu. Lai veicinātu pilnu pārklājumu ražotāji izmanto dažādas metodes, kuras savukārt ir atkarīgas no vienlaicīgas lokalizācijas un kartēšanas (*SLAM*) sistēmas precizitātes. Mūsdienīgie tīrīšanas roboti izmanto heiristiskas metodes pārklājuma ceļa meklēšanai vairāku iemeslu dēļ. Pirmkārt *SLAM* sistēmu neprecizitātes dēļ ir grūti precīzi izbraukt iepriekš ieplānoto trajektoriju, kā arī nav iespējams pietiekami precīzi nokartēt vidi. Otrkārt visi mūsdienās pastāvošie pilna pārklājuma algoritmi ir nepiemēroti mājas apstākļiem.

Šodien neviens ražotājs nepiedāvā efektīvu risinājumu klientiem, kuri vēlās izmantot savās mājās vairākus tīrīšanas robotus vienlaicīgi. Mūsdienas tīrīšanas robotiem nav sistēmu, kas nodrošinātu efektīvu vairāku aģentu sadarbību kopīgas problēmas risināšanai.

Ceļu plānošanas uzdevums dzīvokļos un mājās ir izaicinošs vides nenoteiktības un šķēršļu klātbūtnes dēļ. Viens veids, kā efektīvi meklēt risinājumu šīm izaicinājumam, ir algoritma testēšanas laikā izmantot procedurāli ģenerētas vides, kurās vides plāns un šķēršļi tiek ģenerēti algoritmiski, nevis tiek iepriekš definēti. Tas ļauj izveidot ļoti dažādas un sarežģītas vides ar lielu variāciju. Šādās vidēs var testēt dažādu ceļu plānošanas algoritmu veiktspēju. Šajā darbā tiek salīdzināti dažādi pilna pārklājuma ceļa meklēšanas algoritmi procedurāli ģenerētās telpās, ar fokusu uz mājas tīrīšanas robotiem. Darbā tiek salīdzināti dažādi algoritmi pēc to ātruma, efektivitātes un izturības.

Darba mērķis ir simulēt vairāku aģentu pārklājuma problēmas risinošo algoritmu virtuālajā vidē, pie dažādiem apstākļiem un izpētīt piedāvātā risinājuma efektivitāti. Lai veiksmīgi sasniegtu šo mērķi tika izvirzīti sekojoši darba uzdevumi:

1. Sagatavot virtuālo vidi eksperimentu veikšanai. Šīm nolūkam tika izmantota “Unity 3D”, kur procedurāli tiek ģenerēti dažādi dzīvokli.
2. Īstenot istabas apiešanas algoritmu, ar kuru palīdzību var veiksmīgi apiet iepriekš ģenerētas vides.
3. Savākt datus un normalizēt tos, lai veiktu pilnvērtīgus secinājumus.

Darba pirmajā nodaļā ir aprakstīts iekštelpas vides procedurālas ģenerēšanā “Unity3D” vidē. Darba otrajā nodaļā tiek aprakstīti testētie pilnās apiešanas algoritmi, kā arī pamatota to izvēle. Trešajā nodaļā tiek aprakstīts testēšanas, datu savākšanas un secināšanas process.

1. SAISTĪTIE PĒTIJUMI
   1. Tradicionālas pārklājuma meklēšanas tehnikas

Pārklājuma ceļa meklēšana (CPP) ir bieži sastopama problēma robotikā un automatizācijā, kur mērķis ir atrast optimālu vai tuvu optimālam ceļu, kas ļauj robotam vai vairākiem robotiem apmeklēt katru punktu dotās vides teritorijā, minimizējot zudumus, piemēram, laiku vai enerģijas patēriņu. Šī problēma ir nozīmīga dažādās jomās, tostarp vides monitoringā, inspekcijā, lauksaimniecībā, tīrīšanā, meklēšanā un glābšanā, kā arī uzraudzībā.

Tradicionālās pārklājuma ceļa plānošanas tehnoloģijas galvenokārt balstās uz ģeometriskām, grafu bāzētām vai heiristikām pieejām. Šīs metodes parasti tiešās uz algoritmu izstrādi, kas radītu efektīvu pārklājuma ceļu dotajai videi, ņemot vērā vairākus faktorus, piemēram, robota uztveres un darbības spējas, vides ģeometriju, struktūru un iespējamos šķēršļus. Daži no svarīgākajiem faktoriem, ko jāņem vērā tradicionālo CPP algoritmu izstrādē, ietver: pārklājuma pilnība, ceļa optimizēšana, skaitļošanas sarežģītība, mērogošana un izturība.

* + 1. Šūnu dekompozīcija

Šūnu dekompozīcija ir tehnika, kas sadala vidi vairākos nepārklājošās šūnās, kuras nākotnē varētu apceļot aģents. Viens no agrākajiem darbiem šajā nozarē pieder *Moravec* un *Elfes* kungiem (1985), kas pielietoja okupācijas režģa pamatā esošo pieeju robotu kartēšanai un navigācijai[1]. Vēlāk, zinātnieki *Choset* un *Pignon* (1997) piedāvāja bustrofedona dekompozīcijas metodi, kas sadala vidi šūnās, kas robotiem ir viegli šķērsojamas[2].

* + 1. Bustrofedona dekompozīcija

Bustrofedona dekompozīcija, ko ieviesa *Choset* un *Pignon* (1997), ir populāra šūnu dekompozīcijas pieeja ceļu plānošanai, kas nodrošina pilnīgu pārklājumu. Šī tehnika sadala vidi šūnās, kuras pēc tam aģents šķērso veicot “S” veida kustības. Pētnieki ir turpinājuši attīstīt un uzlabot šo pieeju, piemēram *Acar* (2002) darbā, kur tie piedāvāja pilnīgu ceļu plānošanas algoritmu robotiem ar ierobežotām sensoru spējam[3].

* + 1. Režģa bāzētās metodes

Uz režģi bāzētās pieejas diskretizē vidi, lai iegūtu vides režģi un pēc tam izmanto grafu bāzētos algoritmus pārklājuma nodrošināšanai. Piemēram, *Zelinsky* (1993) ieviesa viļņa frontes bāzēto pārklājuma ceļa meklēšanas algoritmu, kas izmanto režģa rūtiņas, kā pamatelementus vides izmeklēšanai[4]. Vēlāk *Huang* (2001) piedāvāja tā saucamo “optimālo nejaušo pastaigu” režģa bāzētu pārklājuma ceļa meklēšanas algoritmu[5].

* + 1. Aptveroša koka metodes

Tehnikas, kuras pamatojās uz aptveroša koka metodi, veido koku, kas aptver visu vidi, un kuram izseko aģents, lai paveiktu pilnu vides pārklāšanu. *Gabriely* un *Rimon* (2001) ieviesa aptveroša koka pārklājuma (STC) algoritmu, kas veido aptverošo koku par režģa pamatā esošo vidi un garantē pilnīgu pārklājumu[6]. Šo pieeju ir turpmāk izpētījusi un paplašinājuši citi pētnieki, piemēram *Hazon* un  *Kaminka* (2005) darbā, kurā tie piedāvāja tiešsaistes algoritmu STC metodei[7].

* 1. Apmācība ar pastiprinājumu vienīga aģenta pārklājuma ceļa plānošanai

Viena aģenta pastiprinātā mācīšanās (RL) pārklājuma ceļa plānošanai ir pāreja no tradicionālajām noteiktajām un ģeometriskajām metodēm uz adaptīvāku un datu-definēto pieeju. Pastiprinātā mācīšanās ir mašīnmācīšanās nozare, kur aģents mācās pieņemt lēmumus, mijiedarbojoties ar vidi un saņemot atsauksmes balvas vai soda veidā. Aģenta galvenais mērķis ir maksimizēt kopējo apbalvojumu laikā.

Viena aģenta RL piemērošana pārklājuma ceļa plānošanai nodrošina vairākas priekšrocības salīdzinājumā ar tradicionālajām metodēm:

* Pielāgojamība: RL algoritmi var mācīties pielāgoties dažādām vides situācijām, ieskaitot dinamiskas vai neskaidras situācijas. Viņi var modificēt savu rīcību, pamatojoties uz reāllaika atsauksmēm no vides, kas padara tos piemērotākus, lai rīkoties dinamiskās vidēs.
* Vispārīgums: Tā kā RL algoritmi mācās no datiem, tie var potenciāli labi vispārināties jaunām un neredzētām vides situācijām. Tas nozīmē, ka, pēc trenēšanās dažādos vides apstākļos, RL aģents varētu efektīvi darboties iepriekš neredzētās situācijās.
* Mērogojamība: RL algoritmi var potenciāli mērogot līdz lielām un sarežģītām vides situācijām, jo tie nebalstās uz skaidrām vides reprezentācijām. Tā vietā tie mācās pieņemt lēmumus, pamatojoties uz lokāliem novērojumiem.
* Pastāvīga mācīšanās: RL aģenti var turpināt mācīties un uzlabot savu veiktspēju pat pēc sākotnējās apmācības. Tas ļauj tiem precizēt savas politikas un pielāgoties jaunām situācijām vai vides izmaiņām laikā.

1.2.1. Q – apmācība

*Engel* (2005) demonstrēja Q – apmācības potenciālu vienīga aģenta pārklājuma ceļa plānošanas uzdevuma risināšanā[8]. Pētnieks piemēroja Q – apmācības metodi lauksaimnieciskas vides autonomām izsmidzināšanas nolūkam un pieradīja tehnoloģijas efektivitāti.

1.2.2. Dziļie Q – tīkli

Dziļie Q – tīkli (DQN) paplašina tradicionālo Q – mācīšanas metodi, izmantojot dziļos neironu tīklus funkciju aproksimācijai. *Mnih* (2015) veiksmīgi pielietoja DQN “Atari” konsoles spēlēm[9]. Lai gan DQN nav plaši pielietoti pārklājuma ceļa plānošanai, to veiksmīgs pielietojums komplicētās vidēs pierāda potenciālu pielietojamību.

1.2.3. Aktiera – kritiķa metodes

Aktiera – kritiķa metodes ir piemērotas pārklājuma ceļa plānošanai, kā redzams *Konda* un *Tsitsiklis* (2002) darbā, kas prezentēja aktiera – kritiķa algoritmu mācīšanas ar pastiprinājumu metodei[10]. Viņi pierādīja, ka algoritms spēj mācīties un pielāgoties dažādām vidēm un apstākļiem.

1.2.4. Monte Carlo koka pārmeklēšana

Monte Carlo koka pārmeklēšana (MCTS) ir ceļa plānošanas algoritms, kas apvieno Monte Carlo simulācijas ar koka pārmeklēšanu. MCTS tika veiksmīgi pielietots vienīga aģenta pārklājuma ceļa plānošanai, kā to pierādījis *Coulom* (2006), kurš izmantoja MCTS spēlei “Go” [11].

* 1. Apmācība ar pastiprinājumu vairāku aģentu pārklājuma ceļa meklēšanai

Vairāku aģentu pastiprinātā mācīšanās (MARL) paplašina viena aģenta pastiprinātās mācīšanās, kur vairākiem aģentiem ir jākoordinējas un jāsadarbojas, lai sasniegtu kopējo mērķi. Pārklājuma ceļa plānošanas kontekstā vairāku aģentu pastiprinātā mācīšanās var tikt izmantota, lai izstrādātu efektīvus algoritmus, kas koordinētu robota komandu, lai kopīgi pētītu un pārklātu vidi.

Daudzaģentu pastiprinātās mācīšanās (MARL) pielietošana pārklājuma ceļa plānošanai sniedz vairākas priekšrocības:

* Sadalītā lēmumu pieņemšana: MARL algoritmi var atļaut sadalītu lēmumu pieņemšanu starp vairākiem robotiem, ļaujot tiem strādāt kopā un pielāgot savu rīcību, pamatojoties uz lokālo informāciju un sadarbību ar citiem aģentiem. Tas var novest pie uzlabotas efektivitātes un izturības salīdzinājumā ar centralizētām lēmumu pieņemšanas pieejām.
* Mērogojamība: Vairāku aģentu pastiprinātā mācīšanās var potenciāli labāk mērogoties uz lielām un sarežģītām vidēm, jo darba slodze ir sadalīta starp vairākiem aģentiem. Katrs aģents mācās pieņemt lēmumus, pamatojoties uz saviem vietējiem novērojumiem samazinot kopējo problēmas skaitļošanas sarežģītību.
* Izturība: MARL var novest pie izturīgākiem risinājumiem neskaidru situāciju un dinamisku vides apstākļu gadījumā. Izmantojot vairāku aģentu kolektīvo intelektu, sistēma var labāk tikt galā ar negaidītiem šķēršļiem, vides izmaiņām vai pat ar individuālo aģentu neveiksmēm.
* Uzdevumu specializācija: Vairāku aģentu sistēmas var nodrošināt uzdevumu specializāciju starp aģentiem, kur katrs aģents mācās izcili pārvaldīt konkrētus apakšuzdevumus pārklājuma ceļa plānošanas problēmā. Tas var novest pie efektīvākiem pārklājuma ceļa veidošanas stratēģijām.

1.3.1. Neatkarīgi mācekļi

Neatkarīga Q – mācīšanas metode tika pielietota vairāku aģentu pārklājuma ceļa plānošanai *Matignon* (2007) darbā, kas ieviesa sadarbības Q – mācīšanas algoritmu vairāku aģentu pārklājuma ceļa meklēšanai[12]. Darbs pierāda, ka pieeja var efektīvi koordinēt vairākus robotus, risinot nenoteiktības dinamiskās vidēs.

1.3.2. Kopīgu darbību mācekļi

Kopīgu darbību mācības metode ņem vērā visu aģentu kopējās darbības mācību procesā. *Oliehoek* (2006) pielietoja kopīgu darbību mācības metodi vairāku aģentu pārklājuma ceļa meklēšanai, prezentējot decentralizētu algoritmu, kas ir balstīts uz vairāku aģentu Markova lēmumu procesiem (MMDP)[13]. Pētnieks demonstrēja algoritma mērogojamību un izturību dažādos testēšanas scenārijos.

1.3.3. Centralizēta apmācība ar decentralizētu izpildi

Šī pieeja centralizēti apmāca visus aģentus un pēc tam ļauj tiem izpildīt savas politikas decentralizētā veidā. *Lowe* (2017) ieviesa vairāku aģentu aktiera – kritiķa metodi sadarbības un konkurējošās vidēs[14]. Pētījums apskata šādas pieejas pielietojamību dažādiem sadarbības un konkurējošās vides uzdevumiem. Šajā darbā pieeja nav tiešā veidā pielietota pārklājuma ceļa meklēšanai, tomēr darbs demonstrē šīs pieejas potenciālu vairāku aģentu sistēmās.

1.3.4. Komunikācijas pamatotas pieejas

Komunikācijas iekļaušana starp aģentiem var ievērojami uzlabot koordināciju un veiktspēju vairāku aģentu mašīnmācības procesā. *Foerster* (2016) ieviesa diferencējamo starp-aģentu mācīšanas (DIAL) struktūru, kas ļauj aģentiem mācīties komunikācijas politikas, izmantojot *backpropagation* [15]*.* Pētnieku ieviestā pieeja var būt piemērojama vairāku aģentu pārklājuma ceļa plānošanai, kur koordinācija un komunikācija starp aģentiem ir būtiska.

1. ProBlēmas Nostādne
   1. Vairāku aģentu pārklājuma ceļa plānošanas formalizēšana iekštelpu vidēs
      1. Aģenta un vides modelis

Vairāku aģentu pārklājuma ceļa meklēšanas kontekstā vide parasti tiek reprezentēta diskretizēta režģa vai grafa formā, kur katra šūna vai mezgls atbilst noteiktai lokācijai iekštelpu vidē[16]. Šķēršļi, piemēram sienas vai mēbeles, tiek attēlotas kā bloķētas vai nepieejamas šūnas vai mezgli.

Katrs sistēmas aģents ir mobilais robots, aprīkots ar uztveres un vadības spējām, kas ļauj tam uztvert un pārvietoties vidē. Aģenta stāvoklis var ietvert tā pašreizējo pozīciju, orientāciju un citu svarīgu informāciju, piemēram vietējo karti vai citu aģentu stāvokļus. Pieņem, ka aģenti ir homogēni attiecībā uz savām spējām, un to darbības var ietvert kustību uz kaimiņu šūnām vai mezgliem, pagriešanos vai saziņu ar citiem aģentiem. Pārejas modelis, kas apraksta aģenta stāvokļa izmaiņas atkarībā no vides rakstura un robotu dinamikas[17].

* + 1. Mērķi un ierobežojumi

Vairāku aģentu pārklājuma ceļa plānošanā galvenais mērķis ir minimizēt zaudējuma funkciju, piemēram, kopējo laiku vai attālumu, nodrošinot tajā pašā laikā vides pārklājumu[18]. Šis mērķis var tikt formalizēts kā vairāku aģentu optimizācijas problēma, ņemot vērā dažādus ierobežojumus, piemēram, aģentu ierobežotas uztveres un vadības spējas, saziņas ierobežojumus un sadursmju novēršanas prasības[19].

Papildus galvenajam mērķim var ņemt vērā arī sekundāros mērķus, piemēram, enerģijas patēriņa minimizācija vai darba slodzes līdzsvarošana starp aģentiem. Šie sekundārie mērķi var tikt integrēti zudumu funkcijā, vai ar vairāku mērķu optimizācijas tehnikām[20].

* 1. Izaicinājumi vairāku aģentu pārklājuma ceļa plānošanā
     1. Mērogojamība

Jo aģentu skaits un vides lielums palielinās, jo palielinās vairāku aģentu pārklājuma ceļa plānošanas problēmas sarežģītība, kas padara algoritmu izstrādi, kas efektīvi spētu pārvaldīt lielus apjomus, grūtāku[21]. Mērogojamības problēmas var rasties sakarā ar iespējamo aģentu darbību un stāvokļu kombināciju skaita strauju palielināšanos, kā arī palielinātu mijiedarbību skaitu starp aģentiem. Mērogojami algoritmi, kas spēj apstrādāt lielu aģentu skaitu un sarežģītas vides, ir būtiska vairāku aģentu pārklājuma ceļa plānošanas pētniecības daļa.

* + 1. Koordinācija

Vairāku aģentu darbību koordinēšana ir liels izaicinājums vairāku aģentu pārklājuma ceļa plānošanā. Aģentiem ir jāiemācās sadarboties un izvairīties no konfliktiem, piemēram, sadursmēm vai liekiem pārklājumiem, strādājot kopā, lai sasniegtu pilnīgu un efektīvu vides pārklājumu[22]. Koordināciju var panākt, izmantojot dažādas pieejas, tostarp centralizētu lēmumu pieņemšanu, sadalītu lēmumu pieņemšanu vai abu kombināciju. Tomēr efektīvas koordinācijas stratēģijas izstrādāšana, kas līdzsvaro optimālu, skaitļošanas sarežģītību un izturību, joprojām ir nepārtraukta pētniecības problēma.

* + 1. Neskaidrība un dinamiskas vides

Iekštelpu vidēs var mēdz būt neskaidrībās un dinamiskās izmaiņās, piemēram, kustīgie šķēršļi, gaismas apstākļu variācijas vai sensoru trokšņi. Šie faktori var ietekmēt vairāku aģentu pārklājuma ceļa plānošanas algoritmu veiktspēju, tāpēc ir grūti izstrādāt izturīgas un pielāgojamas stratēģijas[23]. Domājot par neskaidrībām un dinamiskām izmaiņām, ir nepieciešams izstrādāt algoritmu, kas spētu efektīvi apstrādāt nepilnīgu vai trokšņainu informāciju un pielāgot savu uzvedību reāllaika novērojumu pamatā.

* + 1. Komunikācijas ierobežojumi

Vairāku aģentu pārklājuma ceļa plānošanā komunikācija starp aģentiem spēlē svarīgu lomu, lai nodrošinātu koordināciju un informācijas apmaiņu. Tomēr iekštelpu vides var būt pakļautas komunikācijas ierobežojumiem, piemēram, ierobežotam joslas platumam, aizkavei vai savienojamībai[24]. Šie ierobežojumi var ietekmēt vairāku aģentu pārklājuma ceļa plānošanas algoritmu veiktspēju, padarot grūtu efektīvu koordinācijas un informācijas apmaiņas uzturēšanu starp aģentiem. Algoritmu izstrāde, kas spēj pārvaldīt komunikācijas ierobežojumus un pielāgot savu uzvedību atkarībā no pieejamajiem komunikācijas resursiem, ir būtiska vairāku aģentu pārklājuma ceļa plānošanas pētniecības daļa.

1. Pastiprināšanas mācīšanas pārskats
   1. Ievads pastiprināšanas mācīšanās
      1. Galvenie jēdzieni

Pastiprināšanas mācīšanās (RL) ir mašīnmācīšanās nozare, kas koncentrējas uz aģentu apmācību, lai pieņemtu lēmumus, balstoties uz viņu mijiedarbību ar vidi [25]. Galvenās RL problēmas sastāvdaļas ir aģents, stāvokļu telpa, darbību telpa, apbalvojuma funkcija un politika. Aģents apgūst optimālu politiku, kartējot stāvokļus uz darbībām, izpētot vidi un saņemot atgriezenisko saiti apbalvojuma vai soda veidā.

* + 1. Markova lēmumu procesi (MLP)

Būtisks RL jēdziens ir Markova lēmumu process (MLP), kas ir matemātiskais *framework*, ko izmanto, lai modelētu lēmumu pieņemšanas problēmas stohastiskās vidēs [26]. MLP tiek definēts ar kortežu (S, A, P, R, γ), kur S ir stāvokļu telpa, A ir darbību telpa, P ir pāreju modelis, R ir apbalvojuma funkcija, un γ ir atlaižu faktors. MLP mērķis ir atrast optimālu politiku, kas maksimizē sagaidāmo kumulatīvo atlaisto apbalvojumu.

* 1. Vienīgā aģenta pastiprināšanas mācīšanas algoritmi
     1. Uz modeli balstītas un no modeļa brīvas metodes

Pastiprināšanas mācīšanās metodes var iedalīt divās plašās kategorijās: uz modeli balstītās un no modeļa brīvas pieejas. Uz modeli balstītās metodes balstās uz skaidru vides dinamikas modeli, piemēram, pāreju modeli un apbalvojuma funkciju, lai plānotu darbības un atjaunotu aģenta zināšanas [27]. Pretēji tam, no modeļa brīvas metodes neprasa skaidras zināšanas par vides dinamiku un tā vietā apgūst tieši no aģenta mijiedarbības ar vidi.

* + 1. Uz vērtībām balstītas metodes

Uz vērtībām balstītas metodes apgūst optimālu vērtības funkciju, kas pēc tam tiek izmantota, lai izrietētu optimālu politiku. Q-mācīšanās ir plaši izmantota no modeļa brīvā, uz vērtībām balstītā metode, kas novērtē darbības-vērtības funkciju Q(s, a) (Learning from delayed rewards, 1995). SARSA ir vēl viena vērtības balstīta metode, kas ir līdzīga Q-mācīšanās, galvenā atšķirība ir tā, ka SARSA ir *on-policy* metode, bet Q-mācīšanās ir *off-policy* [28].

* + 1. Uz politikām balstītas metodes

Uz politikām balstītas metodes tieši apgūst optimālu politiku, nenovērtējot vērtības funkciju. REINFORCE ir klasiska uz politikām balstīta metode, kas izmanto politikas gradientu, lai atjauninātu politikas parametrus [29]. Ir piedāvātas uzlabotas politikas balstītas metodes, piemēram, *Trust Region Policy Optimization* (TRPO) un *Proximal Policy Optimization* (PPO), lai uzlabotu politikas gradientu metožu stabilitāti un efektivitāti[30], [31].

* + 1. Aktiera – kritiķa metodes

Aktiera-kritiķa metodes apvieno gan uz vērtībām balstītas, gan uz politikām balstītas metodes, izmantojot vērtības funkciju (kritiķi), lai novērtētu darbības-vērtības funkciju Q(s, a), un atsevišķu politikas funkciju (aktieri), lai izvēlētos darbības. Asinhronās priekšrocības aktiera-kritiķa (A2C) un tā paralēlā versija A3C ir populāras aktiera-kritiķa metodes [32]. Citas uzlabotas aktiera-kritiķa metodes ietver *Deep Deterministic Policy Gradient* (DDPG) nepārtrauktām darbību telpām, *Twin Delayed DDPG* (TD3) un *Soft Actor-Critic* (SAC) [33]–[35].

* 1. Izpētes un izmantošanas kompromiss
     1. Epsilon – *greedy*

Epsilon-mantkārīgs ir vienkārša izpētes stratēģija, kurā aģents ar varbūtību ε izvēlas nejaušu darbību un ar varbūtību 1-ε izvēlas darbību ar augstāko novērtēto vērtību.

* + 1. Augšējas uzticības robeža

UCB ir sarežģītāka izpētes stratēģija, kas līdzsvaro izpēti un izmantošanu, ņemot vērā gan darbības novērtēto vērtību, gan tās nenoteiktību [36]. Aģents izvēlas darbības, balstoties uz augšējām uzticības robežām, kas sniedz optimistisku novērtējumu par darbības potenciālo vērtību.

* + 1. Tompsona izlase

Tompsona izlase ir vēl viena izpētes stratēģija, kas risina izpētes un izmantošanas kompromisu, uzturot *Bayesian posterior* sadalījumu pār darbības-vērtības funkciju un izvēloties darbības atbilstoši to optimālas iespējas varbūtībai [37]. Šī pieeja ļauj aģentam adaptīvi līdzsvarot izpēti un izmantošanu, balstoties uz nenoteiktību tā novērtējumos.

1. Vairāku aģentu pastiprināšanas mācīšanas
   1. Vairāku aģentu pastiprināšanas mācīšanas pārskats
      1. Problēmas vairāku aģentu pastiprināšanas mācībās

Vairāku aģentu pastiprināšanas mācīšanās (MARL) paplašina pastiprināšanas mācīšanos līdz vides iestatījumiem ar vairākiem aģentiem, ieviešot jaunas problēmas, piemēram, koordināciju, konkurenci un daļēju novērojamību [38]. MARL aģentiem ir jāapgūst pielāgoties citu aģentu stratēģijām, vienlaikus apgūstot savas optimālās politikas. Mērogojamībai un stabilitātei ir svarīga nozīme MARL algoritmu izstrādē.

* + 1. Kooperatīvie, konkurējošie un jaukti scenāriji

Vairāku aģentu iestatījumos aģenti var piedalīties kooperatīvos, konkurējošos vai jauktos (kooperatīvi-konkurējošos) scenārijos [39]. Kooperatīvie scenāriji ietver aģentu kopīgu darbu, lai sasniegtu kopēju mērķi, bet konkurentu scenāriji ietver aģentus, kas cenšas maksimizēt savus individuālos apbalvojumus, palielinot citu aģentu zudumus. Sajaukti scenāriji ietver gan kooperatīvus, gan konkurentus aspektus, kur aģentiem ir individuāli mērķi, bet vienlaikus ir jāsadarbojas, lai sasniegtu kopēju mērķi.

* 1. Vairāku aģentu mācīšanas *frameworks*
     1. Neatkarīga Q-mācīšanās (IQL)

IQL ir vienkāršs viena aģenta Q-mācīšanās paplašinājums vairāku aģentu iestatījumiem [40]. Katrs aģents mācās neatkarīgi, izmantojot Q-mācīšanos, ignorējot citu aģentu klātbūtni. Lai gan ir vienkārši īstenot, IQL saskaras ar nestabilitāti un nesakrišanas problēmām, kas izraisa vides nestacionaritāti, ko izraisa citu aģentu vienlaikus notiekošā mācīšanās.

* + 1. Kopīgas darbības mācīšanās (JAL)

JAL ir pieeja, kas modelē visu aģentu kopīgo darbību telpu un apgūst centralizētu Q-funkciju [41]. Tomēr JAL mērogojamībai ir ierobežojumi, ko izraisa kopīgās darbības telpas eksponenciālais pieaugums, palielinoties aģentu skaitam.

* + 1. Koordinētā pastiprināšanās mācīšanās (CRL)

CRL metodes mērķis ir risināt koordinācijas problēmu, iekļaujot koordinācijas mehānismus mācīšanās procesā [42]. Šie mehānismi var būt izteikti, piemēram, komunikācijas kanāli vai kopīga zināšana, vai netieši, piemēram, apgūstot koordinētas darbības bez skaidras koordinācijas.

* 1. Komunikācija vairāku aģentu pastiprināšanas mācīšanās
     1. Ziņu nodošanas pieejas

Ziņu nodošanas pieejas ļauj aģentiem apmainīties ar informāciju mācīšanās procesa laikā, veicinot koordināciju un sadarbību [43]. Šīs pieejas var būt centralizētas, kur centrālais vadītājs pārvalda komunikāciju, vai decentralizētas, kur aģenti tieši sazinās viens ar otru.

* + 1. Diferencējamā starpaģentu mācīšanās (DIAL)

DIAL ir metode komunikācijas un koordinācijas end-to-end apguvei vairāku aģentu iestatījumos [44]. Tā izmanto diferencējamu komunikācijas kanālu, kas ļauj kopīgi apgūt komunikācijas un lēmumu pieņemšanas politikas, ļaujot aģentiem pielāgot savas komunikācijas stratēģijas, lai uzlabotu veiktspēju.

* + 1. Komunikācijas protokoli un arhitektūras

Vairāku aģentu pastiprināšanas mācīšanās jomā ir piedāvāti dažādi komunikācijas protokoli un arhitektūras, lai veicinātu informācijas apmaiņu un koordināciju [45]. Tajos ietilpst gan izteiktas komunikācijas shēmas, piemēram, ziņu nodošana un kopīga atmiņa, gan netiešas komunikācijas shēmas, piemēram, mācīšanās, vērojot citu aģentu darbības.

* 1. Centralizēta apmācība ar decentralizētu izpildi
     1. Pretfaktiskie vairāku aģentu politikas gradienti (COMA)

COMA ir centralizētas apmācības pieeja, kas izmanto centralizētu kritiķi, lai novērtētu pretfaktiskās priekšrocības katram aģentam, ļaujot stabilai decentralizētu politiku apgūšanai [46]. Decentralizētās politikas tiek izpildītas neatkarīgi katram aģentam laikā, kad tās tiek izvietotas.

* + 1. QMIX un VDN

QMIX un vērtību dekompozīcijas tīkli (VDN) ir metodes decentralizētu politiku apgūšanai ar centralizētu apmācību [47], [48]. Tie izmanto centralizētu vērtības funkciju, kas ir dekomponējama individuālās aģentu vērtības funkcijās, ļaujot decentralizētai izpildei, izmantojot centralizētās apmācības priekšrocības.

1. Iekštelpas vides procedurāla ģenerēšana

Procedurālā iekštelpu vides ģenerēšana ir tehnika, kas ļauj virtuāli veidot telpas un vides. Šī tehnika izmanto algoritmus un noteikumus, lai automātiski ģenerētu nejaušo telpas plānojumu, izmantojot programmatūru. Rezultējošās telpas var tikt izmantotas video spēlēs, simulācijās un citās interaktīvās lietojumprogrammās. Procedurāli ģenerēto vižu labums ceļa meklēšanas algoritmu testēšanai ir tas, ka procedurāli ģenerētas vides piedāvā lielāku variāciju telpu savstarpējam novietojumam, un šķēršļu novietojumam. Algoritmu testēšana telpās, kas pastāv reālajā dzīvē, ierobežo apstākļu variāciju, jo lielākoties dzīvokļu plānojumi balstās uz vienādiem arhitektūras un dizaina principiem, secīgi – tiem bieži ir līdzīgas īpašības. Savukārt robota efektivitāti neietekmē cik skaisti, vai ērti ir izvietotas mēbeles istabā, to ietekmē tikai iespēja precīzi un kvalitatīvi plānot savu ceļu jebkurā vidē. Šajā darbā, procedurālo vides ģenerēšanu var sadalīt trijos būtiskajos posmos: vides skeleta veidošana, durvju izvietošana un mēbeļu izvietošana.

* 1. Vides skeleta veidošana

Vides skelets, šī darba kontekstā ir primitīva vide, kas sastāv no vairākām ortogonālām istabām, kurām ir noteikti izmēri, izvietojums telpā un tips. Vides skeleta būvēšana ir būtiskākais solis procedurālā ģenerēšanā, jo tas nosaka kopējo vides formu un vides kopējo platību. Šajā algoritmā katra istaba ir istabas (*Room*) klases objekts, kas glabā informāciju par istabas izmēriem, istabas centra koordinātēm, istabas stūru koordinātēm, istabas tipu un istabas identifikācijas numuru. Jaunas istabas būvēšana sākās ar nejaušo istabas tipa izvēli, piemēram “virtuve” (*kitchen*). Pēc tipa izvēlēšanas, algoritms nejauši izvēlās jaunas istabas izmērus, ar ierobežojumiem, kuri ir atkarīgi no istabas tipa. Zinot istabas izmērus, algoritms izmeklē visas potenciālas izvietošanas iespējas. Izvietošanai ir daži ierobežojumi: istaba nedrīkst pārklāt vai krustot citu istabu, neviens no istabas punktiem nedrīkst atrasties zem nulles koordinātes x vai z asī. Pēc visu potenciālu izvietošanas punktu atrašanas, uz nejauši izvēlēta punkta tiek būvēta jauna istaba. Process atkārtojās tikmēr nebūs izveidots nejauši izvēlētais istabu skaits.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | |  |  | |  |  | |
| 1.att. Procedurāli ģenerēto vides skeletu piemēri |

* 1. Durvju izvietojums

Nākamais solis vides ģenerēšanā ir durvju ģenerēšana. Darba mērķim nav vajadzīgas fiziskas atveramas durvis, tāpēc to vietā ir noteikta platuma caurumi sienās, kas ļaus aģentam brīvi pārvietoties starp visām istabām. Lai sasniegtu visu istabu pieejamību, tiek veidots plānojumu reprezentējošs grafs, kur katrs grafa mezgls reprezentē vienu no istabām. Iegūtam grafam tiek atrasts mazākais pārklājošais koks (*minimum spanning tree*), lai noteiktu minimālu durvju skaitu un izvietojumu, kas nodrošinās iespēju aģentam iziet cauri visām ģenerētām istabām. Turpmāk ar nelielu varbūtību tiek ģenerētas vel dažas durvis, ar priekšroku lielākām istabām, lai nodrošinātu lielāku variāciju..

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 2.att. durvju izvietošanas algoritma darbības princips pa soļiem | | | |

|  |  |
| --- | --- |
|  |  |
| 3.att. Procedurāli ģenerēto vižu piemēri ar durvīm | |

* 1. Mēbeļu un šķēršļu ģenerēšanā

Reālajā dzīvē dzīvokļos un mājās vienmēr ir mēbeles. Mēbeles kalpo dažādiem nolūkiem, bet tīrīšanas robotiem tie pirmkārt ir šķērslī, kas ir jāņem vērā plānojot apiešanas maršrutu. Lai pilnās apiešanas algoritmu testēšanas rezultāti būtu pilnvērtīgi, arī simulācijā ir jāiekļauj mēbeļu ģenerēšana. Lai izvietot mēbeles uzticami, ir jāņem vērā vairāki noteikumi. Pirmkārt, katrā istabā var atrasties tikai konkrēta tipa mēbelēs, piemēram guļamistabā obligāti jābūt gultai, bet virtuvē nevajadzētu būt tualetei. Istabas tips tiek noteikts vel pirmajā solī, skeleta ģenerēšanas laikā. “Sēklas”, kas tika nejauši ģenerētas ir objekti, kas satur informāciju arī par istabas tipu. Ģenerējot mēbeles algoritms pirmkārt atsaucās uz to informāciju, lai veidotu zināšanu bāzi: kādām mēbelēm obligāti jābūt istabā, kādas mēbeles var būt istabā un kādas mēbeles ģenerēt nedrīkst noteiktajā istabā.

Otrkārt, ir jāņem vērā arī savstarpējo novietojumu mēbelēm. Piemēram mēbeles nedrīkst stāvēt durvīm priekšā, jo tādā gadījuma aģents nevarēs izbraukt tām cauri. Turpmāk, mēbeles ir jāizvieto tā, lai nerādītu pārāk šauras vietas arī istabas iekšā, pretējā gadījumā testēšanas vidē būs nesasniedzami apgabali, un testa rezultātus nevarēs uzskatīt par pilnvērtīgiem. Turklāt jāņem vērā arī mēbeļu izvietošanas loģiskumu. Piemēram ja dzīvojamā istabā ir televizors, tad dīvānam jābūt orientētam pret televizoru.

Atsauksmes

[1] H. Moravec and A. Elfes, “High resolution maps from wide angle sonar,” in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, Institute of Electrical and Electronics Engineers, pp. 116–121. doi: 10.1109/ROBOT.1985.1087316.

[2] H. Choset and P. Pignon, “Coverage Path Planning: The Boustrophedon Cellular Decomposition,” in *Field and Service Robotics*, London: Springer London, 1998, pp. 216–222. doi: 10.1007/978-1-4471-1273-0\_32.

[3] E. U. Acar and H. Choset, “Sensor-based Coverage of Unknown Environments: Incremental Construction of Morse Decompositions,” *Int J Rob Res*, vol. 21, pp. 345–366, 2002.

[4] A. Zelinsky, R. A. Jarvis, J. Byrne, and S. Yuta, “Planning Paths of Complete Coverage of an Unstructured Environment by a Mobile Robot,” 2007.

[5] W. H. Huang, “Optimal line-sweep-based decompositions for coverage algorithms,” *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, vol. 1, pp. 27–32 vol.1, 2001.

[6] Y. Gabriely and E. Rimon, “Spanning-tree based coverage of continuous areas by a mobile robot,” *Ann Math Artif Intell*, vol. 31, no. 1, pp. 77–98, 2001, doi: 10.1023/A:1016610507833.

[7] N. Hazon and G. A. Kaminka, “Redundancy, Efficiency and Robustness in Multi-Robot Coverage,” in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 735–741. doi: 10.1109/ROBOT.2005.1570205.

[8] Y. Engel, S. Mannor, and R. Meir, “Reinforcement learning with Gaussian processes,” in *ICML 2005 - Proceedings of the 22nd International Conference on Machine Learning*, 2005. doi: 10.1145/1102351.1102377.

[9] V. Mnih *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, 2015, doi: 10.1038/nature14236.

[10] V. R. Konda and J. N. Tsitsiklis, “On actor-critic algorithms,” *SIAM J Control Optim*, vol. 42, no. 4, 2003, doi: 10.1137/S0363012901385691.

[11] R. Coulom, “Efficient selectivity and backup operators in Monte-Carlo tree search,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2007. doi: 10.1007/978-3-540-75538-8\_7.

[12] L. Matignon, G. J. Laurent, and N. Le Fort-Piat, “Independent reinforcement learners in cooperative Markov games: A survey regarding coordination problems,” *Knowledge Engineering Review*, vol. 27, no. 1. 2012. doi: 10.1017/S0269888912000057.

[13] H. Nakashima, M. P. Wellman, G. Weiss, and P. Stone, “Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems,” in *Adaptive Agents and Multi-Agent Systems*, 2006.

[14] R. Lowe, Y. I. WU, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, “Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper\_files/paper/2017/file/68a9750337a418a86fe06c1991a1d64c-Paper.pdf

[15] J. N. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, “Counterfactual Multi-Agent Policy Gradients,” in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, in AAAI’18/IAAI’18/EAAI’18. AAAI Press, 2018.

[16] E. Galceran and M. Carreras, “A survey on coverage path planning for robotics,” *Rob Auton Syst*, vol. 61, no. 12, pp. 1258–1276, 2013, doi: https://doi.org/10.1016/j.robot.2013.09.004.

[17] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[18] H. Choset, “Coverage for robotics - A survey of recent results,” *Ann Math Artif Intell*, vol. 31, no. 1–4, 2001, doi: 10.1023/A:1016639210559.

[19] L. M. Rekleitis, G. Dudek, and E. E. Milios, “Multi-robot exploration of an unknown environment, efficiently reducing the odometry error,” in *IJCAI International Joint Conference on Artificial Intelligence*, 1997.

[20] M. J. Matarić, “Reinforcement Learning in the Multi-Robot Domain,” *Auton Robots*, vol. 4, no. 1, 1997, doi: 10.1023/A:1008819414322.

[21] L. E. Parker, “ALLIANCE: an architecture for fault tolerant multirobot cooperation,” *IEEE Transactions on Robotics and Automation*, vol. 14, no. 2, pp. 220–240, 1998, doi: 10.1109/70.681242.

[22] R. Zlot, A. T. Stentz, M. B. Dias, and S. Thayer, “Multi-robot exploration controlled by a market economy,” in *Proceedings - IEEE International Conference on Robotics and Automation*, 2002. doi: 10.1109/robot.2002.1013690.

[23] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, “Collaborative multi-robot exploration. In: Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065).Vol 1. IEEE; 2000,” *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 1, no. February, 2000.

[24] M. A. Batalin and G. S. Sukhatme, “Spreading Out: A Local Approach to Multi-robot Coverage,” in *Distributed Autonomous Robotic Systems 5*, 2002. doi: 10.1007/978-4-431-65941-9\_37.

[25] R. S. Sutton and A. G. Barto, “Reinforcement Learning: An Introduction. Cambridge: MIT Press.,” *MA: MIT Press.[Google Scholar]*, 1998.

[26] A. Feinberg, “Markov Decision Processes: Discrete Stochastic Dynamic Programming (Martin L. Puterman),” *SIAM Review*, vol. 38, no. 4, 1996, doi: 10.1137/1038137.

[27] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, “Planning and acting in partially observable stochastic domains,” *Artif Intell*, vol. 101, no. 1–2, 1998, doi: 10.1016/s0004-3702(98)00023-x.

[28] G. A. Rummery and M. Niranjan, “On-line q-learning using connectionist systems cued/f-infeng/tr 166,” *Update*, no. September, 1994.

[29] R. J. Williams, “Simple statistical gradient-following algorithms for connectionist reinforcement learning,” *Mach Learn*, vol. 8, no. 3–4, 1992, doi: 10.1007/bf00992696.

[30] J. Schulman, S. Levine, P. Moritz, M. Jordan, and P. Abbeel, “Trust region policy optimization,” in *32nd International Conference on Machine Learning, ICML 2015*, 2015.

[31] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal Policy Optimization Algorithms,” Jul. 2017.

[32] V. Mnih *et al.*, “Asynchronous methods for deep reinforcement learning,” in *33rd International Conference on Machine Learning, ICML 2016*, 2016.

[33] T. P. Lillicrap *et al.*, “Continuous control with deep reinforcement learning,” Sep. 2015.

[34] S. Fujimoto, H. Van Hoof, and D. Meger, “Addressing Function Approximation Error in Actor-Critic Methods,” in *35th International Conference on Machine Learning, ICML 2018*, 2018.

[35] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in *35th International Conference on Machine Learning, ICML 2018*, 2018.

[36] P. Auer, N. Cesa-Bianchi, and P. Fischer, “Finite-time analysis of the multiarmed bandit problem,” *Mach Learn*, vol. 47, no. 2–3, 2002, doi: 10.1023/A:1013689704352.

[37] D. Russo, B. Van Roy, A. Kazerouni, I. Osband, and Z. Wen, “A Tutorial on Thompson Sampling,” pp. 1–96, Jul. 2017.

[38] L. Busoniu, R. Babuska, and B. De Schutter, “A Comprehensive Survey of Multiagent Reinforcement Learning,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 2, pp. 156–172, 2008, doi: 10.1109/TSMCC.2007.913919.

[39] L. Panait and S. Luke, “Cooperative multi-agent learning: The state of the art,” *Auton Agent Multi Agent Syst*, vol. 11, no. 3, 2005, doi: 10.1007/s10458-005-2631-2.

[40] M. Tan, “Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents,” in *Machine Learning Proceedings 1993*, 1993. doi: 10.1016/b978-1-55860-307-3.50049-6.

[41] M. L. Littman, “Markov games as a framework for multi-agent reinforcement learning,” in *Machine Learning Proceedings 1994*, 1994. doi: 10.1016/b978-1-55860-335-6.50027-1.

[42] C. Guestrin, M. G. Lagoudakis, and R. E. Parr, “Coordinated Reinforcement Learning,” in *International Conference on Machine Learning*, 2002.

[43] K. Zhang, Z. Yang, H. Liu, T. Zhang, and T. Başar, “Fully decentralized multi-agent reinforcement learning with networked agents,” in *35th International Conference on Machine Learning, ICML 2018*, 2018.

[44] J. N. Foerster, Y. M. Assael, N. De Freitas, and S. Whiteson, “Learning to communicate with deep multi-agent reinforcement learning,” in *Advances in Neural Information Processing Systems*, 2016.

[45] S. Sukhbaatar, A. Szlam, and R. Fergus, “Learning multiagent communication with backpropagation,” in *Advances in Neural Information Processing Systems*, 2016.

[46] J. Foerster *et al.*, “Stabilising experience replay for deep multi-agent reinforcement learning,” in *34th International Conference on Machine Learning, ICML 2017*, 2017.

[47] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. Foerster, and S. Whiteson, “QMIX: Monotonic value function factorisation for deep multi-agent reinforcement Learning,” in *35th International Conference on Machine Learning, ICML 2018*, 2018.

[48] P. Sunehag *et al.*, “Value-decomposition networks for cooperative multi-agent learning based on team reward,” in *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, 2018.