
Production Systems and Information Engineering
Volume 11 (2), pp. 114–128.
https://doi.org/10.32968/psaie.2023.2.9

FUZZY SZABÁLYBÁZIS OPTIMALIZÁLÁS A HFRIQ-LEARNING-RENDSZERBEN

TOMPA TAMÁS
Miskolci Egyetem
Informatikai Intézet
Általános Informatikai Intézeti Tanszék
tompa@iit.uni-miskolc.hu

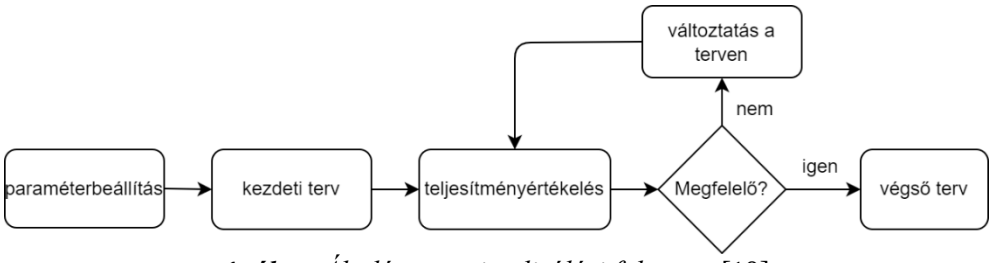
KOVÁCS SZILVESZTER
Miskolci Egyetem
Informatikai Intézet
Általános Informatikai Intézeti Tanszék
szkovacs@iit.uni-miskolc.hu

Absztrakt. A HFRIQ-learning (heurisztikusan gyorsított FRIQ-learning) a ’FIVE’ fuzzy szabályinterpolációs módszeren alapuló Q-tanuló-algoritmus, amelybe szakértő által megadott tudásbázis illeszthető fuzzy produkciós szabályok formájában. A tanulási folyamat során a kezdeti szakértői tudásbázist a rendszer optimalizálja olyan módon, hogy ha szükséges, akkor új fuzzy szabálypontot hoz létre, egyébként pedig a meglévő inkrementális szabályrendszert (Q-függvényt) hangolja. A hangolási folyamat során a szabálypontok pozíciója (antecedense és konzekvense) a gradiens módszer alkalmazása következtében a Q-függvény gradiense által módosul. A cikk célja annak bemutatása, hogy a tanulási folyamat során a szabálypontok optimalizálása hogyan valósul meg a HFRIQ-learning-rendszerben.

Kulcsszavak: megerősítéses tanulás, heurisztikusan gyorsított megerősítéses tanulás, szakértői tudásbázis, fuzzy szabályinterpoláció, Q-learning, FRIQ-learning

1. Bevezetés

Az optimalizálási módszerek célja változók olyan értékeinek megtalálása, amelyek minimalizálnak (vagy maximalizálnak) egy célfüggvényt. Ezen algoritmusok az adott változók értékeit addig módosítják (hangolják), amíg a valamilyen szempontból optimális megoldás elő nem áll. A következő ábra az általános optimalizálási folyamatot szemlélteti:

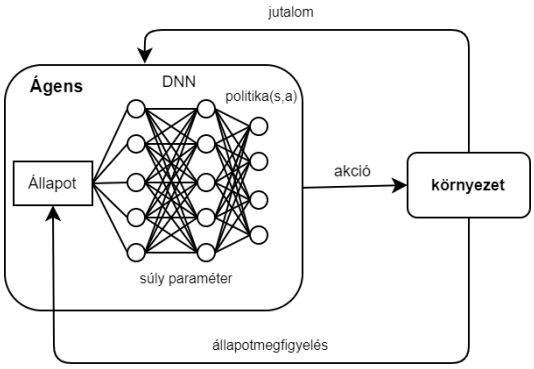


1. ábra. *Általános optimalizálási folyamat* [10]

Az optimalizálási folyamat általában egy függvény minimum- vagy maximumpontjának (szélsőértékének) keresését jelenti. A legtöbb esetben egy hibafüggvény minimumpontjának keresése a cél, azaz a változók azon értékeinek a megtalálása, amely mellett a függvény értéke a legkisebb, majd ez által úgy igyekszik elmozdítani (elhangolni) az adatpontokat, hogy a hiba értéke csökkenjen.

Az iteratív módszerek közül a gradiens módszer (gradient descent, GD – gradiens süllyedés) a legelterjedtebb, amely iterációkon keresztül jut el egy függvény minimumához, a függvény gradiense, azaz az iránymenti deriváltjai által úgy, hogy mindig a legmeredekebb lejtő irányba halad. Gradiens módszer alapú hangolási eljárást a mély tanulás [16] (Deep Learning) alapú „Deep Q-learning Network” (DQN) [26] is alkalmaz. A DQN egy olyan Q-learning-algoritmus, amely a mély tanulás, azon belül is a mély megerősítéses tanulás (Deep Reinforcement Learning) [1] [6] [17] módszerek csoportjába sorolható. A mély megerősítéses tanulás a megerősítéses tanulás és a mély tanulás kombinációja. Ezen módszerek közös jellemzője, hogy általában nagyszámú bemeneti adattal dolgoznak és többrétegű neurális hálózatokat (Deep Neural Network – DNN) alkalmaznak, melyek topológiájában több rejtett réteg is található (innen a „mély” elnevezés). Minden egyes réteg egy másfajta reprezentációját adja a bementi adatoknak, amely egyfajta jellemzőkinyerésnek tekinthető, így rétegről rétegre előrehaladva egyre komplexebb összefüggések kerülnek beazonosításra (például: pixelek → élek → alakzatok → tárgyak). Az elterjedtebb mély megerősítéses tanulási módszerekről bővebb áttekintést az [1] [6] és [17] hivatkozások adnak. Az optimalizálási módszereket a Deep Learning algoritmusok esetében általában a neurális hálózat súlyainak optimalizálására alkalmazzák, amely által az úgy igyekszik módosítani a hálózat súlyainak értékét, hogy a hiba értéke csökkenjen.

A mély megerősítéses tanulás modelljét a következő ábra szemlélteti [16]:



2. ábra. *A mély megerősítéses tanulás modellje* [16]

A DQN-módszer neurális hálózatot alkalmaz a Q-függvény leírására, amely által a Q-függvény a következő módon írható fel [4]:

$$Q(s,a) \approx Q(s,a,\theta) \tag{1}$$

ahol $Q(s,a)$ a Q-függvény, s az állapot, a az akció, a θ pedig a neurális hálózat súlyait reprezentálja. A hiba (L) ebben az esetben a TD-hiba alapján határozható meg, amelyet az adott optimalizálási módszer minimalizálni igyekszik [4]:

$$\mathrm{L} = E\left[\left(r_k + \gamma \max_a Q(s_{k+1}, a_{k+1}, \theta) - Q(s_k, a_k, \theta)\right)^2\right] \tag{2}$$

ahol $r_k + \gamma \max_a Q(s_{k+1}, a_{k+1}, \theta) - Q(s_k, a_k, \theta)$ összefüggés a TD-hiba, k az iteráció (időpillanat), r_k a jutalom, γ diszkontálási tényező, s_k az állapot, a_k az akció, θ pedig a neurális hálózat súlyai.

A klasszikus gradiens módszernek több változata is található a szakirodalomban, ilyen például a sztochasztikus gradiens módszer (Stochastic Gradient Descent – SGD), amely a klasszikus gradiens módszerrel ellentétben nem az összes lehetséges hibafüggvénypontban határozza meg a gradienst és ez által az összes rendelkezésre álló minta alapján hangol minden egyes iterációban, hanem egyesével, véletlenszerűen veszi a mintapontokat iterációnként. Ennek előnye, hogy nagy dimenziószámmal rendelkező problémák esetében csökken a számítási és memóriaigény. A „mini-batch" gradiens módszer a mintaadatok halmazát kisebb részhalmazokra bontja, majd ezen részhalmazok alapján határozza meg a hibát és frissíti (azaz hangolja) a modell paramétereit. A gradiens módszer alapú optimalizálási eljárások hátránya, hogy nem garantálják a globális optimum megtalálását. A gradiens módszerek elterjedtebb változatairól és azok hatékonyságának összehasonlításáról a [7] és [19] irodalmak adnak bővebb áttekintést.

A részecskeraj alapú optimalizálás (Particle Swarm Optimization – PSO) [8] szintén egy iteratív algoritmus, amely működése a keresési térben, általában egyenletes

eloszlás szerint elhelyezett részecskéken (raj) alapszik, amely részecskék matematikai összefüggések alapján mozognak. A részecskék a legjobb pontot keresik a térben, majd a saját legjobb pozíciójuk és a raj legjobb pozíciója alapján mozdulnak el. A legjobb ismert pozíció frissül iterációnként attól függően, hogy a raj talált-e a legjobb ismert pozíciónál még jobbat vagy sem, tehát a részecskék mozgását (és a mozgás irányát) az adott iterációban megtalált legjobb pozíció befolyásolja. Ha a raj már nem talál jobb pozíciót, akkor a leállási feltétel bekövetkezése után a raj legjobbja (részecskéje) adja meg a függvény optimum-, azaz minimumpontját. A PSO-algoritmus esetében, nincs szükség a függvény gradiensének és így annak parciális deriváltjainak meghatározásra (a gradiens módszereknél ez alapkövetelmény), így ez a módszer jól alkalmazható olyan függvények esetében melyek gradiense nem ismert, illetve zajjal terhelt, valamint időben változó problémák esetében is.
További optimalizálási módszerekről a [10] sorszámú szakirodalom, a megerősítéses tanulásban alkalmazott elterjedtebb optimalizálási módszerekről pedig a [18] hivatkozás ad bővebb áttekintést.

2. Heurisztikusan gyorsított FRIQ-learning (HFRIQ-learning)

A heurisztikusan gyorsított FRIQ-learning (Heuristically Accelerated Fuzzy Rule-Interpolation based Q-learning – HFRIQ-learning) [24] a FRIQ-learning (Fuzzy Rule-Interpolation based Q-learning) [28] módszer kiterjesztése, amely által az alkalmas külső szakértői tudásbázis beágyazására [20], illetve hangolására [21]. A módszer a „FIVE” (Fuzzy Rule Interpolation based on Vague Environment) [11] fuzzy szabály-interpolációs eljárást alkalmazza a Q-függvény leírására, amely által az állapot-akció tér folytonos.

2.1. A „FIVE” Fuzzy szabályinterpolációs módszer

A „FIVE” Kovács–Kóczy [11] [12] [14] által kifejlesztett egylépéses szabály-interpolációs módszer, amely az interpolációs feladatot egy úgynevezett bizonytalan környezetbe (Vague Environment, VE) [9] helyezi át. A Klawonn-féle bizonytalan környezet alapgondolata az univerzum elemei közötti hasonlóságon, illetve megkülönböztethetetlenségen alapszik [9]. Az elemek hasonlóságának mértéke súlyozott távolság által definiálható, ahol a súlytényező az $s(x)$ skálafüggvény [9] [12] [13]:

$$s(x) = |\mu'(x)| = \left|\frac{d\mu}{dx}\right| \tag{3}$$

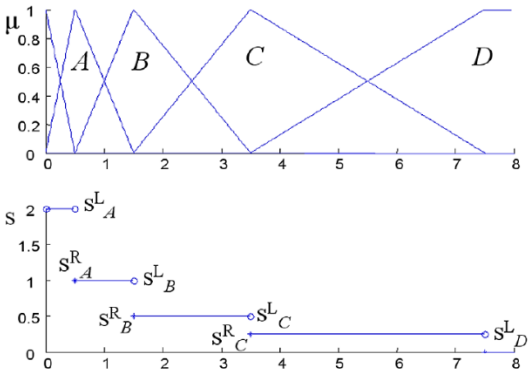
létezik ha: $\min\{\mu_i(x), \mu_j(x)\} > 0 \;\rightarrow\; |\mu'_i(x)| = |\mu'_j(x)|$ (4)

Ahol $\forall i, j \in I$ fuzzy halmazok, $\mu'(x)$ a fuzzy halmaz tagsági függvényének deriváltja, ebben az esetben tehát az $s(x)$ skálafüggvény a $\mu(x)$ tagsági függvény deriváltja, [12]. Az (x_1, x_2) elemek közötti, bizonytalan környezetbeli δ_s távolságuk meghatározásának módja az $s(x)$ skálafüggvényen alapszik [9]:

$$\delta_s(x_1, x_2) = \left| \int_{x_2}^{x_1} s(x)dx \right| \tag{5}$$

Az X bizonytalan környezetben az (x_1, x_2) elemek ε mértékben hasonlónak (megkülönböztethetetlennek) tekinthetőek, ha a közöttük lévő δ_s távolság legfeljebb ε mértékű, azaz $\delta_s(x_1, x_2) \leq \varepsilon$ [12] [13]. A bizonytalan környezetek (antecedens, konzekvens, szabálybázis) előre számolhatók (így biztosítva a módszer gyorsaságát) amelyben minden szabály egy-egy szabálypontként ábrázolható.

Az alábbi 3. ábra fuzzy halmazokat (az ábra felső részében lévő grafikon) és az őket jellemző skálafüggvényeket (az ábra alsó részében lévő grafikon) szemlélteti, amelyek ezáltal alkalmasak az adott fuzzy partíció alakjának leírására [13]:



3. ábra. *Fuzzy halmazok (felső grafikon) és az őket jellemző skálafüggvények (alsó grafikon)* [13]

Minél kisebb a skálafüggvény értéke, az elemek annál kevésbé megkülönböztethetők egymástól, azaz ugyanolyan alaphalmazbeli távolság esetében egyre közelebb vannak egymáshoz. Ha a skálafüggvény értéke nulla, az azt eredményezi, hogy az elemek nem megkülönböztethetőek, mert egyformán közel helyezkednek el. Egy valós gyakorlati alkalmazás esetében ez azt eredményezi, hogy például minden olyan esetben, amikor 2 méternél messzebb találhatok objektumok, akkor nem kell fékezni, amikor pedig ennél közelebb vannak, akkor egyre jobban kell fékezni.

A „FIVE”-módszer a multidimenziós mivolta következtében a Shepard interpolációs operátort [5] alkalmazza, amely által a singleton (egyértékű) c_k konzekvens a következő összefüggés alapján, további defuzzifikációs lépések nélkül határozható meg [13]:

ha $x = a_k$ minden k-ra,

$$y(\boldsymbol{x}) = \begin{cases} c_k & \\ \sum_{k=1}^{r}\left(\left({c_k}/{\delta_{s,k}^{\lambda}}\right)/\left(\sum_{k=1}^{r}{1}/{\delta_{s,k}^{\lambda}}\right)\right) & \text{egyébként} \end{cases} \tag{6}$$

Ahol $\boldsymbol{x}$ a többdimenziós megfigyelés, c_k a k-adik létező szabály konzekvense, r a szabályok száma az R szabálybázisban, λ a Shepard-kivető, $\delta_{s,k}$ pedig a súlyozott (euklideszi) távolság, amely a következő formával definiálható [12] [13][13]:

$$\delta_{s,k} = \delta_s(\boldsymbol{a_k}, \boldsymbol{x}) = \left[\sum_{i=1}^{m}\left(\int_{a_{k,i}}^{x_i} s_{x_i}(x_i)dx_i\right)^2\right]^{1/2} \tag{7}$$

Ahol $\boldsymbol{x}$ az m-dimenziós crisp megfigyelés, $\boldsymbol{a_k}$ a magja az m-dimenziós szabály antecedens $\boldsymbol{A_k}$-nak, s_{x_i} pedig az i-edik skálafüggvény az m-dimenziós antecedens univerzumban.

A „FIVE" [13] tehát egy alkalmazásorientált FRI (Fuzzy Rule Interpolation) módszer, amely alacsony számítási igénye miatt [2] [3] jól használható valós idejű alkalmazásokban, illetve robotikai irányításokban. Továbbá, ezen alkalmazott FRI-módszer által a rendszer komplexitása csökkenthető a ritka szabálybázis következtében, illetve a rendszer abban az esetben is szolgáltat kimenetet, mikor a klasszikus fuzzy következtetési eljárások nem [15].

2.2. A HFRIQ-learning

A HFRIQ-learning-rendszer tudásbázisa egy ritka fuzzy szabálybázis által leírt, egy r_i ($i \in [1, m]$) szabály formája az m méretű R szabálybázisban a következő [28]:

$$r_i\text{: } \boldsymbol{If}\ s_1\ is\ S_1^i\ \boldsymbol{And}\ s_2\ is\ S_2^i\ \boldsymbol{And}\ \ldots\ \boldsymbol{And}\ s_n\ is\ S_n^i\ \ \boldsymbol{And}\ a\ is\ A^i\ \boldsymbol{Then}\ \tilde{Q}(\boldsymbol{s}, a) = q^i \tag{8}$$

ahol S_j^i az i-edik ($i \in [1, m]$) szabály j-edik ($j \in [1, n]$) állapotdimenziójának fuzzy halmaza az n-dimenziós $\boldsymbol{S}$ állapottérben, $\boldsymbol{s} \in \boldsymbol{S}$ az n-dimenziós állapotmegfigyelés, s_j a j-edik dimenziója az $\boldsymbol{s}$ állapotmegfigyelésnek, A^i az i-edik szabály egydimenziós akció univerzumának (U) fuzzy halmaza, $a \in U$ az akció, $\tilde{Q}(\boldsymbol{s}, a)$ a FIVE FRI [13] által becsült Q-függvény, q^i pedig az i-edik szabály konzekvense (Q-értéke).

Az R_{expert} szakértői tudásbázis formátuma hasonló a (8) formula által definiált fuzzy szabályokhoz, azzal az eltéréssel, hogy az $\hat{r}$ szakértői szabályok antecedense az állapot, konzekvense pedig az ebben az állapotban preferált akció [20]:

$$\hat{r}_i\text{: } \textbf{If}\ s_1 \boldsymbol{is}\ \hat{S}_1^i\ \boldsymbol{And}\ s_2 \boldsymbol{is}\ \hat{S}_2^i\ \boldsymbol{And}\ \ldots \boldsymbol{And}\ s_n \boldsymbol{is}\ \hat{S}_n^i\ \boldsymbol{Then}\ \text{a} = \hat{A}^i \tag{9}$$

ahol $\hat{r}_i$ az i-edik ($i \in [1, \hat{m}]$) szakértői szabály az R_{expert} szabálybázisban, $\hat{S}_n^i =$

$\left[\hat{S}_1^i, \hat{S}_2^i, \ldots \hat{S}_n^i\right]$ az i-edik szakértői szabály n-dimenziós állapot megfigyelése, $\hat{A}^i$ az ehhez az $\hat{S}_n^i$ állapotmegfigyeléshez tartozó akció, i ($i \in [1, \hat{m}]$) pedig a szabály indexe az $\hat{m}$ méretű szakértői szabályrendszerben.

Annak következtében, hogy a szakértői szabályrendszer injektálható legyen a rendszerbe szükséges a formátumának átalakítása állapot-akció-Q-érték formátumra. Ekkor az átalakított szakértői szabályok antecedense az állapot-akció, konzekvense pedig egy becsült $\tilde{Q}_{init}$ érték lesz. A becsült kezdeti $\tilde{Q}_{init}$ érték a környezet által maximálisan adható megerősítés (g_{max}) ismeretében határozható meg [20]. A HFRIQ-learning-rendszer tanulási folyamata ezen R_{expert} szakértői szabályrendszer és a 2^{n+1} darabszámú (n: állapotdimenziók száma), 0 konzekvensértékkel rendelkező $r_i^{\square}$ sarokponti szabályok összefésülésével létrejött fuzzy szabálybázissal kezdődik [20] [20][28]. Abban az esetben, ha ellentmondás alakul ki, azaz szakértői szabály sarokponti szabályra illeszkedik (de eltérő a konzekvensük), akkor az ellentmondás feloldása következtében ezen két szabály összevonásra kerül egyetlen szabállyá. A létrejött kezdeti szakértői szabályrendszer a tanulási folyamat során inkrementálisan növekszik a rendszer által létrehozott szabályokkal [29] [28]. Új szabály akkor kerül beillesztésre a szabálybázisba, ha a Q-frissítés ($\Delta\tilde{Q}$) értéke nagyobb, mint egy ε_Q érték ($\Delta\tilde{Q} > \varepsilon_Q$) [29] és a legközelebbi szabálypont is távolinak tekinthető [23] [24]. A szabályközelség meghatározásának az alapja a szabályok között definiált, dimenziónként számított távolságok [23] [24]. Abban az esetben, ha a $\Delta\tilde{Q}$ érték kicsi ($\Delta\tilde{Q} < \varepsilon_Q$), akkor a teljes szabálybázis konzekvense kerül frissítésre a következő módon [29]:

$$\tilde{Q}^{k+1}(\boldsymbol{s}, a) = \tilde{Q}^k(\boldsymbol{s}, a) + \Delta\tilde{Q}^{k+1}(\boldsymbol{s}, a) \tag{10}$$

$$\Delta\tilde{Q}^{k+1}(\boldsymbol{s}, a) = \alpha * \left(g(\boldsymbol{s}, a, \boldsymbol{s}') + \gamma * \max_{a' \epsilon U} \tilde{Q}^k(\boldsymbol{s}', a') - \tilde{Q}^k(\boldsymbol{s}, a)\right) \tag{11}$$

ahol $\gamma \in [0,1]$ a leszámítolási tényező, $\alpha \in [0,1]$ a tanulási ráta, a pedig az $\boldsymbol{s}$-ben végrehajtott akció. Az új megfigyelt állapot $\boldsymbol{s}'$, $g(\boldsymbol{s}, a, \boldsymbol{s}')$ a megfigyelt jutalom az $\boldsymbol{s} \rightarrow \boldsymbol{s}'$ állapotátmenetre, $\tilde{Q}^k$ és $\tilde{Q}^{k+1}$ pedig a k-adik és a $(k+1)$-edik iteráció FIVE FRI-módszer által becsült Q-értéke [29]:

$$\tilde{Q}(\boldsymbol{s}, a) = \begin{cases} q^i & \text{ha } (\boldsymbol{s}, a) = (\boldsymbol{s}^i, a^i) \text{ valamennyi } i - re, \\ \sum_{i=1}^{m} \left(\left({q^i}/{(\delta_v^i)^\lambda} \right) \Big/ \left(\sum_{j=1}^{m} {1}/{(\delta_v^j)^\lambda} \right) \right) & \text{egyébként} \end{cases} \tag{12}$$

ahol q^i az i-edik ($i \in [1, m]$) szabály konklúziója, ($\mathbf{s}$, a) a megfigyelés, δ_v^i a skálázott távolság [13] az ($\mathbf{s}$, a) megfigyelés és az i-edik szabály ($\boldsymbol{s}^{\boldsymbol{i}}, a^i$) antecedense között, λ a Shepard-paraméter, m pedig a szabályok száma.

Ha a $\Delta\tilde{Q}$ értéke kicsi és van már létező szabály a megfigyelés közelében, akkor a

gradiens módszer alapú hangolási eljárás a megfigyeléshez legközelebb elhelyezkedő szabálypont antecedensét és konzekvensét fogja hangolni [24]. A szabálypont új pozíciója az alábbi módon kerül meghatározásra [24]:

$$\boldsymbol{s}_{k+1} = \boldsymbol{s}_k - \left(2 * TDerror * \frac{\partial \tilde{Q}(\boldsymbol{s}, a)}{\partial \boldsymbol{s}}\right) * \alpha \tag{13}$$

$$a_{k+1} = a_k - \left(2 * TDerror * \frac{\partial \tilde{Q}(\boldsymbol{s}, a)}{\partial a}\right) * \alpha \tag{14}$$

$$q_{k+1} = q_k - \left(2 * TDerror * \frac{\partial \tilde{Q}(\boldsymbol{s}, a)}{\partial q}\right) * \alpha \tag{15}$$

ahol a $\boldsymbol{s}_{k+1}, a_{k+1}, q_{k+1}$ a gradiens-módszer által meghatározott új állapot, akció- és Q-értékek, $\boldsymbol{s}_k, a_k, q_k$ a régi állapot, akció- és Q-értékek, α a gradiens-módszer tanulási rátája, $\frac{\partial \tilde{Q}(\boldsymbol{s},a)}{\partial \boldsymbol{s}}, \frac{\partial \tilde{Q}(\boldsymbol{s},a)}{\partial a}, \frac{\partial \tilde{Q}(\boldsymbol{s},a)}{\partial q}$ a Q-függvény állapot, akció és Q-érték szerinti parciális deriváltjai, a $TDerror$ értéke pedig a következő [24]:

$$TDerror = g(\boldsymbol{s}, a, \boldsymbol{s}') + \gamma * \max_{a' \epsilon U} \tilde{Q}^k(\boldsymbol{s}', a') - \tilde{Q}^k(\boldsymbol{s}, a) \tag{16}$$

Az alkalmazott hangolási módszer következtében, a szabálypontok vándorlása miatt előfordulhat olyan eset, hogy több szabálypont is közel kerül egymáshoz. Ebben az esetben az egymáshoz közel kerülő és ez által hasonló információt leíró szabálypontok egyesítésre kerülnek egyetlen szabállyá [23] [24], ami által a szabálybázis mérete a tanulási folyamat csökkenthető. A szabálytávolság alapú szabálybázis redukálási módszerről bővebb információ a [23] [24] [25] hivatkozásokban található. További, a tanulási folyamat után opcionálisan alkalmazható szabálybázis csökkentési módszereket a [22] [27] [30] hivatkozások mutatnak be.

A HFRIQ-learning tanulási folyamata akkor ér véget, ha nem kerül új szabály hozzáadásra az inkrementális szabálybázisba, a Q-frissítés értéke elenyészően kicsi, a létező szabálypontok pozíciója nem változik, és nem kerülnek szabályok összevonásra.

3. Fuzzy szabálypontok hangolása a HFRIQ-learning-rendszerben

A szakértő által definiált a priori szabályrendszerben amennyiben a megadott szakértői produkciós szabály valamelyik állapothoz nem megfelelő akcióértéket rendel (azaz a szakértő szabályrendszer csak részben tekinthető helyesnek), úgy a szabálybázisba felvett szakértői szabály állapot-akció antecedense is rossz helyre kerül. Ebben az esetben, a csak részben helyes szabályrendszer negatív hatással lehet a tanulási folyamat hatékonyságára [20] [20][21], így szükség lehet egy hangolási eljárásra,

amely képes a szabályok állapot-akció pontját, azaz antecedensét elhangolni (optimalizálni) a megfelelő irányba, szabálypontba.

3.1. Hangolandó szabálypontok meghatározása

Az FRIQ-learning Q-függvény tartópontjainak hangolása (optimalizálása) során meghatározandó, hogy mely szabálypontok kerüljenek optimalizálásra a tanulási fázis során. A módszerben [20] [24] a Q-függvény a fuzzy szabálypontok „FIVE" FRI-módszer fuzzy interpolációjával állnak elő.

A gradiens módszernek több verziója ismert a bemutatottak alapján. A GD-módszer minden egyes iterációban minden egyes mintapontot figyelembe vesz a hibafüggvény, illetve a deriváltak számításakor, míg az SGD-módszer egyesével (véletlenszerűen) veszi a mintapontokat, majd ezek alapján határozza meg a gradienst és hibafüggvényt iterációnként. Jelen esetben mintapontoknak a szabálypontok tekinthetők, melyek hangolása a TD-hiba (16) értékének függvényében történik.

A fuzzy szabályok antecedens- és konzekvensértékeinek hangolása a $x_{k+1} = x_k - \nabla F(x_k) * \alpha$ összefüggés alapján történik, ahol az $F(x_k)$ függvény parciális deriváltja ($\nabla F(x_k)$ gradiens) a láncszabály alkalmazásával a következőképpen határozható meg:

$$\nabla F(x_k) = \frac{\partial MSE(x_k)}{\partial x_k} = \frac{\partial (TDerror)^2}{\partial x_k} = 2 * TDerror * \frac{\partial \tilde{Q}^k(\boldsymbol{s}, a)}{\partial x_k} \quad (17)$$

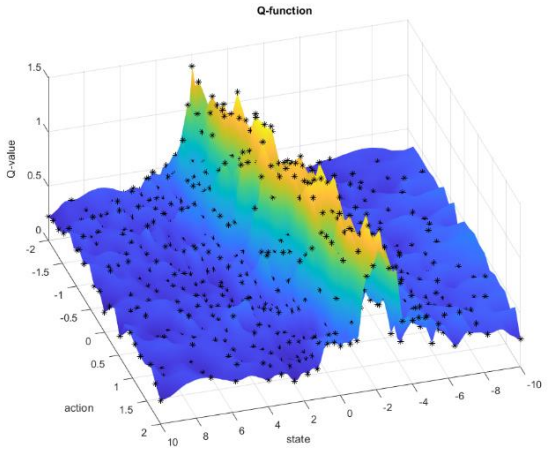
A tanulási folyamat során a szabálypontok abban az esetben hangolódnak, ha a Q-frissítés-értéke nagynak tekinthető $(\Delta\tilde{Q} > \varepsilon_Q)$ és az éppen aktuális állapot-akció megfigyelés közelében (meghatározott közelségmérték alapján [23] [24] [25]) található már létező szabály.

Annak vizsgálata, hogy az összes szabálypont antecedens és konzekvens értékeinek hangolása a tanulási folyamat során milyen hatással van a Q-függvényre egy egyszerű, egy állapot- és egy akciódimenzióval rendelkező mintapéldán történik. A mintapélda paraméterei a következők:

- állapotváltozó: $s_1 \in [-10{,}10]$
- akcióváltozó: $a \in [-2{,}2]$
- $\alpha = 0{,}5$
- $\gamma = 0{,}4$
- $\varepsilon = 0{,}5$
- $g_{max} = 1$
- szakértői szabály: 0 állapotpontban 0 értékű akció
- jutalomfüggvény: a környezet +1 jutalmat ad, ha az ágens állapotváltozójának értéke –1 és +1 közötti, ellenkező esetben a jutalom értéke 0

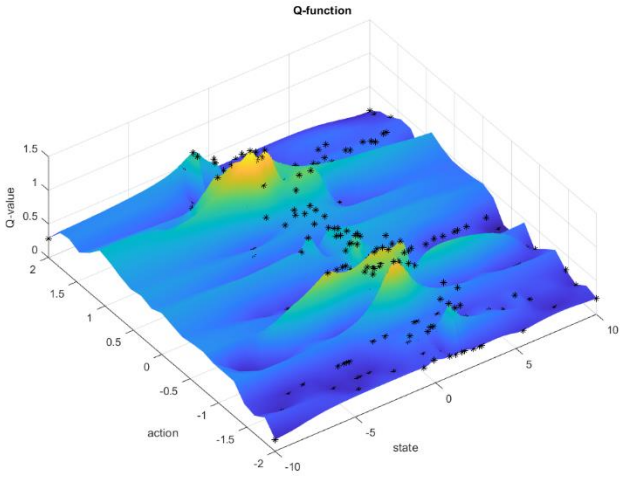
Az 1. ábrán az összehasonlítás alapjául szolgáló, a gradiens módszeren alapuló hangolási eljárás [24] alkalmazása nélkül kapott „referencia" Q-függvény felülete látható, ahol a „*" (csillag) karakterek a szabálypontokat (530 darab) jelölik, a szabályok között megengedett minimális távolság pedig az univerzumok hosszának

100-ad része:



4. ábra. *Az összehasonlítás alapjául szolgáló „referencia” Q-függvény felülete*

Az első vizsgált esetben a javasolt hangolási módszerrel [24] az összes szabálypont hangolásra kerül, beleértve a szakértői szabályokat is. Az ebben a futási esetben kapott Q-függvény felületét a 5. ábra szemlélteti:



5. ábra. *Összes létező szabálypont hangolása esetében kapott Q-függvény felülete*

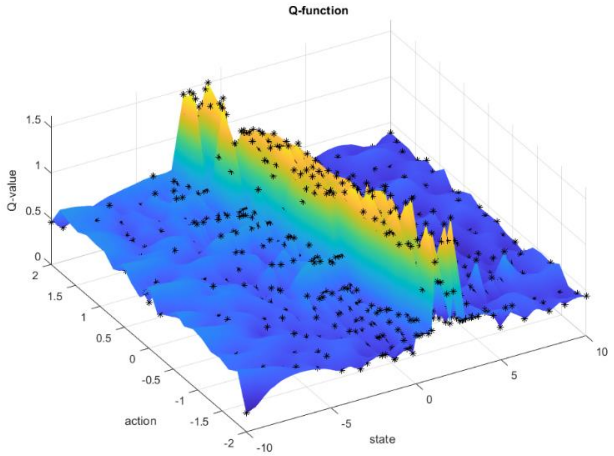
Megállapítható, hogy az összes szabálypont hangolásával a 4. ábrán látható referenciafüggvény felületéhez képest a függvény felülete romlott, nem alakult ki a felület „gerince”, több kisebb csúcs keletkezett csupán. Ennek oka az lehet, hogy az állapot-akció tér azon pontjainak antecedense és Q-értéke, melyek ritkán kerülnek bejárásra

és így frissítésre (a felderítés-kiaknázás következtében), az összes szabálypont hangolása miatt elromlik. A kiaknázás során bejárt út frissítései hatással vannak a ritkán, csak a felderítés során tesztelt területekre. A visszajelzés nélküli (kevésbbé bejárt) területek Q-értéke a gyakorta bejárt területek átlag Q-értéke felé hangolódik. Emiatt egyes szabályok Q-következményértéke elromlik, azaz rossz irányban módosul.
Az összes szabálypont egyidejű hangolása ezért nem járható út. Megoldás lehet az, hogy csak azon szabályok kerüljenek hangolásra, amely az éppen aktuális állapot-akció megfigyelési pont közelében találhatók, azaz a legközelebb helyezkednek el ahhoz. A javasolt módszer megoldja a fentebb említett problémát és helyes irányban hangolja a Q-függvény szabálypontjait, nem változtatva azon szabályokat, melyek az aktuális állapot-akció megfigyelési ponttól távol, a kevésbé látogatott területekre esnek. A javasolt módszerben a közelség mértékének meghatározása a [23] [24] [25] publikációkban bemutatottak alapján történik.

A hangolási folyamat során tekintettel kell lenni a hangolandó (az éppen közeli) szabálypont típusára. A sarokponti típusú szabályok antecedensei kötöttek (az állapot-akció tér univerzum-hiperkocka sarokpontjai), a hangolás során nem változhatnak. A hangolás ezért a sarokponti típusú szabályok esetében csak a szabályok konzekvensére vonatkozik. A szakértői és a rendszer által felvett típusú szabályok esetén a szabályok antecedense (állapot-akció pontja) és konzekvense (Q-értéke) is hangolásra kerül.

Abban az esetben, ha a tanulási folyamat során valamely szakértői szabály antecedense kerül hangolására, akkor ezen produkciós (állapot-akció) szakértői szabály módosul. Abban az esetben, ha a hangolási folyamat során az antecedens nem, vagy csak kismértékben változik, akkor a szakértői szabály a környezeti tesztek által igazoltnak, helyesen definiált szakértői szabálynak tekinthető.

A 6. ábrán az az eset látható, mikor nem az összes szabálypont, hanem csak a megfigyeléshez közeli szabályok kerültek hangolásra. Ebben az esetben a függvény felülete nem romlott el, a függvény gerince viszonylag összefüggően kialakult, a függvény formája a referencia Q-függvényhez hasonló lett, azzal a különbséggel, hogy az kisimult, a felülete simább lett (a gradiensalapú hangolásnak köszönhetően).



6. ábra. *Csak a megfigyeléshez legközelebbi szabálypontok hangolása esetében kapott Q-függvény felülete*

A kapott Q-függvény felületeken sok olyan szabálypont található (a „*” karakter által jelölt pozíciókban), melyek viszonylag egymáshoz közel (és sűrűn) helyezkednek el. Ezeken az állapot-akció területeken a gradiens módszer alapú hangolási eljárásnak köszönhetően a közeli szabályok egyre közelebb kerülnek egymáshoz. A hangolási folyamat következő lépése az, hogy az egymáshoz viszonylag közel kerülő szabályok, a szabályok közötti távolság, illetve távolságküszöbök alapján kerülnek összevonásra (egyesítésre), a szabályok számának csökkentése érdekében [23] [24] [25].

4. Összefoglalás

Bemutatásra került a HFRIQ-learning-rendszer gradiens módszer alapú szabálybázis-optimalizálási módszere. Mintapélda segítségével igazolásra került, hogy az összes szabálypont egyidejű hangolása negatív hatással van a Q-függvényre annak következtében, hogy a kiaknázás során bejárt út frissítései hatással vannak a ritkán, csak a felderítés során tesztelt területekre. Így a kevésbbé bejárt területek Q-értéke a gyakorta bejárt területek átlag Q-értéke felé hangolódik, ami miatt az egyes szabályok Q-következményértéke elromlik, azaz rossz irányban módosul. Bemutatásra került egy javasolt módszer, amely ezt a problémát kiküszöböli. A bemutatott módszer következtében a HFRIQ-learning megerősítéses tanulási rendszer tudásbázisának hangolása során az állapot-akció tér ritkán bejárt területein lévő fuzzy Q-szabályok elhangolódása csökkenthető, ha az összes fuzzy szabálypont egyidejű hangolása helyett, csak azon szabályok kerülnek hangolásra, amelyek az éppen aktuális állapot-akció megfigyelési pont közelében találhatók.

5. Köszönetnyilvánítás

„A Kulturális és Innovációs Minisztérium ÚNKP-23-4-I-ME/5 kódszámú Új Nemzeti Kiválóság Programjának a Nemzeti Kutatási, Fejlesztési és Innovációs Alapból finanszírozott szakmai támogatásával készült.”

Irodalomjegyzék

[1] Arulkumaran, Kai, et al. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34, 6, pp. 26–38. https://doi.org/10.1109/msp.2017.2743240

[2] Bartók, Roland & József Vásárhelyi (2022). Design of a FPGA accelerator for the FIVE fuzzy interpolation method. *International Journal of Computer Applications in Technology*, 68,4, pp. 321–331. https://doi.org/10.1504/ijcat.2022.125185

[3] Bartók, Roland & József Vásárhelyi (2019). Examining Cache Handling of the FIVE Method on Multicore Systems. *2019 IEEE 17th World Symposium on Applied Machine Intelligence and Informatics (SAMI). IEEE*. https://doi.org/10.1109/sami.2019.8782721

[4] Brunton, Steven L. & Kutz, J. N. (2019). Data-driven science and engineering: Machine learning, dynamical systems, and control. Cambridge University Press. https://doi.org/10.1017/9781009089517

[5] Shepard, D. (1968). A two dimensional interpolation function for irregularly spaced data. *Proc. 23rd ACM Internat. Conf.*, pp. 517–524. https://doi.org/10.1145/800186.810616

[6] François-Lavet, Vincent et al. (2018). An introduction to deep reinforcement learning. arXiv preprint arXiv:1811.12560. https://doi.org/10.1561/9781680835397

[7] Haji, Saad Hikmat & Adnan Mohsin Abdulazeez (2021). Comparison of optimization techniques based on gradient descent algorithm: A review. *PalArch’s Journal of Archaeology of Egypt/Egyptology*, 18, 4, pp. 2715–2743.

[8] Kennedy, James & Russell Eberhart (1995). Particle swarm optimization. *Proceedings of ICNN'95-international conference on neural networks*. Vol. 4. IEEE. https://doi.org/10.1109/icnn.1995.488968

[9] Klawonn, F. (1994). Fuzzy Sets and Vague Environments. *Fuzzy Sets and Systems*, Vol. 66, pp. 207–221. https://doi.org/10.1016/0165-0114(94)90311-5

[10] Kochenderfer, Mykel J. & Wheeler, T. A. (2019). *Algorithms for optimization*. Mit Press.

[11] Kovács, Sz., Kóczy, L. T. (1997). Approximate Fuzzy Reasoning Based on Interpolation in the Vague Environment of the Fuzzy Rule base as a Practical

Alternative of the Classical CRI. *Proceedings of the 7th International Fuzzy Systems Association World Congress*, Prague, Czech Republic, pp. 144–149.

[12] Kovács, Sz., Kóczy, L. T. (1997). The use of the concept of vague environment in approximate fuzzy reasoning. *Fuzzy Set Theory and Applications.* Tatra Mountains Mathematical Publications, Mathematical Institute Slovak Academy of Sciences, Bratislava, Slovak Republic, Vol.12, pp. 169–181.

[13] Kovács, Sz. (2006). Extending the Fuzzy Rule Interpolation 'FIVE' by Fuzzy Observation. In: Reusch, B. (ed.). Computational Intelligence, Theory and Applications, – Advances in Soft Computing, Springer Germany, pp. 485–497. https://doi.org/10.1007/3-540-34783-6_48

[14] Kovács, Sz. (1996). New Aspects of Interpolative Reasoning. *Proceedings of the 6th. International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Granada, Spain, pp. 477–482.

[15] Kovacs, Szilveszter (2006). Fuzzy Rule Interpolation in Practice. *SCIS & ISIS SCIS & ISIS 2006*. Japan Society for Fuzzy Theory and Intelligent Informatics.

[16] LeCun, Yann, Bengio, Yoshua & Hinton, Geoffrey (2015). Deep learning. *Nature*, 521, 7553, pp. 436–444. https://doi.org/10.1038/nature14539

[17] Li, Yuxi (2017). *Deep reinforcement learning: An overview*. arXiv preprint arXiv:1701.07274. https://doi.org/10.48550/arXiv.1701.07274

[18] Mazyavkina, N., Sviridov, S., Ivanov, S. & Burnaev, E. (2021). Reinforcement learning for combinatorial optimization: A survey. *Computers & Operations Research*, 134, 105400. https://doi.org/10.1016/j.cor.2021.105400

[19] Santra, Santanu, Hsieh, Jun-Wei & Lin, Chi-Fang (2021). Gradient descent effects on differential neural architecture search: A survey. *IEEE Access*, 9, pp. 89602–89618. https://doi.org/10.1109/access.2021.3090918

[20] Tompa, Tamás & Szilveszter Kovács (2020). Applying Expert Heuristic as an a Priori Knowledge for FRIQ-Learning. *Acta Polytechnica Hungarica*, 17, 4. https://doi.org/10.12700/aph.17.4.2020.4.2

[21] Tompa, Tamás & Kovács, Szilveszter (2023). Benchmark example for the Heuristically accelerated FRIQ-learning. *2023 24th International Carpathian Control Conference (ICCC). IEEE*. https://doi.org/10.1109/iccc57093.2023.10178919

[22] Tompa, Tamás & Kovács, Szilveszter (2017). Clustering-based fuzzy knowledgebase reduction in the FRIQ-learning. *2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI), IEEE*. https://doi.org/10.1109/sami.2017.7880302

[23] Tompa, Tamás & Kovács, Szilveszter (2018). Determining the minimally allowed rule-distance for the incremental rule-base contruction phase of the FRIQ-learning. *2018 19th International Carpathian Control Conference (ICCC), IEEE*. https://doi.org/10.1109/carpathiancc.2018.8399677

[24] Tompa, Tamás & Kovács, Szilveszter (2022). Heuristically accelerated FRIQ-

learning. *20th Jubilee International Symposium on Intelligent Systems and Informatics (SISY 2022), IEEE*. https://doi.org/10.1109/iccc57093.2023.10178919

[25] Tompa, Tamás & Kovács, Szilveszter (2023). Tudásbázis redukálás a heurisztikusan gyorsított FRIQ-learning rendszerben. *Production Systems and Information Engineering*, 11, 2, pp. 1–12. https://doi.org/10.32968/psaie.2022.4.4

[26] Mnih, V., Kavukcuoglu, k., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G. et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518 (7540), p. 529. https://doi.org/10.1038/nature14236

[27] Vincze, Dávid, Tóth, Alex & Niitsuma, Mihoko (2020). Antecedent redundancy exploitation in fuzzy rule interpolation-based reinforcement learning. *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. https://doi.org/10.1109/aim43001.2020.9158875

[28] Vincze, Dávid & Kovács, Szilveszter (2009). Fuzzy rule interpolation-based Q-learning. *2009 5th International Symposium on Applied Computational Intelligence and Informatics. IEEE*. https://doi.org/10.1109/saci.2009.5136311

[29] Vincze, Dávid & Kovács, Szilveszter (2010). Incremental rule base creation with fuzzy rule interpolation-based Q-learning. *Computational Intelligence in Engineering*. Springer, Berlin, Heidelberg, pp. 191–203. https://doi.org/10.1007/978-3-642-15220-7_16

[30] Vincze, Dávid & Kovács, Szilveszter (2015). Rule-base reduction in Fuzzy Rule Interpolation-based Q-learning. *Recent Innovations in Mechatronics*, 2, 1–2, pp. 1–6. https://doi.org/10.17667/riim.2015.1-2/10.