Johanyák, Zs. Cs., Kovács, Sz.: Fuzzy rendszer generálása szabálybázis bővítéssel, AGTEDU 2007, 2007 november 8, Kecskemét, ISSN: 1586-846x, pp. 241-246. http://www.johanyak.hu

FUZZY RENDSZER GENERÁLÁSA SZABÁLYBÁZIS BŐVÍTÉSSEL

FUZZY SYSTEM GENERATION BY RULE BASE EXTENSION

Johanyák Zsolt Csaba[1], Kovács Szilveszter[2]
[1]Kecskeméti Főiskola, GAMF Kar, Kalmár Sándor Informatikai Intézet, Informatika Szakcsoport

ÖSSZEFOGLALÁS

A fuzzy rendszerek kialakításának egyik legkritikusabb szakasza a partíciók és a szabálybázis összeállítása. Dolgozatunk azzal az esettel foglalkozik, amikor bemeneti és kimeneti adatpárok formájában rendelkezésre álló tanító halmaz alapján alakítjuk ki a rendszert. Ilyenkor két egymásnak ellentmondó igény kielégítése a cél. Ezek a következők: az approximációs képesség maximalizálása és a szabályok számának minimális szinten tartása. Cikkünkben bemutatunk két olyan általunk kidolgozott módszert, amely néhány kezdő szabály definiálása után egy hangolási algoritmus keretében fokozatosan bővíti a szabálybázist.

ABSTRACT

The identification of the partitions and the rule base is one of the crucial steps in course of the generation of a fuzzy system. One has to fulfil two contradictory requirements: maximum approximation capability of the system and low number of rules. This paper presents two novel methods, which try to fulfil these requirements in two steps. First two initial rules are defined and next the rule base is extended gradually by an iterative tuning algorithm.

KULCSSZAVAK/KEYWORDS
RBE-DSS, RBE-SI, ritka szabálybázis
RBE-DSS, RBE-SI, sparse rule base

BEVEZETÉS

A fuzzy rendszerek kutatásának egyik fontos kérdése a mintaadatok alapján történő automatikus modellgenerálás. A kompozíciós elven alapuló klasszikus fuzzy következtetési módszerek (pl. Zadeh, Mamdani) megkövetelik a szabálybázis fedő jellegét, azaz minden lehetséges bemenő érték (megfigyelés) esetén rendelkezésre kell álljon legalább egy olyan szabály, amelynek antecedens része metszi vagy átfedi a megfigyelést minden bemeneti dimenzióban.

Az antecedens dimenziók és a bennük megjelenő nyelvi értékek számának emelkedése az antecedens tér lefedéséhez szükséges szabályszám robbanásszerű növekedésével jár [4]. A problémára megoldást szolgáltathat a ritka (nem fedő) szabálybázisok és a szabály-interpoláción alapuló következtetési módszerek alkalmazása [4].

Cikkünkben két olyan általunk kifejlesztett automatikus rendszergenerálási módszert mutatunk be, amelyek egy kétlépéses folyamat keretében iteratív hegymászó hangolási algoritmust alkalmazva fokozatosan bővítik a szabálybázist. Ezáltal a kívánt approximációs pontosságot alacsony szabályszám mellett sikerül elérni.

RITKA SZABÁLYBÁZIS AUTOMATIKUS GENERÁLÁSA

A fuzzy modell kialakítása során ritka szabálybázist alapvetően két módon nyerhetünk. Az első megközelítés [4] egy teljesen fedő szabálybázisból indul ki, és a nem releváns szabályok elhagyásával csökkenti a tudásbázis komplexitását.

A második megközelítés közvetlenül kíván előállítani olyan szabálybázist, ami nem fedi teljes mértékben az antecedens teret. Az ebbe a csoportba tartozó módszerek általában az alábbi két megközelítés egyikével dolgoznak.

- Az ún. optimális fuzzy szabályok azonosítására törekszenek (pl. [3]).
- Fuzzy klaszterezés segítségével hozzák létre a szabályokat (pl. [5]).

A továbbiakban az általunk kidolgozott új (harmadik) megközelítést alkalmazó RBE elvet és a rá alapozott RBE-DSS valamint RBE-SI eljárásokat ismertetjük.

RENDSZERGENERÁLÁS ITERATÍV SZABÁLYBÁZIS KITERJESZTÉSSEL

A módszerek ismertetése során csak egykimenetű rendszerekkel foglalkozunk, a többkimentű rendszerek szabálybázisát egykimenetű rendszerek szabálybázisai aggregációjának tekintjük.

Bár a módszerek nem kötődnek egy adott halmazalak típushoz, de az egyszerűség kedvéért széleskörű alkalmazása következtében trapéz alakú nyelvi értékekkel fogunk dolgozni. A továbbiakban adatsornak nevezzük azt az összetartozó $x^k = \{x_1^k, ..., x_N^k, x_{N+1}^k\}, k = 1, ..., M$ adathalmazt, amely minden antecedens dimenzióban tartalmaz egy bemeneti értéket ($x_1^k, ..., x_N^k$), valamint tartalmazza az adott bemenetek esetén elvárt egyetlen kimeneti értéket (x_{N+1}^k). Az előzőekben *M*-el jelöltük a mintaadathalmaz adatsorainak számát, *N*-el az antecedens dimenziók számát, és az egyszerűbb jelölés érdekében a konzekvens dimenzióra *N+1*-ként hivatkoztunk.

A SZABÁLYBÁZIS KITERJESZTÉS ELVE

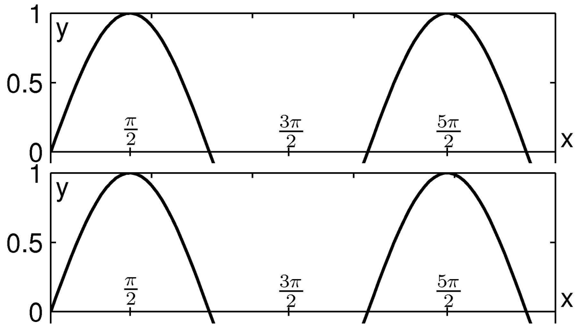
A szabálybázis kiterjesztés (RBE-Rule Base Extension) [2] elve azt mondja ki, hogy a fuzzy rendszert két lépésben hozzuk létre. Az első lépésben két szabályt képzünk, egyet a maximális és egyet a minimális kimenet leírására. A második lépésben egy iteratív hangolási folyamat keretében azonosítjuk a nyelvi értékek paramétereit, valamint új fuzzy halmazokat és szabályokat hozunk létre a rendszer teljesítményének javítása érdekében.

Az első lépést minden dimenzióban az alapértelmezett mag ($W_{C,i}$) és tartó ($W_{S,i}$) szélességek meghatározásával kezdjük. Értékeiket az adott dimenzióban érvényes értelmezési tartomány szélessége (terjedelem -DR_i) arányában határozzuk meg. A tartomány alsó ($x_{i,min}$) és felső ($x_{i,max}$) korlátait előírt értékek vagy a rendelkezésre álló mintaadatok határozzák meg.

Ezt követően az első két szabály definiálásával egy kiinduló szabálybázist hozunk létre. A két szabály célja a minimális és a maximális kimeneti értékeket jellemző relációk leírása. Ennek érdekében először megkeressük a két szélsőértéket, majd keresünk egy-egy rájuk illeszkedő adatsort. Amennyiben több adatsor is ugyanazt a kimeneti minimumot/maximumot tartalmazza, akkor azt választjuk, amelyik közelebb van az antecedens tér széleihez.

Tegyük fel, hogy a (1) függvény által leírt görbét (1. ábra) modellező fuzzy rendszer előállítása a célunk. Ekkor két maximumunk van, éspedig a $\pi/2$ és az $5\pi/2$ értékeknél. Az elsőt ($\pi/2$) választjuk ki, mert az van közelebb az *x* alsó korlátjához.

$$y = \sin x, \quad x \in [0, 10] \tag{1}$$



1. ábra. Modellezett függvénykapcsolat

Ezt követően fuzzy halmazokat rendelünk az adatokhoz úgy, hogy ezek referencia pontjai essenek egybe a két adatsor értékeivel. Az alapértelmezett mag és tartó szélességeket alkalmazzuk mindegyik dimenzióban. Annak érdekében, hogy elkerüljük a fuzzy halmazok számának túlzott növekedését a hasonló (egymáshoz közel álló) nyelvi értékeket egyesítjük az antecedens dimenziókban. Az egyesítéseket az alábbi két metaszabály alapján határozzuk el.

1. *Metaszabály*. Amennyiben két nyelvi érték referencia pontja egy megadott $d_{i,min}$ határértéknél közelebb kerül egymáshoz, akkor ezeket egyesítjük. A $d_{i,min}$ határértéket a partíció terjedelmének arányában adjuk meg

$$d_{i,\min} = C_d \cdot DR_i \tag{2}$$

ahol C_d dimenziótól független együttható. A kísérletek során általában $C_d = 0{,}01$ értékkel dolgoztunk.

2. *Metaszabály*. Amennyiben két nyelvi érték paramétereinek átlagos eltérése egy megadott $dp_{i,min}$ határértéknél kisebb, akkor ezeket egyesítjük. A $dp_{i,min}$ határértéket a partíció terjedelmének arányában adjuk meg

$$dp_{i,\min} = C_{dp} \cdot DR_i\,, \tag{3}$$

ahol C_{dp} dimenziótól független együttható. A kísérletek során általában $C_{dp} = 0{,}005$ értékkel dolgoztunk.

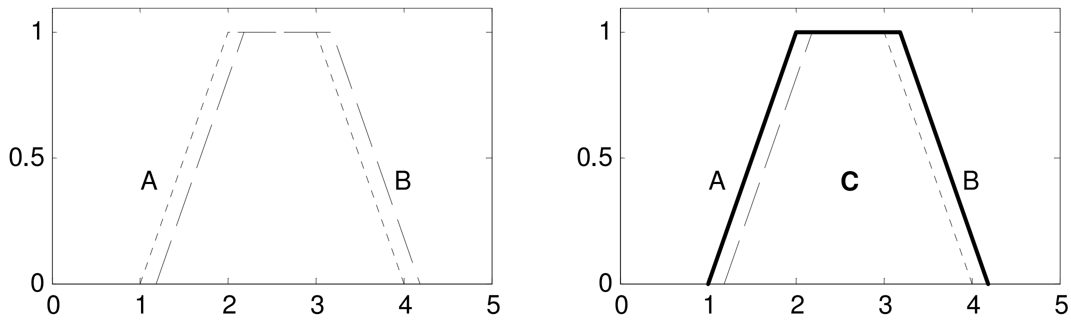
A fuzzy halmazok egyesítését a Kóczy által javasolt CNF[1] unió (ld. 2. ábra) műveletével valósítjuk meg, ami a két halmaz konvex burkaként határozza meg uniójukat. A jelen esetben, ahol minden nyelvi érték CNF és trapéz alakú az unióval kapott trapézt két α-vágatával (*0* és *1* szinten) meghatározhatjuk a következőképpen

$$\inf\{[C]_\alpha\} = \min\{\inf\{[A]_\alpha\}, \inf\{[B]_\alpha\}\},\ \alpha = 0, 1\ , \tag{4}$$

$$\sup\{[C]_\alpha\} = \max\{\sup\{[A]_\alpha\}, \sup\{[B]_\alpha\}\},\ \alpha = 0, 1\ , \tag{5}$$

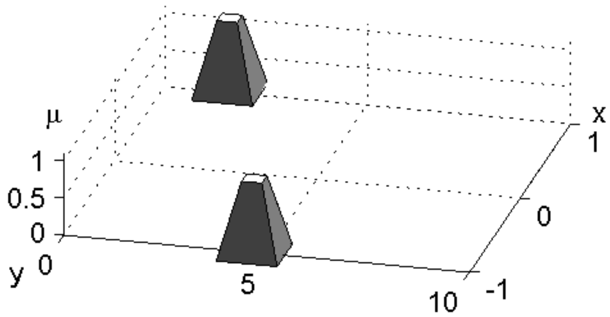
ahol *A* és *B* az eredeti két nyelvi érték, míg *C* az unióval előállított fuzzy halmaz.

[1] CNF – Convex Normal Fuzzy



2. ábra. Két egymáshoz közel álló fuzzy halmaz (bal oldal) és CNF uniójuk (jobb oldal)

A kezdő nyelvi értékekre alapozva létrehozzuk az első két szabályt, amit a 3. ábra illusztrál. Mindkét szabálynak egy-egy csonka gúla felel meg, amit a szabályban szereplő nyelvi értékek definiálnak.



3. ábra. Az első két szabályt tartalmazó szabálybázis

PARAMÉTER AZONOSÍTÁS ÉS SZABÁLYBÁZIS KITERJESZTÉS

A paraméterek optimális értékeinek meghatározását egy heurisztikus algoritmus segítségével oldjuk meg. A cél az, hogy megtaláljuk azt a paramétersort, amelyik biztosítja az alkalmazott teljesítménymutató lehető legjobb értékét.

Minden iterációs menet során egyenként vesszük sorra az összes partíció minden paraméterét, majd az aktuális paraméterhez kiszámítunk két új értéket úgy, hogy az eredeti értéket növeljük, illetve csökkentjük egy előre megadott lépésközzel. A két új értékre kiszámítjuk a fuzzy rendszer teljesítménymutatóját. Amennyiben az eredeti értékre vonatkozó mutató nem ismert, akkor arra is elvégezzük a számításokat. A három lehetséges paraméter érték közül azt tartjuk meg, amelyik a legjobb teljesítménymutatót eredményezte. Minden iterációs menet végén, amikor végighaladtunk egyszer a rendszer összes hangolni kívánt paraméterén összehasonlítjuk az aktuális teljesítménymutatót (PI_k) az előző iterációs menet végén (PI_{k-1}), vagy az első iteráció esetén kiinduláskor (PI_0) mért teljesítménymutatóval.

Amennyiben PI jobban javult egy felső határértéknél ($\Delta PI \geq \Delta PI_{\max}$), akkor a lépésköz nagyságát meghatározó együttható értékét megduplázzuk. Amennyiben a teljesítménymutató javulása egy előre megadott alsó határérték ($\Delta PI \leq \Delta PI_{\max}$), alá esik, akkor az együttható értékét felezzük. Amennyiben az együttható már korábban elérte alsó korlátját (C_{min}), akkor nem osztjuk 2-vel, hanem egy új szabályt generálunk. Ezután a C együtthatót újból kezdeti értékére állítjuk, majd folytatjuk az iterációt. A paraméterazonosítási folyamat akkor áll le, ha elértük az előre megszabott maximális iterációs menetszámot vagy a teljesítménymutató elért egy előre meghatározott jósági értéket (PI_{tr}).

TELJESÍTMÉNYMUTATÓ

A fuzzy rendszer értékelését egy olyan mintaadathalmaz segítségével végezzük, amelynél ismerjük a bemenetekhez elvárt kimeneti értékeket. Minden bementi adatsorra kiszámítjuk a rendszer által előállított kimenetet, és megvizsgáljuk annak eltérését az elvárt kimenettől. Egy olyan értékelő-számra van szükségünk, ami lehetővé teszi két fuzzy rendszer összevetését, és a számunkra előnyösebb kiválasztását. Több különböző teljesítménymutató alkalmazhatóságát is megvizsgáltuk, és közülük az RMSEP bizonyult a legkönnyebben értelmezhetőnek és alkalmazhatónak. Az átlagos négyzetes eltérés négyzetgyökének relatív értéke (RMSEP – Root Mean Square Error in Percentage) a kimeneti alaphalmaz terjedelméhez viszonyítva, annak százalékában jellemzi a rendszert. Képlete a következő

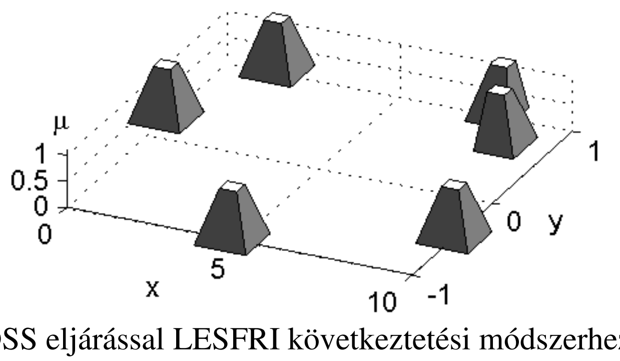
$$PI_{RMSEP} = \sqrt{\frac{\sum_{j=1}^{M}\left(y_j - \hat{y}_j\right)^2}{M}} \cdot DR_{N+1}^{-1} \cdot 100\,, \tag{6}$$

ahol DR_{N+1} a kimeneti dimenzió terjedelmét jelöli, és N az antecedens dimenziók száma, M a mintaadatok száma, y az előírt kimenet $\hat{y}$ a rendszer által számított kimenet.

ÚJ SZABÁLY GENERÁLÁSA

Az RBE megközelítésben az új szabályt úgy hozzuk létre, hogy az illeszkedjen a konzekvens alaphalmaz azon pontjához, ahol a legnagyobb az eltérés a mintaadatok által előírt és a fuzzy rendszer által számított kimenet között. Ennek érdekében az érintett mintaadatsor kikeresése után új nyelvi értékeket hozunk létre úgy, hogy ezek referencia pontjai essenek egybe az adatsor azonos dimenzióbeli értékeivel. Az új fuzzy halmazok alakját a két módszer eltérően határozza meg.

Az ***RBE-DSS*** (Rule Base Extension based on Default Set Shapes - Szabálybázis kiterjesztés alapértelmezett halmazalak használatával) az első két szabály létrehozása során már alkalmazott, partícióra jellemző alapértelmezett halmazalakokat használja az új nyelvi értékek létrehozása során. A partíciónkénti halmazszám túlzott növekedésének elkerülése érdekében két metaszabályt alkalmazunk minden antecedens és konzekvens dimenzióban az esetleges nyelvi érték összevonások szükségességének elbírálására. Ezek azonosak a szabálybázis második szabályának létrehozása során megismert metaszabályokkal, azzal a pontosítással, hogy itt az újonnan létrehozott nyelvi érték és az eredeti fuzzy halmazok viszonylatában vizsgáljuk az összevonhatóságot. A 4. ábra az RBE-DSS segítségével hangolt rendszer szabálybázisának antecedens terét mutatja be 35 iterációs ciklus után (*PI=2,91%*). A rendszergenerálás során a LESFRI [1] következtetési módszert alkalmaztuk. Megfigyelhető, hogy a szabályantecedensek a Kosko [3] által tárgyalt ún. optimális fuzzy szabályoknak megfelelően a görbe forduló- és végpontjainál helyezkednek el.



4. ábra. Az RBE-DSS eljárással LESFRI következtetési módszerhez generált rendszer szabálybázisa

Az RBE-DSS előnye egyszerűségében és gyorsaságában rejlik, gyenge pontja viszont az, hogy a segítségével létrehozott új szabály beillesztése a szabálybázisba a teljesítménymutató ideiglenes romlásához vezethet.

Az ***RBE-SI*** (Rule Base Extension based on Set Interpolation - Szabálybázis kiterjesztés halmaz-interpoláció használatával) az új szabályok létrehozásakor az előző módszernél tapasztalt ideiglenes teljesítménymutató romlást azáltal kívánja elkerülni, hogy a szabályban megjelenő új nyelvi értékeket halmaz-interpolációval állítja elő. A további számítások megkönnyítése érdekében célszerű az alkalmazott halmaz-interpolációs eljárást a szabály-interpolációs módszerhez igazítani. Az RBE-DSS-hez hasonlóan a partíciónkénti halmazszám túlzott növekedésének elkerülése érdekében itt is alkalmazzuk a két metaszabályt a nyelvi érték összevonások szükségességének elbírálására.

A halmaz-interpoláción alapuló RBE-SI előnye, hogy alkalmazásának eredményeképpen az új szabály bevezetése várhatóan legfeljebb kis mértékben rontja a teljesítménymutató pillanatnyi értékét, hátránya viszont az, hogy megnöveli a végső rendszer generálásához szükséges számítások mennyiségét és ezáltal a folyamat időigényét.

ÖSSZEFOGLALÁS

Több bemenettel rendelkező fuzzy rendszereknél az antecedens tér teljes lefedése, ami a kompozíciós elven alapuló klasszikus fuzzy következtetési módszerek alkalmazásának előfeltétele, csak igen nagy szabályszám mellett oldható meg, ami egyben komoly erőforrásigényt is jelent. Ilyen körülmények között különösen előnyös lehet olyan fuzzy rendszerek előállítása, amelyek ritka szabálybázissal rendelkeznek és szabály-interpolációs következtetésre épülnek.

Cikkünkben két ilyen saját fejlesztésű eljárást mutattunk be, amelyek egy iteratív algoritmus keretében szakaszosan bővítve a szabálybázist állítják elő az alacsony szabályszámú és kis tárigényű fuzzy rendszert.

IRODALOMJEGYZÉK

1. Johanyák, Zs. Cs., Kovács, Sz.: Fuzzy Rule Interpolation by the Least Squares Method, HUCI 2006, November 24-25, 2006 Budapest, pp. 495-506.
2. Johanyák, Zs. Cs., Kovács, Sz.: Sparse Fuzzy System Generation by Rule Base Extension, IEEE INES 2007, June 29 – July 1, 2007, Budapest, pp. 99-104.
3. Kosko, B.: Optimal fuzzy rules cover extrema, International journal of intelligent systems, vol. 10, 249 - 255, 1995.
4. Kóczy, L. T. and Hirota, K.: Size reduction by interpolation in fuzzy rule bases, IEEE Transactions on Systems, Man, and Cybernetics, vol. 27, 14 - 25, 1997.
5. Wong, K. W., Kóczy, L. T., Gedeon, T. D., Chong, A. and Tikk, D. :Improvement of the Cluster Searching Algorithm in Sugeno and Yasukawa's Qualitative Modeling Approach, Lecture Notes in Computer Science, vol. 2206, pp. 536–549, 2001.

SZERZŐK

Johanyák Zsolt Csaba, főiskolai adjunktus, Kecskeméti Főiskola, GAMF Kar, Kalmár Sándor Informatikai Intézet, Informatika Szakcsoport, johanyak.csaba@gamf.kefo.hu, H-6000 Kecskemét, Izsáki út 10.

Kovács Szilveszter, tudományos főmunkatárs, Kecskeméti Főiskola, GAMF Kar, Kalmár Sándor Informatikai Intézet, Informatika Szakcsoport, kovacs.szilveszter@gamf.kefo.hu

A szerzők köszönetüket fejezik ki a KF GAMF Karának a kutatáshoz nyújtott támogatásért.