Egy- és multiágenses Megerősítéses Tanulás összehasonlítása HighwayEnv környezetben

Gujgiczer Dániel Tamás* Szabó Ádám*
Bécsi Tamás*

**Közlekedés- és Járműirányítási Tanszék, Budapesti Műszaki és Gazdaságtudományi Egyetem*
(e-mail: gujgiczerd@edu.bme.hu, szabo.adam@kjk.bme.hu, becsi.tamas@kjk.bme.hu)

Kivonat: A hagyományos egyágenses Megerősítéses Tanulás széles körben elterjedt módszerré vált számos közlekedési probléma megoldására. Ezek az algoritmusok ígéretes eredményeket mutatnak a vizsgált környezetekben, azonban az egyedül tanított ágensek teljesítménye más ágensekkel való találkozáskor megkérdőjelezhető. A publikáció célja, hogy bemutassa a multiágenses Megerősítéses Tanulás előnyeit az autonóm járműirányításban, mely egy egyszerű, többsávos autópályát szimuláló környezetben történik. Az egy- és multiágenses tanulási stratégiák vegyes forgalomban kerülnek összevetésre. Emellett megvizsgáljuk az ágensek számára visszacsatolt állapotreprezentáció összetételének hatását.

1. BEVEZETÉS

A Google Deepmind által létrehozott AlphaGo Zero (Mnih et al., 2013), (Mnih et al., 2015) sikereit követően a mély megerősítéses tanulás egyre népszerűbb kutatási témakörré vált. Azóta számos különböző esetben alkalmazták sikeresen, például társasjátékok (Silver et al., 2017), videójátékok (Berner et al., 2019) és a robotika (Lillicrap et al., 2019) területén. A felhasználási területek bővüléséhez nagyban hozzájárult a módszer képessége a komplex problémák, például szekvenciális döntési folyamatok megoldására, valamint a számítástechnika fejlődése. Ennek következményeképp egyre több kutató hasznosítja a mély megerősítéses tanulás adta lehetőségeket a közúti közlekedés területén (Farazi et al., 2020), mely magába foglalja az autonóm vezetést (Aradi, 2020), a közlekedési jelzőlámpák vezérlését (Kővári et al., 2022), az energiahatékony vezetést (Wu et al., 2019), valamint az optimális útvonaltervezést (Bello et al., 2017).

Habár léteznek end-to-end keretrendszerek az autonóm járműirányítás (Ly & Akhloufi, 2020) kihívásainak leküzdésére, a kutatók nagy része a feladatot moduláris rendszerekkel kívánja megoldani és az egyes részfeladatokra koncentrál. Ezek közé tartozik az útvonaltervezés, a környezetérzékelés (Farooq et al., 2023), a lokális trajektóriatervezés (Moghadam et al., 2021), valamint az alacsony szintű szabályozás.

Míg a mozgástervezés alsó rétegei függetlenek a közlekedés többi résztvevőjétől, a magasabb szintű stratégiai döntések során figyelembe kell venni a többi járművet is. Ezek modellezése gyakran különböző járműkövetési modelleken alapul, ilyen például az „Intelligent Driver Model” (IDM) (Treiber et al., 2000).

(Li et al., 2015) egy intelligens előzési döntéshozatali módszert mutat be autópályán történő autonóm járműirányítás esetére Q-tanulás használatával. Az ismertetett eredmények alapján az algoritmus felülmúlja a hagyományos döntéshozatali stratégiákat. Sűrű forgalomban az ágensnek kezelni kell a változó számú közeli járműveket. Ennek kezelésére (Leurent & Mercat, 2019) egy „attention-based” architektúrát mutat be, mellyel az ágens performanciája nagymértékben javul, valamint segíti az interakciók vizualizációját. Ahogy a mélytanulási algoritmusok komplexitása növekszik, egyre inkább szükségessé válik, hogy megértsük a döntéshozatalukat a sikeres felhasználásuk érdekében, ezért (Bello et al., 2017) egy új keretrendszert javasol azok vizsgálatára.

Az egyágenses algoritmusok adott helyzetekben a saját jutalmuk maximalizálására törekednek a tanulás során, mely akár önző, akár altruista viselkedéshez is vezethet a jutalmazási stratégia függvényében. Azonban valós forgalmi helyzetekben az ágensnek nem egy determinisztikus környezetben kell boldogulnia. Ennek oka egyrészt, hogy a valóságban a rendszer állapotai rendszerint csak valamilyen bizonytalansággal mérhetők. Ennek a leírására a részlegesen megfigyelhető Markov döntési folyamatok (POMDP) használhatók. Ezen kívül a forgalom résztvevőinek a viselkedése is különféle lehet. Ezt különböző paraméterezéssel ellátott járművezető modellekkel (Mavrogiannis et al., 2022) lehet figyelembe venni. A bizonytalanság jöhet továbbá a forgalomban részt vevő más ágensektől, aminek a kezelésre az egyágenses algoritmusok nincsenek felkészítve. Ennek megoldására használható a multiágens megerősítéses tanulás (MARL), ahol több ágens egyidejű tanításával vegyesen vehetnek részt ágensek és járművezető modellek által irányított járművek a forgalomban. Egy skálázható MARL keretrendszert mutat be (Chen et al., 2022), ahol az ágensek feladata a gyorsítósávról történő besorolás segítése. (Kaushik et al., 2018) egy újszerű MARL architektúrát hoz létre, mely egyidejűleg több vezetési magatartást képes megtanulni. Az

algoritmus az ágensek „parameter-sharing” segítségével az eredmények gyorsabb konvergenciához vezet.

Ebben a publikációban az egy- és multiágenses megerősítéses tanulás autópályás vezetési környezetben elért eredményei kerülnek összevetésre. Multiágens tanulás esetén a tanítás különböző ágensszámokkal valósult meg, miközben a környezetben szereplő járművek száma változatlan maradt. Mivel minden ágens feladata ugyanaz, ezért a tanítás „parameter-sharing” használatával történt. A betanított ágensek viselkedése különböző autópályás haladási szituációkban került kiértékelésre. Mindemellett a különböző állapottér összetételének ágensek közti, közvetlen kommunikáció nélküli együttműködésre gyakorolt hatása is vizsgálatra került.

A publikáció felépítése a következőképp alakul. A 2. fejezetben a felhasznált algoritmusok és hiperparamétereik kerülnek ismertetésre. A 3. fejezet a szimulációs környezetet mutatja be. A 4. fejezetben a tanítás tulajdonságai kerülnek ismertetésre. Az 5. fejezetben a szimuláció eredményei és az algoritmus teljesítménye kerülnek kiértékelésre. Végül a 6. fejezet a következtetések levonása mellett a jövőbeni lehetőségekre tartalmaz kitekintést.

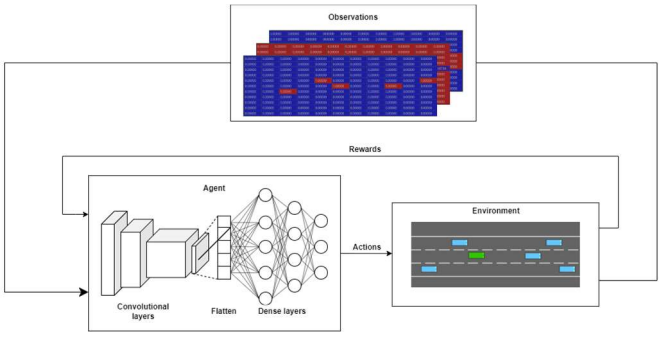
2. ALGORITMUSOK

A megerősítéses tanulás a három fő gépi tanulási módszer egyike a felügyelt tanulás és a felügyelet nélküli tanulás mellett. A felügyelt tanulással ellentétben, ebben az esetben nincs szükség címkézett bemenet-kimenet párokra. A megerősítéses tanulás ágensei a környezettel való folytonos kölcsönhatás során tanulnak, mely egy Markov döntési folyamatként írható le, egy rendezett lista formájában: $\{S, A, P, R\}$, ahol S a környezet állapota, A az ágens számára elérhető lehetséges beavatkozások vektora, P az állapotváltozás valószínűsége, valamint R a jutalomfüggvény. Az ágens célja az optimális „policy” meghatározása, mellyel a kumulatív jutalmát maximalizálja:

$$G = \sum_{t=1}^{T} \gamma^t r_t \tag{1}$$

ahol γ az úgynevezett „discount factor”, ami meghatározza a jelen döntéseinek hatását a jövőbeni jutalmakra, és r_t a jutalom t időpillanatban.

Az ágens minden lépésnél megkapja a környezet aktuális állapotát leíró s_i reprezentációt, ami alapján a π_i „policy” függvényének megfelelően kiválaszt egy a_i beavatkozást, amit elküld a környezetnek. A környezet végrehajtja a beavatkozást, ami egy $P(s_i, a_i | s_{i+1})$ állapotváltozást eredményez. A választott beavatkozás minőségét a jutalom foglalja magába, melyet az ágens az optimális „policy” megtalálására használ. A tanulási folyamatot az 1. ábra mutatja be.



1. ábra: Az ágens és a környezet közti kölcsönhatás a megerősítéses tanulásban

2.1 Double Deep Q Network

Az úgynevezett Double Deep Q Network (DDQN) a népszerű Deep Q Network (DQN) algoritmus hatékonyabb változata. Mind a DQN, mind a DDQN „value-based” algoritmusok, ahol a neurális hálózatot függvényapproximátor a „value” függvény meghatározására használják. A DQN algoritmus hajlamos némileg túlbecsülni a Q-értékeket, ezt hivatott kompenzálni a DDQN algoritmus, amihez két neurális hálózatot használ. A beavatkozás kiválasztása az online hálózat alapján történik, az elérhető jutalom viszont a „target” hálózat segítségével kerül meghatározásra. Így a Bellman egyenlet kis mértékben módosul:

$$Q(s_t, a_t; \theta_t) = r_{t+1} + \gamma Q\left(s_{t+1}, \underset{a}{\operatorname{argmax}}\, Q(s_{t+1}, a_t; \theta_t)\,;\, \theta_t^-\right) \tag{2}$$

ahol $Q(s, a)$ megadja, hogy az a beavatkozást választva az s állapotban, mekkora jutalom érhető el az epizód végéig. θ_t az online neurális háló súlyait, valamint θ_t^- a „target” hálózat súlyait tartalmazza t időpillanatban.

2.2 Multi-ágens megerősítéses tanulás

Amikor a környezetben egyszerre több tanuló egység is szerepel, akkor a folyamat „Markov game” formában írható le (Littman, 1994). Több ágens jelenléte a környezetben számos problémához vezethet, mint a nem stacioner környezet, „shadowed equilibrium”, részleges megfigyelhetőség és jutalmazás kérdésköre. (Gronauer & Diepold, 2022)

Az ágens által megvalósítani kívánt viselkedés alapján a közúti helyzetekre tekinthetünk kooperatív és nem kooperatív játékként. Míg a jutalmazási stratégia megválasztása egyértelműnek tűnhet, a jutalom elosztása az ágensek között egy összetett probléma. Például kooperatív játék esetén az egységes jutalmazás eredményezhet egy önzetlen, kooperatív stratégiát, azonban az úgynevezett. “lazy agent” problémához is vezethet (Sunehag et al., 2017).

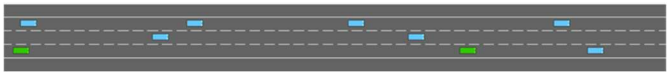
Multiágens megerősítéses tanulás esetén három fő tanítási séma különböztethető meg. Az első a centralizált megközelítés, mely során egyetlen neurális hálózat kezeli az összes ágens megfigyeléseit és beavatkozásait. Ezzel a módszerrel kis ágensszám esetén jó eredmények érhetőek el,

azonban skálázhatósági problémák léphetnek fel. Decentralizált tanulás esetén minden ágens saját „policy”-vel rendelkezik, melyek párhuzamosan kerülnek tanításra. Ezáltal a tanulási ideje folyamatosan nő, ahogy az ágensek egyre inkább alkalmazkodnak egymáshoz. A harmadik megközelítés a „parameter-sharing”, ahol az ágensek egy közös „policy”-t fejlesztenek. Ennek köszönhetően kevésbé erőforrás-igényes, valamint kézenfekvő megoldás, ha minden ágens számára ugyanaz a feladat. Ezen kívül az ágensek teljesítménye tovább javítható kommunikáció segítségével. Ily módon az ágensek megoszthatják az egyéni megfigyeléseiket és szándékaikat a többi ágenssel.

3. FELHASZNÁLT KÖRNYEZET

3.1 Autópálya környezet

A szimulációk során a „HighwayEnv” (Leurent, 2018) környezetet használtuk. A környezet többféle közlekedési szituációt (pl.: autópálya, kereszteződés, körforgalom) tartalmaz az autonóm járműirányításban történő döntéshozatali feladatok modellezésére. A kutatás során a „highway” környezetet használtuk, mely egy többsávos autópályás forgalmi szituációt ír le. A 2. ábrán a környezet grafikus megjelenítése látható, melyen a zöld színnel jelöljük az ágensek által, míg kékkel a kiválasztott járművezető modellek által irányított járműveket.



2. ábra: A környezet vizualizációja

Az ágens feladata a beállított referenciasebesség tartása, valamint a többi járművel való ütközés elkerülése. Ezen felül a jobbra tartás is figyelembe vett szempont. A jutalomfüggvény ezekből az összetevőkből áll össze. A sebesség alapú jutalom egy bizonyos sebességtartományon belül kerül kiosztásra, lineárisan leképezve nulla és a maximális jutalom között. Ez a következőképp írható le:

$$R_s = \begin{cases} \frac{v - v_{min}}{v_{max} - v_{min}} R_{s_{max}}, & ha\ v_{min} < v \leq v_{max} \\ 0, & egyébként \end{cases} \quad (3)$$

ahol R_s a sebesség alapú jutalom, v az irányított jármű adott pillanatbeli sebessége, v_{min} és v_{max} a jutalmazás sebességhatárai, valamint $R_{s_{max}}$ a maximális sebességnél elérhető jutalom.

Más járművekkel történő ütközés esetén az ágens büntetést kap, valamint terminálju az epizódot:

$$R_c = \begin{cases} R_{crashed}, & ha\ \text{ütközött} \\ 0, & egyébként \end{cases} \quad (4)$$

ahol R_c az ütközési jutalomrész és R_{crash} a büntetés értéke.

A jobbra tartás jutalma R_l szintén lineáris leképezéssel áll elő az alapján, hogy a jármű melyik sávban halad. Ebben az esetben tehát a leginkább jobboldali sáv jelenti a maximális, míg a legbelső sáv a 0 jutalmat.

A jutalomfüggvényben ezek a részek összegződnek, majd 0 és 1 között kerülnek normalizálásra:

$$r = R_s + R_c + R_l \quad (5)$$

$$R = \frac{r - R_c}{R_{s_{max}} + R_{l_{max}} - R_c} \quad (6)$$

Fontos megjegyezni, hogy az ágensek a magasabb szintű döntéshozatalt végzik, melyeket egy alacsonyabb szintű arányos (P) hosszirányú és arányos-derivatív (PD) keresztirányú szabályozó követ. A járművek kinematikáját a Kinematikai Kerékpármodell (Polack et al., 2017) írja le. A nem ágens által irányított járművek viselkedését az IDM modell, míg a keresztirányú viselkedést a “Minimizing Overall Braking Induced by Lane change” (MOBIL) modell írja le (Kesting et al., 2006). A környezet állapotát leíró reprezentációnak az „Occupancy Grid”-et választottuk, mely egy 3D mátrixként fogható fel. A jármű körüli kétdimenziós teret cellákra osztja, így az irányított jármű mindig közepén található. A mátrix harmadik dimenzióját a megfigyelt jellemzők száma határozza meg, melyek így külön csatornákon vannak eltárolva. Az állapotreprezentáció elemei az ágenshez képesti relatív értékeket tartalmazzák, valamint 0 és 1 közé vannak normalizálva. A választott reprezentáció előnye, hogy annak mérete független a közelben található járművek számától.

Mivel az ágensek csak a véghezvinni kívánt beavatkozásról döntenek, a választott beavatkozás csak diszkrét értéket vehet fel, melyek a balra és jobbra történő sávváltás, a gyorsítás vagy lassítás, valamint a sebesség- és sávtartás.

3.2 Módosítások

A környezet alapvetően támogatja a MARL algoritmusok használatát, ugyanakkor ez a beállítás még nem minden részkörnyezet esetén elérhető. Ezért a „highwayenv” módosítására volt szükség, mely során az állapotreprezentáció, a választható beavatkozások, valamint a jutalmak számítása úgy változott, hogy az egyidejűleg több ágens kezelésére is alkalmas legyen. Mindez az egyes járművek megfigyeléseinek és beavatkozásainak vektorba rendezésével valósult meg. Mivel az ágensek száma határozza meg e vektorok hosszát, egyágenses tanításra is alkalmazható. A tanítási folyamat felgyorsításának érdekében a környezet egyik gyorsabb verziója, a „highway-fast” került felhasználásra. Ez a szimuláció frekvenciáját és a környezetben szereplő járművek számát csökkenti, valamint az epizódok hosszát is. Emellett a nem ágens által irányított járművek közti ütközések sem kerülnek figyelembevételre.

4. A TANÍTÁS FOLYAMATA

A környezet konfigurálása után egy DDQN ágens került implementálásra. Figyelembe véve a választott állapot reprezentációt, a feladat megoldásához konvolúciós neurális hálózatot használunk. Az ágens és a hálózat hiperparamétereit az 1. Táblázat tartalmazza.

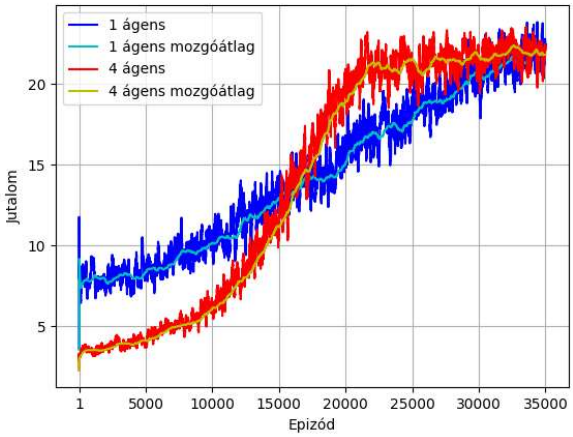
1. Táblázat – A tanításhoz használt hiperparaméterek

Paraméter	Érték
Learning rate	0,0001
Discount factor	0,99
Epsilon decay	0,999995
Epsilon min	0,01
Batch size	128
Burn-in	256
Sync every	20
Rejtett konvolúciós rétegek száma	3
Szűrők száma (rétegenként)	32, 128, 2
Rejtett teljesen összekötött rétegek	3
Neuronok száma (rétegenként)	4096, 256, 128
Aktivációs függvény	ReLU
Sebesség jutalmazási tartomány	[20, 30]
Sebességi jutalom	0,4
Ütközési büntetés	-1
Jobbra tartási jutalom	0,1

A tanítások egy három sávos, 20 járművet tartalmazó környezetben történtek. Multiágens esetben a járművek közül négyet irányított ágens. A tanítási folyamat, ahogy az a 3. ábrán is látható, 35000 epizódon keresztül tartott.

Az első iteráció során megfigyelt jellemzők a következők voltak:

- A jármű adott cellában való jelenlétét jelző indikátor $(presence)$
- Az utat és a környezetét elkülönítő indikátor (on_road)
- Az adott cellában található jármű hosszirányú sebessége (v_x)
- Az adott cellában található jármű keresztirányú sebessége (v_y)



3. ábra: A tanítási folyamat átlagos jutalomértékei

Ezután pedig a következő jellemzőkkel került bővítésre a megfigyelési tér:

- A jármű referencia irányszögének koszinusza $(\cos\gamma_{ref})$
- A jármű referencia irányszögének koszinusza $(\sin\gamma_{ref})$

A megfigyelési tér ily módosítása révén az irányított járművek a többi ágens pillanatnyi állapota mellett a jövőbeli szándékairól is információt nyertek.

A kutatás során a multiágens megerősítéses tanulás a „parameter-sharing” segítségével történt, teljesen kompetitív jutalmazással.

5. SZIMULÁCIÓS EREDMÉNYEK

A kutatás során először egy egyágenses algoritmust tanult be mindkét korábban említett állapottér esetére. Ezt követően a tanítás multiágens megerősítéses algoritmussal lett megismételve, egyidőben 4 ágenst irányítva.

A betanított hálózatok teljesítményét egy 100 epizódból álló teszt segítségével értékeltük ki. A tesztet mindegyik hálózat esetén megismételtük egy, négy, nyolc és tizenkettő irányított járművel, míg a teljes járműszám változatlan maradt. A vizsgált metrikák a következők voltak:

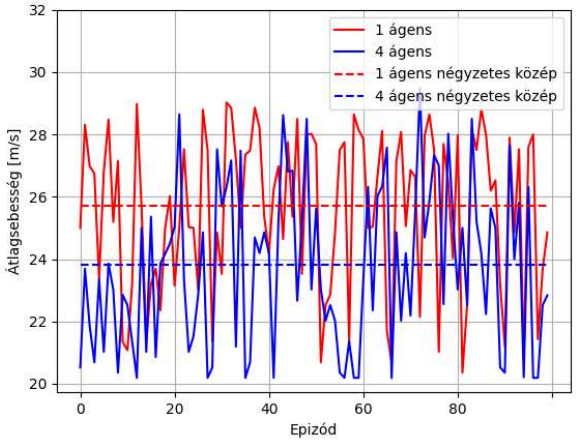
- Ütközések száma a 100 epizód során
- Az ágensek által irányított járművek áltagsebességei
- Átlagos jutalom

5.1 Az ágensszám hatása

Az eredeti állapottérrel végrehajtott tesztek eredményeit a 2. Táblázat és a 4. ábra mutatják be.

2. Táblázat – Teszteredmények az eredeti állapottér használatakor

Ágensek száma (kiértékelés)	Ágensek száma (tanítás)	Ütközések	Átlagos jutalom
1	1	5	24,56
	4	2	23,94
4	1	30	21,19
	4	14	23,39
8	1	55	18,82
	4	32	21,07
12	1	79	14,26
	4	46	19,92



4. ábra: Az első ágens átlagos sebességértékei az eredeti állapottér esetén

A 2. Táblázat alapján megállapítható, hogy a multiágens „policy” használatával lényegesen alacsonyabb ütközésszámot sikerült elérni megnövelt ágensszám esetén is. A robusztusabb viselkedés a magasabb átlagos jutalmak alapján is megállapítható. Az egyágenses algoritmus némileg magasabb átlagjutalmakat ér el abban az esetben, ha nincs a környezetben másik tanított ágens, azonban a sikertelen epizódok száma is magasabb ekkor. Ennek oka, hogy a sebesség alapú jutalom maximalizálása érdekében magasabb sebességet választ, ez azonban gyakrabban vezet ütközéshez. Ezzel szemben, a multiágens „policy” alacsonyabb átlagsebességet ér el, melynek eredményeképp csökken az ütközések száma, és így egyenletesebben oszlanak el a jutalmak az epizódok során. Emellett az is észrevehető, hogy a multiágens algoritmus teljesítménye is csökken az ágensszám növelésével, az egyágenses „policy” esetén azonban ez a csökkenés sokkal jelentősebb. Ezáltal az elért jutalmak közti különbség is növekvő.

5.2 A megfigyelt jellemzők hatása

Következő lépésként a kibővített állapottérrel tanított ágensek kerültek tesztelésre. Az eredményeket a 3. Táblázat és az 5. ábra mutatják be.

3. Táblázat – Teszteredmények a kibővített állapottér használatakor

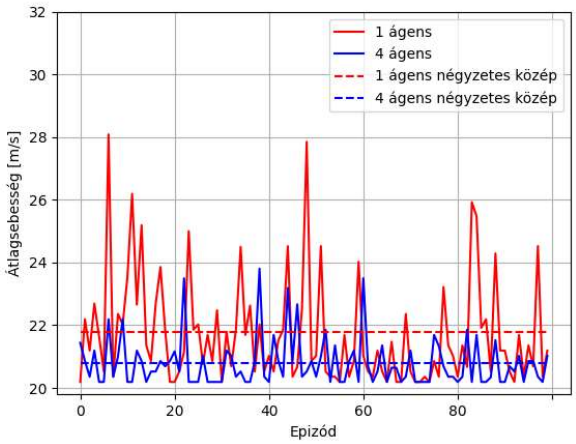
Ágensek száma (kiértékelés)	Ágensek száma (tanítás)	Ütközések	Átlagos jutalom
1	1	7	20,91
	4	0	22,33
4	1	15	20,32
	4	2	22,33
8	1	45	17,01
	4	16	21,02
12	1	79	12,42
	4	51	17,49

Az állapottér változtatásával mindkét algoritmus jobb eredményeket ért el mind a második, mind a harmadik tesztelési esetben. Ugyan némileg kisebb az átlagos jutalmuk, viszont figyelemre méltó javulás mutatkozik az ütközések számában. A kiegészítő információk által biztosított előny megszűnik, ha az ágensek számát egy bizonyos határérték fölé növeljük. Másrészről, az első (egy ágenst tartalmazó) tesztelési esetben érdekes módon az ütközések száma jelentősen nem változott az első tesztesethez képest, azonban az átlagos jutalom csökkent az óvatosabb stratégia következtében. Mindkét algoritmus megtanulta felhasználni a kiegészítő információkat, azonban erre csak a többi ágens viselkedésének becsléséhez, valamint az ehhez való alkalmazkodás érdekében van szükségük. Az utolsó tesztesetnél látható, hogy a kritikus ágensszám felett már az állapottér kibővítésével sem érhető el magasabb jutalom vagy alacsonyabb ütközésszám.

A két tanítási halmaz során az ütközések számát összehasonlítva látható, hogy a kívánt irányszögek szinuszának ($\sin \gamma_{ref}$) és koszinuszának ($\cos \gamma_{ref}$) hozzáadása a megfigyelési térhez a multiágens „policy” esetén jelentősebb hatást eredményez. Például a második tesztelési esetben az ütközések száma multiágens „policy” esetén 85,71%-kal, míg egyágenses „policy” esetén 50%-kal csökkent). Ez megmutatja a multiágens tanítás előnyeit, hiszen a tanítás során az ágens megismerhette a többi ágens által irányított jármű viselkedését, így alkalmazkodni tudott. Ezen kívül az állapottér kibővítésének hatása az ágensek szándékaival a kommunikáció pozitív hatásait is megmutatja.

Az 5. ábrán látható átlagsebességek ugyanazt a tendenciát mutatják, mint az előző halmaz esetén. Az egyágenses „policy” hajlamos bátrabban cselekedni, azaz magasabb sebességet választani, mely néha ütközéssel végződik, így az epizódok jutalmainak ingadozását okozza. A korábban tapasztalt jutalomingadozás mértéke mindkét esetben

csökkent. Azonban a jutalmak multi-ágens policy esetén kiemelkedőbbek, ahol a legtöbb jutalom értéke 22 és 24 közötti.



5. ábra: Az első ágens átlagos sebességei kibővített állapottér esetén

Fontos kiemelni, hogy a multiágens „policy” teljesen kompetitív jutalmazással lett tanítva. Emiatt az ágensektől önzőbb viselkedés lenne elvárható. Azonban az ütközések büntetésének, valamint a tanítás közbeni változatos viselkedések megfigyelésének köszönhetően a multi-ágens algoritmus egy robusztusabb stratégiát tanult meg, mely összességében jobb teljesítményt eredményezett.

6. KONKLÚZIÓ

Ebben a kutatásban az egyágenses és multiágenses megerősítéses tanulás közti különbséget mutattuk be egy autópályás döntéshozatali helyzetben. A tanított ágensek különböző számú, a megtanult „policy” által irányított ágenst feltételező esetekben kerültek tesztelésre. A multiágens módszer jelentősen kevesebb ütközést, valamint alacsonyabb átlagsebességet eredményezett az egyágenses módszerhez képest. Továbbá a kibővített megfigyelési tér hatásai is vizsgálatra kerültek. Látható, hogy ez segített az ütközések számát csökkenteni a sebességek egy biztonságos átlagérték körüli stabilizálásával. Emellett fontos hangsúlyozni, hogy az ismertetett algoritmusok által elért performancia még messze nem elegendő a való életben lejátszódó forgalmi helyzetekben történő alkalmazásra, hiszen még két ütközés is elfogadhatatlanul magas érték 100 epizód esetén. Az eredményeket összehasonlítva azonban megállapítható a MARL előnye forgalmi helyzetek kezelésében.

Az elért eredmények tovább javíthatók a hálózat méretének növelésével, valamint fejlettebb algoritmusok használatával. Ezen kívül szükséges a környezet stabilabb inicializálása, mivel néhány epizód esetén már a járművek kezdeti állapota miatt elkerülhetetlen volt az ütközés. Valamint célszerű a környezet több paraméterének (pl.: sávok száma) véletlenszerű változtatása. Továbbá tervezzük a „curriculum learning” alkalmazását, ahol az ágensek száma (egyben a feladat nehézsége) a tanítási folyamat előre haladásával együtt növekszik.

KÖSZÖNETNYILVÁNÍTÁS

A publikációban szereplő kutatást a BME-KJK az Európai Unió támogatásával valósította meg, az Autonóm Rendszerek Nemzeti Laboratórium keretében. (RRF-2.3.1-21-2022-00002)
A kutatást a Magyar Tudományos Akadémia a Bolyai János Kutatási Ösztöndíjjal támogatta.

HIVATKOZÁSOK

Aradi, S. (2020). Survey of deep reinforcement learning for motion planning of autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, *23*(2), 740–759.

Bello, I., Pham, H., Le, Q. V, Norouzi, M., & Bengio, S. (2017). *Neural Combinatorial Optimization with Reinforcement Learning*.

Berner, C., Brockman, G., Chan, B., Cheung, V., Debiak, P., Dennison, C., Farhi, D., Fischer, Q., Hashme, S., Hesse, C., Józefowicz, R., Gray, S., Olsson, C., Pachocki, J., Petrov, M., d. O. Pinto, H. P., Raiman, J., Salimans, T., Schlatter, J., … Zhang, S. (2019). *Dota 2 with Large Scale Deep Reinforcement Learning*.

Chen, D., Hajidavalloo, M., Li, Z., Chen, K., Wang, Y., Jiang, L., & Wang, Y. (2022). *Deep Multi-agent Reinforcement Learning for Highway On-Ramp Merging in Mixed Traffic*.

Farazi, N. P., Ahamed, T., Barua, L., & Zou, B. (2020). *Deep Reinforcement Learning and Transportation Research: A Comprehensive Review*.

Farooq, M. S., Khalid, H., Arooj, A., Umer, T., Asghar, A. B., Rasheed, J., Shubair, R. M., & Yahyaoui, A. (2023). A Conceptual Multi-Layer Framework for the Detection of Nighttime Pedestrian in Autonomous Vehicles Using Deep Reinforcement Learning. *Entropy*, *25*(1). https://doi.org/10.3390/e25010135

Gronauer, S., & Diepold, K. (2022). Multi-agent deep reinforcement learning: a survey. *Artificial Intelligence Review*, 1–49.

Kaushik, M., S, P., & Krishna, K. M. (2018). *Parameter Sharing Reinforcement Learning Architecture for Multi Agent Driving Behaviors*.

Kesting, A., Treiber, M., & Helbing, D. (2006). MOBIL: General lane-changing model for car-following models. *Proceedings of the Transportation Research Board Annual Meeting*.

Kővári, B., Tettamanti, T., & Bécsi, T. (2022). Deep Reinforcement Learning based approach for Traffic Signal Control. *Transportation Research Procedia*, *62*, 278–285. https://doi.org/https://doi.org/10.1016/j.trpro.2022.02.035

Leurent, E. (2018). An Environment for Autonomous Driving Decision-Making. In *GitHub repository*. GitHub.

Leurent, E., & Mercat, J. (2019). *Social Attention for Autonomous Decision-Making in Dense Traffic*.

Li, X., Xu, X., & Zuo, L. (2015). Reinforcement learning based overtaking decision-making for highway autonomous driving. *2015 Sixth International Conference on Intelligent Control and Information Processing (ICICIP)*, 336–342. https://doi.org/10.1109/ICICIP.2015.7388193

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., & Wierstra, D. (2019). *Continuous control with deep reinforcement learning*.

Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994* (pp. 157–163). Elsevier.

Ly, A. O., & Akhloufi, M. (2020). Learning to drive by imitation: An overview of deep behavior cloning methods. *IEEE Transactions on Intelligent Vehicles*, *6*(2), 195–209.

Mavrogiannis, A., Chandra, R., & Manocha, D. (2022). *B-GAP: Behavior-Rich Simulation and Navigation for Autonomous Driving*.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). *Playing Atari with Deep Reinforcement Learning*.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., & others. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529–533.

Moghadam, M., Alizadeh, A., Tekin, E., & Elkaim, G. H. (2021). A deep reinforcement learning approach for long-term short-term planning on frenet frame. *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*, 1751–1756.

Polack, P., Altché, F., d'Andréa-Novel, B., & de La Fortelle, A. (2017). The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles? *2017 IEEE Intelligent Vehicles Symposium (IV)*, 812–818.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., & Hassabis, D. (2017). *Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm*.

Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W. M., Zambaldi, V., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J. Z., Tuyls, K., & Graepel, T. (2017). *Value-Decomposition Networks For Cooperative Multi-Agent Learning*.

Treiber, M., Hennecke, A., & Helbing, D. (2000). Congested traffic states in empirical observations and microscopic simulations. *Phys. Rev. E*, *62*(2), 1805–1824. https://doi.org/10.1103/PhysRevE.62.1805

Wu, Y., Tan, H., Peng, J., Zhang, H., & He, H. (2019). Deep reinforcement learning of energy management with continuous control strategy and traffic information for a series-parallel plug-in hybrid electric bus. *Applied Energy*, *247*, 454–466. https://doi.org/https://doi.org/10.1016/j.apenergy.2019.04.021