

Energiaoptimalizálás WSN hálózatokon belül rajintelligenciával és megerősítéses tanulással

Energy Optimization in WSNs Using Swarm Intelligence and Reinforcement Learning

FILEP Levente ¹, Dr. GÁL Zoltán ²

^{1,2} Debreceni Egyetem Informatikai Kar
4028 Debrecen, Kassai út 26., Magyarország, <http://www.inf.unideb.hu>

¹ filep.levente@inf.unideb.hu

² gal.zoltan@inf.unideb.hu

Abstract

Wireless sensor networks play an important role in numerous application areas. Sensor nodes typically have limited energy resources; therefore, optimizing their energy consumption is essential. This study presents a hybrid approach in which a reinforcement learning agent estimates an appropriate number of cluster heads, while a swarm intelligence algorithm selects the most suitable nodes for this purpose in order to extend the lifetime of the network.

Keywords: Wireless Sensor Networks, Reinforcement Learning, Swarm Intelligence, Energy Optimization, Clustering

Kivonat

A vezeték nélküli szenzorhálózatok napjainkban is fontos szerepet töltenek be számos alkalmazási területen. A szenzorcsomópontok jellemzően korlátozott energiakészlettel rendelkeznek, ezért energiafogyasztásuk optimalizálása elengedhetetlen. A tanulmány egy hibrid megközelítést mutat be, amelyben egy megerősítéses tanulási ágens megbecsüli a megfelelő klaszterfejszámot, míg egy rajintelligencia-algoritmus kiválasztja az erre a célra legalkalmasabb csomópontokat a hálózat élettartamának növelése érdekében.

Kulcsszavak: vezeték nélküli szenzorhálózatok, megerősítéses tanulás, rajintelligencia, energiaoptimalizálás, klaszterezés

1. BEVEZETÉS

A vezeték nélküli szenzorhálózatok fontos szerepet töltenek be számos gyakorlati alkalmazási területen, mint például a környezetmonitorozásban, az intelligens mezőgazdaságban és az ipari automatizálásban. A térben elosztott szenzorcsomópontok valós idejű adatokat gyűjtenek, dolgoznak fel és továbbítanak. Mivel ezek a csomópontok jellemzően akkumulátorról működnek, és gyakran nehezen hozzáférhető helyeken vannak telepítve, a hálózat energiahatékony működése alapvető követelmény [9].

A csomópontok energiafogyasztásának jelentős részét a kommunikációs adó-vevő működtetése teszi ki, az adatátvitel energiaigénye pedig az átviteli távolsággal növekszik. Az energiafogyasztás csökkentésére több módszer alkalmazható, például az adóteljesítmény szabályozása, az energiahatékony útvonalválasztás és a klaszterezés [4,9]. Utóbbi során a szenzorcsomópontok klaszterekbe szerveződnek, amelyekben egy klaszterfej (Cluster Head, CH) gyűjti össze és továbbítja az adatokat a bázisállomás (Base Station, BS) felé. Bár a CH energiafogyasztása nagyobb, a klaszter többi csomópontja rövidebb távolságra továbbít, így csökken a hálózat összesített energiafogyasztása. A klaszterfej szerepének időszakos váltása továbbá kiegyenlíti az energiatelheret, növelve a hálózat élettartamát [3].

A klaszterezés hatékonysága nagymértékben függ a CH-k számától és elhelyezkedésétől. Ezek meghatározása összetett optimalizálási probléma, amelynél nagy hálózatok esetén a teljes keresési tér bejárása gyakorlatilag kivitelezhetetlen. Emiatt gyakran heurisztikus és metaheurisztikus módszereket alkalmaznak közel optimális megoldások keresésére.

A rajintelligencia a természetben megfigyelhető kollektív viselkedésből inspirálódó metaheurisztikus megközelítés, amely decentralizált ágensek együttműködésével keres kedvező megoldásokat [2,6,11]. Nagy WSN-ekben azonban az ilyen algoritmusok számítási költsége jelentőssé válhat. A gépi tanulás, különösen a megerősítéses tanulás, lehetőséget kínál a szükséges klaszterfejszám hálózat aktuális állapotához igazodó, adaptív becslésére [8].

A tanulmány ezért hibrid megoldást alkalmaz: egy RL-ágens a hálózat aktuális állapota alapján megbeszéli a megfelelő klaszterfejszámot, majd egy rajintelligencia-algoritmus kiválasztja a legkedvezőbb CH-jelölteket. A megközelítés célja a WSN energiahatékonyságának és élettartamának növelése.

2. IRODALMI ÁTTEKINTÉS

A javasolt megoldás a LEACH klaszterezési protokollra, rajintelligencia-algoritmusokra és PPO-alapú megerősítéses tanulási ágensre épül.

2.1. WSN-klaszterezés

A Low-Energy Adaptive Clustering Hierarchy (LEACH) az egyik korai, alacsony többletterhelésű, energiahatékony klaszterezési protokoll [3]. A klaszterképzést és a CH-k kiválasztását decentralizáltan végzi, ezért csak kis mennyiségű kiegészítő adatforgalmat igényel. Az újonnan kiválasztott CH a választás eredményét továbbítja a szomszédos csomópontoknak. Egyszerűsége, skálázhatósága és dinamikus hálózati környezetben mutatott robusztussága miatt a LEACH ma is széles körben alkalmazott protokoll, amelyet gyakran más optimalizálási módszerekkel kombinálnak [3].

A szakirodalomban számos olyan megoldás található, amely a szenzorhálózatok klaszterezését heurisztikus vagy metaheurisztikus optimalizálási eljárásokkal kapcsolja össze [4,9].

2.2. Rajintelligencia

A rajintelligencia bioinspirált metaheurisztikus optimalizálási módszerek csoportja, amely a természetben megfigyelhető kollektív viselkedésből, például madárrajok mozgásából, valamint egyes fizikai jelenségekből merít inspirációt [2,11]. Az egyes ágensek egyszerű szabályok alapján működnek, az intelligens viselkedés pedig nem az egyedi képességeikből, hanem az ágensek közötti kölcsönhatásból és decentralizált együttműködésből alakul ki.

Rajintelligencia-alapú optimalizálás során minden ágens egy lehetséges megoldást reprezentál, amelynek minőségét egy előre definiált fitnessfüggvény (FF) értékeli ki. A raj közösen járja be a keresési teret, és az iterációk során fokozatosan az optimális vagy ahhoz közeli megoldások felé konvergál [2,6,11].

A javasolt AISI modellben a Bald Eagle Search (BES) optimalizálási algoritmus kerül alkalmazásra kiegészítő rajintelligencia-módszerként [1].

2.3. Megerősítéses tanulás és neurális hálózatok

A megerősítéses tanulás során egy ágens a környezettel való kölcsönhatásokon keresztül tanul, és a meghozott döntések alapján jutalmat vagy büntetést kap. Célja olyan döntési stratégia kialakítása, amely hosszú távon maximalizálja az összesített jutalmat [8]. A folyamat matematikai leírására gyakran Markov-döntési folyamatot (Markov Decision Process, MDP) alkalmaznak, amely az állapotokat, a lehetséges döntéseket, az állapotátmeneteket és a jutalomfüggvényt foglalja egységes keretbe. A tanulás a feltárás és a már megszerzett ismeretek kiaknázásának iteratív váltakozásával, közvetlen felügyelet nélkül történik [8].

A neurális hálózatok olyan számítási modellek, amelyek adatokból összetett mintázatokat és reprezentációkat képesek megtanulni [10]. A mély megerősítéses tanulás a megerősítéses tanulási módszereket több-rétegű neurális hálózatokkal kapcsolja össze [8,10].

A Proximal Policy Optimization (PPO) széles körben alkalmazott mély megerősítéses tanulási algoritmus [12]. A policy-gradient módszerek családjába tartozik, amelyek az ágens döntési stratégiáját a várható hozam gradiensének becslése alapján frissítik. E módszerek egyik fontos problémája a frissítési lépés megfelelő méretének meghatározása: a túl nagy lépés jelentősen ronthatja a megtanult stratégiát, míg a túl kicsi lépés lassítja a tanulást. A PPO ezt egy korlátozó („clipping”) mechanizmussal kezeli, amely meghatározott tartományban tartja a stratégia frissítését, ezáltal stabil és gyakorlatban jól alkalmazható tanulási eljárást biztosít [12].

3. JAVASOLT MODELL

Az AISI modell működése két egymásra épülő lépésből áll. Elsőként a megerősítéses tanulási ágens a WSN aktuális állapotából becsüli meg, hogy hány klaszterfej alkalmazása indokolt. Ezt követően a rajintelligencia-algoritmus a megadott CH-szám alapján kiválasztja azokat a szenzorcsomópontokat, amelyek a klaszterfej szerepére a legalkalmasabbak.

A hálózat állapotának leírásához a teljes WSN-terület több megfigyelési régióra kerül felosztásra. Az ágens nem az egyes szenzorcsomópontok koordinátáit és energiaszintjét használja közvetlen bemenetként, hanem régióként az aktív csomópontok számát és az adott régió összenergiaszintjét figyeli. A régiók számát egy τ hiperparaméter határozza meg.

Ez a megközelítés függetleníti az ágens bemenetének méretét a hálózat csomópontszámától. Ennek köszönhetően ugyanaz a betanított modell különböző méretű WSN-ekben is használható, és olyan hálózatokra is alkalmazható, amelyek több szenzorcsomópontot tartalmaznak, mint a tanítás során használt konfiguráció.

Az AISI modellben a klaszterfejek száma nem előre rögzített érték, hanem a hálózat pillanatnyi állapothoz igazodik. A LEACH-alapú megoldásokban gyakran alkalmazott, körülbelül 5%-os CH-arány általánosan használható beállítás, de nem feltétlenül optimális a WSN teljes élettartama során. Ezért az ágens minden új klaszterfej-választáskor a hálózat aktuális állapota alapján becsüli meg a szükséges CH-k számát.

4. ALKALMAZOTT MÓDSZERTAN

A javasolt modell kiértékelése MATLAB-környezetben végzett szimulációk segítségével történt. A következőkben bemutatásra kerül a szimulációs környezet, az alkalmazott paraméterek, valamint az ágens és a rajintelligencia-algoritmus működésének főbb beállításai.

4.1. WSN-környezet

Az AISI modell és a klasszikus LEACH megoldás összehasonlítása három különböző hálózati méret mellett történt: 50, 100 és 150 szenzorcsomóponttal. A csomópontok kezdeti energiaszintje 1 J volt, elhelyezésük pedig véletlenszerűen történt egy 200×200 méteres területen. A bázisállomás a terület középpontjában, a (100, 100) koordinátán került elhelyezésre.

A szenzorcsomópontok elhelyezésénél legalább 5 méteres távolság került előírásra a bázisállomástól. Erre azért volt szükség, mert a BS-hez túl közel elhelyezkedő csomópontok nagyon alacsony energiafogyasztással működnének, ami mesterségesen megnövelné a hálózat szimulált élettartamát.

Mind a LEACH, mind az AISI modell esetében a csomópontok a távolság alapján a legközelebbi klaszterfejhez, illetve szükség esetén közvetlenül a bázisállomáshoz kapcsolódnak.

A vizsgált hálózatok, különösen az 50 csomópontos konfiguráció, a szimulációs terület méretéhez viszonyítva ritka csomópont-elrendezésnek tekinthetők. Emiatt egyes csomópontok viszonylag nagy távolságra kerülhetnek a klaszterfejektől vagy a bázisállomástól, ami jelentősen növelheti az adatátvitel energiaigényét.

4.2. Energiafelhasználási modell

A szenzorcsomópontok energiafogyasztásának jelentős része az adatátvitelhez kapcsolódik, amely nagymértékben függ a kommunikációs távolságtól. Az energiaigény meghatározásához többutas terjedési modell került alkalmazásra. A modellben a távolságfüggés kitevője $b = 2$, ha $d < d_0$, és $b = 4$, ha $d \geq d_0$, ahol $d_0 = 87 \text{ m}$ [7]. A küszöbérték felett ezért az átviteli energiaigény a távolság növekedésével lényegesen gyorsabban emelkedik. A 200×200 méteres szimulációs terület megválasztása lehetővé teszi, hogy ez a hatás hangsúlyosabban jelenjen meg, és ezáltal a klaszterezés hatékonysága is jobban vizsgálható legyen.

A WSN működésének vizsgálata szimulációs epizódokban történik, ahol egy epizód egy hálózati lépésnek felel meg. Minden epizódban kiszámításra kerül az egy szenzormérés továbbításához szükséges energia minden csomópontra. Az adott epizód teljes energiafogyasztása a klaszterezés minőségének egyik mérőszámaként szolgál, ahol az alacsonyabb érték kedvezőbb konfigurációt jelez.

Egy csomópont energiaigénye csak abban az esetben kerül figyelembevételre, ha elegendő energiával rendelkezik az adatátvitelhez. Amennyiben energiatartaléka nem elegendő a továbbításhoz, inaktívvá válik, és energiafogyasztása már nem kerül beleszámításra az adott epizód összesített energiafelhasználásába.

4.3. A klaszterfejek számának hatása

A LEACH-alapú megoldásokban gyakran az aktív csomópontok 5%-a kerül klaszterfejként kiválasztásra [3]. A szimulációs eredmények alapján azonban ez az arány nem minden hálózati állapotban optimális.

A megfelelő klaszterfejszám az aktív csomópontok számától, azok energiaszintjétől és egymáshoz viszonyított távolságától is függ. Nem megfelelő CH-szám esetén kedvezőtlenebb energiafelhasználás és az inaktív csomópontok számának növekedése figyelhető meg.

4.4. Rajintelligencia-alapú klaszterfej-kiválasztás

A rajintelligencia-algoritmus bemenetét a WSN csomópontjai és az ágens által becsült klaszterfejszám alkotják. Az optimalizálás célja azon csomópontok kiválasztása, amelyek klaszterfejként történő alkalmazásával az adatátvitel energiagénye minimalizálható.

$$FF = \alpha \cdot \sum_{i \in N} d(x_i, x_{ch}(i)) + (1 - \alpha) \cdot \sum_{j \in M} d(x_j, x_{BS}(j)) \quad (1)$$

Mivel az átviteli energia szorosan összefügg a kommunikációs távolsággal, az (1) képletben megadott fitnessfüggvény a kommunikációs távolságok minimalizálására épül. Ennek során a normál csomópontok és a hozzájuk tartozó klaszterfejek közötti, valamint a klaszterfejek és a bázisállomás közötti távolság egyaránt figyelembevételre kerül. A két tényező súlyozását az α paraméter szabályozza.

4.5. Megerősítéses tanulási ágens

Az ágens megvalósításához a MATLAB beépített PPO megerősítéses tanulási modellje került alkalmazásra [12]. A vizsgálatok során $\tau = 16$ érték lett beállítva, amely a terület 4×4 -es, összesen 16 megfigyelési régióra történő felosztását jelenti. Az ágens bemenetét az egyes régiókban található aktív csomópontok száma és az adott régió összenegatív energiaszintje alkotja, kimenete pedig a javasolt klaszterfejszám.

$$r_t = (E - \Delta E_t) - \beta \cdot \bar{N}_t \quad (2)$$

Az ágens minden tanítási lépésben a hálózat aktuális állapota alapján hoz döntést, amelyet követően a (2) képlet szerinti lépésenkénti jutalmat kapja. A jutalomfüggvény az alacsony energiafelhasználást jutalmazza, míg az inaktív csomópontok számát β súlyozási tényezővel bünteti. A szimuláció során a klaszterfejek időszakosan újraválasztásra kerülnek; a két ilyen kiválasztás közötti időszak jelenti a klaszterezési kört. A klaszterezési kör végén az ágens további $\gamma \cdot L$ jutalmat kap, ahol L a WSN addig elért élettartamát, γ pedig egy skálázási együtthatót jelöl.

A betanítás 1000 tanítási epizódon keresztül történt, epizódonként legfeljebb 200 tanítási lépéssel. A túl alacsony vagy túl magas klaszterfejszámra vonatkozó becslések büntetésre kerültek. Egy tanítási epizód akkor fejeződött be, amikor a maximális lépésszám elérésre került, vagy a hálózat teljes energiatartaléka kimerült. A tanítás során a csomópontok kezdeti energiaszintje epizódonként 0.1 és 1.0 J közötti értékre lett beállítva.

4.6. Főbb szimulációs paraméterek

A szimulációk során $\tau = 16$ és $\beta = 0.99$ érték került alkalmazásra. Rajintelligencia-algoritmusként a Bald Eagle Search (BES) került felhasználásra [1], $\alpha = 0.6$ súlyozási tényezővel. A populáció mérete 10 egyedre, az optimalizálás maximális iterációszáma pedig 100-ra lett beállítva. Az AISI modellben a klaszterfejek száma az aktív csomópontok 5–15%-a közötti tartományra lett korlátozva.

4.7. Statisztikai kiértékelés

Annak vizsgálatára, hogy az AI-ágens a hálózat aktuális állapotához igazodva becsüli-e meg a klaszterfejek számát, nem pedig pusztán véletlenszerű értéket választ a megengedett tartományból, több LEACH-alapú konfiguráció került összehasonlításra. A klasszikus 5%-os beállítás mellett egy 7.5%-os fix klaszterfejarányt, valamint egy véletlenszerűen, 5–10% közötti klaszterfejarányt alkalmazó változat is vizsgálatra került.

Mivel az egyes vizsgált konfigurációk eredményei különálló, egymástól független szimulációs futtatásokból származnak, azok statisztikai összehasonlításához a Wilcoxon rank-sum teszt, más néven Mann–Whitney U-próba [5] került alkalmazásra. Ez egy nemparaméteres statisztikai eljárás két független minta összehasonlítására, amely akkor is alkalmazható, ha az eredmények eloszlása nem ismert vagy nem feltételezhető normálisnak. A teszt annak vizsgálatára szolgál, hogy a különböző modellek által elért hálózati élettartamok

közötti eltérések statisztikailag kimutathatók vagy azok a szimulációs futtatások véletlenszerű ingadozásával magyarázhatók.

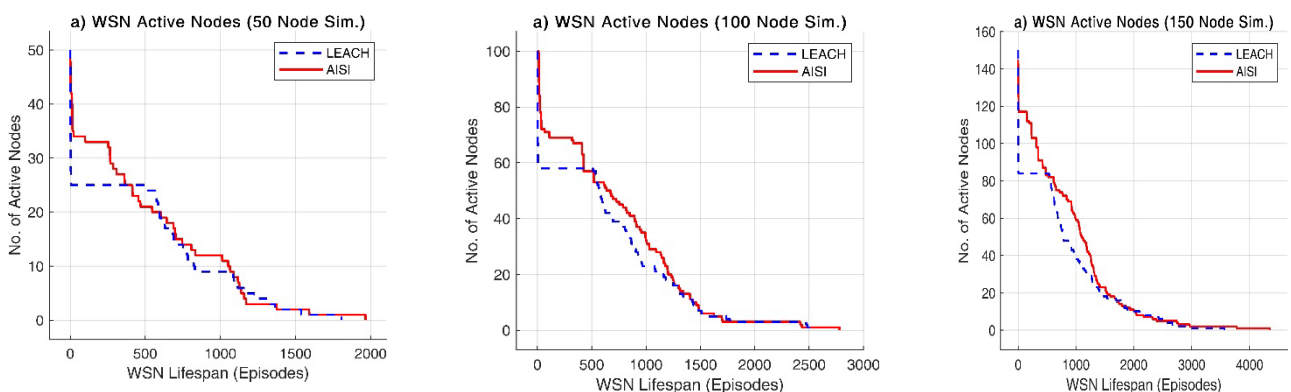
5. A MODELL TELJESÍTMÉNYÉNEK KIÉRTÉKELÉSE

A modell teljesítményének kiértékelése több szempont alapján történt, többek között a WSN élettartama, a maradék energiaszint és az epizódonkénti energiafelhasználás figyelembevételével.

5.1. Hálózat élettartalma

A csomópontok ritka elrendezése következtében a szimuláció kezdeti szakaszában több csomópont energiatartaléka rövid időn belül kimerül, ami mindhárom vizsgált hálózati konfigurációban jól megfigyelhető.

Az 50 csomópontos hálózat esetében a LEACH modell 1807 epizódos élettartamot ért el, míg az AISI alkalmazásával ez 1967 epizódra növekedett, ami 8.85%-os javulást jelent. Bár az aktív csomópontok számának alakulása a két modell esetében hasonló volt, az AISI használatával a hálózat magasabb maradék energiaszintet őrzött meg, ami hozzájárult a hosszabb élettartamhoz.



1. ábra. Aktív szenzorcsomópontok számának alakulása: 50, 100 és 150 csomópontos WSN esetén

Az epizódonkénti energiafogyasztás kezdetben magasabb volt az AISI esetében, mivel több csomópont maradt aktív. Körülbelül az 500. epizódnál a két modell aktív csomópontjainak száma kiegyenlítődött, és ettől a ponttól az SI-optimalizálás hatására az AISI alacsonyabb energiafelhasználást mutatott. A későbbi szakaszban az energiafogyasztás ismét magasabbá vált, ami az AISI esetében továbbra is nagyobb számú aktív csomóponttal magyarázható.

A 100 csomópontos konfigurációban a LEACH hálózat élettartama 2503 epizód volt, míg az AISI modell 2780 epizódot ért el, ami 11.07%-os növekedést jelentett. A 150 csomópontos esetben a különbség még jelentősebb volt: a LEACH 3566, az AISI pedig 4348 epizódos élettartamot ért el, amely 21.93%-os javulásnak felel meg.

Mindhárom vizsgált hálózati méretnél hasonló tendencia volt megfigyelhető. Az SI-optimalizálás következtében az AISI kezdetben több csomópontot tart aktív állapotban, ezért az energiafelhasználása is magasabb. Körülbelül az 500. epizódra az aktív csomópontok száma a két módszernél közel azonos szintre kerül, azonban az AISI ekkorra több maradék energiával rendelkezik, ami lehetővé teszi a hálózat működési idejének további meghosszabbítását.

Az összesített eredmények alapján a hálózat szenzorsűrűségének növekedésével mindkét modell élettartama növekedett, ugyanakkor az AISI által elért javulás nagyobb csomópontszám mellett egyre jelentősebbé vált. Az 50, 100 és 150 csomópontos konfigurációkban rendre 8.85%, 11.07% és 21.93% élettartam-növekedés volt tapasztalható a LEACH-hez képest.

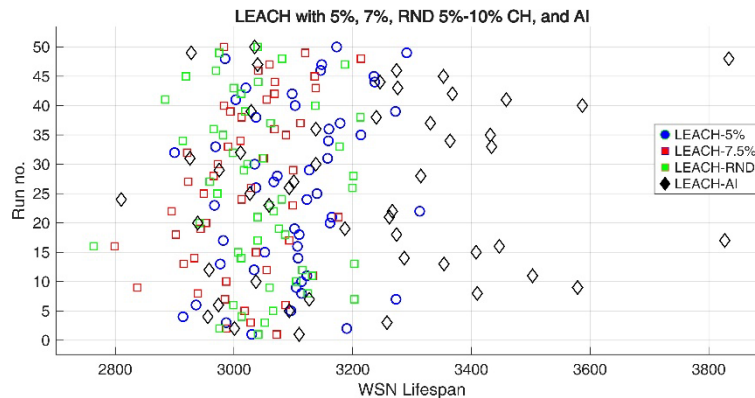
5.2. Az AI-alapú klaszterfejszám-becslés vizsgálata

A következő vizsgálat célja annak meghatározása, hogy az AISI modellnél tapasztalt élettartam-növekedés milyen mértékben származik az AI-ágens által becsült klaszterfejszámból, illetve mennyiben magyarázható önmagában azzal, hogy a klaszterfejek száma nem rögzített. Ennek vizsgálatához a klasszikus, 5%-os klaszterfejarányt alkalmazó LEACH modellt, a továbbiakban LEACH-5, mellett több további konfiguráció is összehasonlításra került.

Elsőként egy LEACH-7.5 konfiguráció került kialakításra, amelyben az aktív csomópontok 7.5%-a tölt be klaszterfej-szerepet. Emellett egy LEACH-RND változat is kialakításra került, amelyben a klaszterfejek

száma minden kiválasztáskor véletlenszerűen változik az aktív csomópontok 5–10%-a között. A felső határ 10%-ban került meghatározásra, mivel a korábbi vizsgálatok alapján ennél nagyobb klaszterfejarány jellemzően már nem eredményez kedvezőbb működést. Ez a korlátozás a véletlenszerű modell számára kedvezőbb feltételt biztosít, mint az AISI modellben eredetileg alkalmazott 5–15%-os tartomány.

Az AI-ágens önálló hatásának vizsgálatához az ágens a klasszikus LEACH modellhez is csatlakoztatásra került; ez a konfiguráció a továbbiakban LEACH-AI megnevezéssel szerepel. Ebben az esetben a klaszterfejek kiválasztása továbbra is a LEACH működését követi, azonban a klaszterfejek számát az ágens becsüli meg. A LEACH-RND modellel való közvetlen összehasonlíthatóság érdekében az ágens kimenete ebben az esetben szintén az aktív csomópontok 5–10%-a közötti tartományra lett korlátozva. Amennyiben az ágens valóban a hálózat aktuális állapotához igazodó klaszterfejszámot becsül meg, a LEACH-AI teljesítményének el kell különülnie a klaszterfejszámot véletlenszerűen megválasztó LEACH-RND modell eredményeitől.



2. ábra. A LEACH modell összehasonlítása fix, véletlenszerű és AI-alapú klaszterfejszám-becslés esetén

A 2. ábra alapján a LEACH-AI számos futtatásban hosszabb hálózati élettartamot eredményezett a klasszikus LEACH modellnél, ugyanakkor néhány esetben annál gyengébb teljesítmény is megfigyelhető volt. Ennek egyik lehetséges oka, hogy az ágens bizonyos hálózati állapotokban nem megfelelő klaszterfejszámot becsül, ami a tanítás további finomításának szükségességére utalhat. A legjobb egyedi futtatások összehasonlításakor a LEACH-AI 3834, míg a LEACH-5 3313 epizódos élettartamot ért el, ami 15.73%-os különbségnek felel meg.

A klasszikus LEACH-5 modell átlagos hálózati élettartama 3101.02 epizód, szórása pedig 98.04 volt. A LEACH-RND esetében 3045.22 epizódos átlag és 90.56-os szórás adódott. A Wilcoxon rank-sum teszt alapján a két modell eredményei közötti különbség statisztikailag szignifikáns volt ($p = 0.006740$). A LEACH-7.5 konfiguráció még alacsonyabb, 3019.08 epizódos átlagos élettartamot eredményezett, 84.25-ös szórással. A LEACH-5 és LEACH-7.5 közötti különbség szintén statisztikailag szignifikánsnak bizonyult ($p = 0.000053$), ami azt mutatja, hogy ebben a vizsgált konfigurációban a klaszterfejarány 7.5%-ra növelése kedvezőtlenebb hálózati élettartamot eredményezett.

A LEACH-AI konfiguráció átlagos hálózati élettartama 3221.62 epizód volt, 227.23-as szórással. A LEACH-AI és LEACH-RND eredményei közötti különbség statisztikailag szignifikáns volt ($p = 0.000052$), és ugyanez volt megfigyelhető a LEACH-AI és LEACH-7.5 összehasonlításakor is ($p = 0.0000020$). A klasszikus LEACH-5 és a LEACH-AI között szintén statisztikailag szignifikáns eltérés mutatkozott ($p = 0.010747$).

Az eredmények alapján sem a klaszterfejarány egyszerű 7.5%-ra növelése, sem annak véletlenszerű változtatása nem eredményezett élettartam-növekedést a klasszikus LEACH-5 modellhez képest. Ezzel szemben az AI-ágens által becsült változó klaszterfejszám átlagosan hosszabb hálózati élettartamot eredményezett. A LEACH-AI és LEACH-RND közötti statisztikailag szignifikáns különbség arra utal, hogy az ágens nem pusztán véletlenszerűen választ értéket a megengedett tartományból, hanem a hálózat aktuális állapotának figyelembevételével képes kedvezőbb klaszterfejszámot becsülni. Az egyes gyengébb futtatások ugyanakkor azt is jelzik, hogy az ágens további tanítása és finomhangolása még javíthatja a becslések megbízhatóságát.

5.3. Az aktív csomópontok számának alakulása

A teljes hálózati élettartam mellett érdemes megvizsgálni azt is, hogy az egyes modellek mellett milyen ütemben csökken az aktív szenzorcsomópontok száma. Ennek érdekében meghatározásra került, hogy a LEACH és az AISI hányadik epizódban éri el az aktív csomópontok 75%, 50%, 25% és 10%-os küszöbértékét.

A 90%-os küszöbérték nem került figyelembevételre, mivel azt a hálózatok már a szimuláció első néhány epizódjában elérik, így ez az összehasonlítás nem mutat számottevő különbséget a modellek között.

Az aktív csomópontok küszöbértékeinek elérése LEACH és AISI modellek esetén

1. táblázat

Küszöbérték	50 csomópont	100 csomópont	150 csomópont
75%	3 / 15	3 / 35	3 / 233
50%	7 / 415	571 / 638 (+11.73%)	584 / 777 (+33.05%)
25%	779 / 833 (+6.93%)	943 / 1161 (+23.12%)	1040 / 1266 (+21.73%)
10%	1157 / 1158 (+0.09%)	1425 / 1431 (+0.42%)	1782 / 1745 (-2.08%)

Az 1. táblázat celláiban a LEACH / AISI sorrendben az adott küszöbérték elérésének epizód száma, zárójelben pedig az AISI relatív eltérése szerepel. A középső küszöbértékeknel az AISI egyértelműen később éri el az adott aktívcsomópont-arányt, vagyis hosszabb ideig képes nagyobb számú csomópontot működésben tartani. A 150 csomópontos esetben a 10%-os küszöb ugyan valamivel korábban következik be az AISI esetében, ami részben annak következménye, hogy a korábbi szakaszokban több csomópont maradt aktív, és ez nagyobb energiafelhasználással járt. A megmaradt csomópontok ugyanakkor magasabb maradék energiaszinttel rendelkeznek, ezért a hálózat teljes élettartama ezt követően is hosszabb marad az AISI esetében.

6. ÖSSZEFOGLALÁS

Az AISI modell a megerősítéssel tanuló és a rajntelligencia-alapú optimalizálást egyesíti a WSN-ek élettartamának növelése érdekében. Az AI-ágens a hálózat aktuális állapota alapján becsüli meg a szükséges klaszterfejek számát, míg a rajntelligencia-algoritmus kiválasztja az erre legalkalmasabb csomópontokat. A szimulációk során az AISI minden vizsgált hálózati méret esetén hosszabb élettartamot ért el a klasszikus LEACH modellnél.

A kiegészítő vizsgálatok alapján az AI-ágens működése statisztikailag szignifikáns különbséget eredményezett a fix és a véletlenszerű klaszterfejszámot alkalmazó változatokhoz képest. Ez arra utal, hogy az ágens nem pusztán véletlenszerűen választ a megengedett tartományon belül, hanem a hálózat aktuális állapotát felhasználva képes kedvezőbb klaszterfejszámot becsülni. Ugyanakkor az egyes gyengébb futtatások azt is jelzik, hogy az ágens további tanítása és finomhangolása még javíthatja a becslések megbízhatóságát.

KÖSZÖNETNYILVÁNÍTÁS

Ezt a munkát a Debreceni Egyetem QoS-HPC-IoT Laboratóriuma, valamint a TKP2021-NKTA-34 projekt támogatta. A TKP2021-NKTA-34 számú projekt a Kulturális és Innovációs Minisztérium támogatásával, a Nemzeti Kutatási, Fejlesztési és Innovációs Alapból, a TKP2021-NKTA pályázati program finanszírozásában valósult meg. A kutatást továbbá az Osztrák–Magyar Akció Alapítvány OMAA programja is támogatta, projektazonosító: 121ou2.

IRODALMI HIVATKOZÁSOK

- [1] Alsattar, H. A.; Zaidan, A. A.; Zaidan, B. B.: Novel meta-heuristic bald eagle search optimisation algorithm. *Artificial Intelligence Review*, 2020, 53, 2237–2264. DOI: 10.1007/s10462-019-09732-5.
- [2] Chakraborty, A.; Kar, A. K.: *Swarm Intelligence: A Review of Algorithms*. In: *Nature-Inspired Computing and Optimization: Theory and Applications*. Springer, 2017, 475–494. DOI: 10.1007/978-3-319-50920-4_19.
- [3] Heinzelman, W. R.; Chandrakasan, A.; Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, Maui, HI, USA, 2000. DOI: 10.1109/HICSS.2000.926982.
- [4] Kumar, S.; Agrawal, R.: A comprehensive survey on meta-heuristic-based energy minimization routing techniques for wireless sensor network: classification and challenges. *Journal of Supercomputing*, 2022, 78, 6612–6663. DOI: 10.1007/s11227-021-04128-1.

- [5] Mann, H. B.; Whitney, D. R.: On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, Institute of Mathematical Statistics, 1947, 18(1), 50–60. DOI: 10.1214/aoms/1177730491.
- [6] Mavrovouniotis, M.; Li, C.; Yang, S.: A survey of swarm intelligence for dynamic optimization: Algorithms and applications. *Swarm and Evolutionary Computation*, 2017, 33, 1–17. DOI: 10.1016/j.swevo.2016.12.005.
- [7] Miranda, J.; Abrishambaf, R.; Gomes, T.; Gonçalves, P.; Cabral, J.; Tavares, A.; Monteiro, J.: Path loss exponent analysis in Wireless Sensor Networks: Experimental evaluation. 2013 11th IEEE International Conference on Industrial Informatics (INDIN), Bochum, Germany, 2013, 54–58. DOI: 10.1109/INDIN.2013.6622857.
- [8] Morales, E. F.; Escalante, H. J.: A brief introduction to supervised, unsupervised, and reinforcement learning. In: *Biosignal Processing and Classification Using Computational Learning and Intelligence*. Academic Press, 2022, 111–129.
- [9] Srivastava, A.; Mishra, P. K.: A Survey on WSN Issues with its Heuristics and Meta-Heuristics Solutions. *Wireless Personal Communications*, 2021, 121, 745–814. DOI: 10.1007/s11277-021-08659-x.
- [10] Suk, H. I.: An introduction to neural networks and deep learning. In: *Deep Learning for Medical Image Analysis*. Academic Press, 2017, 3–24.
- [11] Tang, J.; Liu, G.; Pan, Q.: A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends. *IEEE/CAA Journal of Automatica Sinica*, 2021, 8(10), 1627–1643.
- [12] Wang, Y.; He, H.; Tan, X.: Truly proximal policy optimization. In: *Uncertainty in Artificial Intelligence*. PMLR, 2020, 113–122.