

BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM
GÉPÉSZMÉRNÖKI KAR
ENERGETIKAI GÉPEK ÉS RENDSZEREK TANSZÉK

TDK Dolgozat
Felütköző sugár hőátadási tényezőjének meghatározása neurális
hálózattal

2012

Szerző: Ács Gergely

Gépészeti modellezés (MSc), 2. évfolyam

Konzulens: Dr. Czél Balázs

adjunktus, Energetikai Gépek és Rendszerek Tanszék

A dolgozat bemutatását a TÁMOP - 4.2.2.B-10/1--2010-0009 "Új tehetséggondozó programok és kutatások a Műegyetem tudományos műhelyeiben" című program támogatta.

Tartalomjegyzék

1. Bevezetés	2
2. Elméleti áttekintés.....	3
2.1. Inverz hővezetési feladatok megoldása lágy számítási eljárásokkal.....	3
2.2. A neurális hálózatokról röviden	4
2.3. A hőátadási tényező meghatározása.....	5
2.4. Felütköző sugár vizsgálata inverz módszerrel	5
3. A probléma vizsgálatának koncepciója	7
3.1. A tervezett mérés	7
3.2. A feladat megoldás menete	8
4. Az egyenes feladat megoldása.....	9
4.1. Az egyenes feladat megoldásának validálása	11
5. Az inverz feladat megoldása.....	13
5.1. Érzékenységi vizsgálat	13
5.2. Az alkalmazott neurális hálózat	17
6. Eredmények	20
6.1. Táblázatos függvény-megadás lineáris interpolációval.....	21
6.2. Táblázatos függvény-megadás spline interpolációval	24
6.3. Gauss-görbével való megadás.....	27
6.4. Polinomiális függvénymegadás.....	30
7. További kutatási lehetőségek.....	32
8. Összefoglalás	33
9. Irodalomjegyzék.....	34

1. Bevezetés

Mind ipari, mind hétköznapi esetekben gyakran találkozhatunk felütköző sugarak alkalmazásával. Talán a leghétköznapiabb példa a minden otthonban megtalálható hajszárító, de a vízcsap alá tartott tárgyak is egy-egy felütköző sugárnak vannak kitéve. Ha ezeket a példákat „felnagyítjuk”, hamar eljutunk az ipari alkalmazásokig, melyek mérnöki szempontból érdekesebbek és fontosabbak. A hajszárítóból így lesz hőlégfúvó, a csap alá tartott tárgyból pedig hőkezelésen áteső munkadarab, melyet vízugarakkal hűtenek.

A felütköző sugár nagy előnye, hogy a folyamatosan cserélődő folyadék vagy gáz konstans hőmérsékletet biztosít és az áramlástan hatásai miatt igen magas hőátadási tényezővel rendelkezik. Ám míg a hőmérsékletét könnyen tudjuk mérni, ezáltal szabályozni is, a hőátadási tényező meghatározása problémás lehet. E dolgozat témája egy olyan eljárás, mely egy viszonylag egyszerű mérésből a lehető legtöbb információt szolgáltatva meghatározza meg a keresett mennyiséget. A módszer egy inverz hővezetési feladat megoldásán alapul.

Inverz hővezetési feladatnak nevezzük azokat a problémákat, melyek során ismert hőmérséklet-eloszlás alapján következtetünk a hővezetést leíró egyenlet egy vagy több ismeretlen paraméterére. Ez lehet anyagi jellemző, peremfeltétel (mint jelen esetben) vagy kezdeti feltétel, hogy a legáltalánosabb eseteket említsem, de előfordulhatnak jóval speciálisabb paraméterek is. Inverz megközelítés során a mérés relatíve egyszerű, ám igen komoly matematikai problémát jelent a kiértékelés. Az esetek döntő többségében valamilyen iteratív eljárást alkalmaznak. E dolgozat a mesterséges neurális hálózatokat alkalmazza erre a célja, ami nem más, mint egy algoritmus, mely a biológia idegrendszerek tulajdonságait próbálja utánózni, ez által rendelkezik a tanulás képességével. A neurális hálózatok alkalmazásának nagy előnye, hogy a sikeres tanítás után az inverz feladat megoldása nem igényel iterációt.

A dolgozat egy síkfelületre felütköző körszimmetrikus sugarat vizsgál. A paraméterek úgy lettek megválasztva, hogy a későbbiekben az eredményeket mérésrel is igazolni lehessen. A sugár a korong felső lapját éri, hőátadási tényezője időben állandó, sugárirányban viszont változó, hőmérséklete hely és idő szerint konstans. A test fennmaradó felületei a környező levegővel érintkeznek. A hőmérséklet-eloszlást véges differencia algoritmus határozza meg, mely alapján történik a háló tanítása és ellenőrzése is. A számítások saját kóddal, Matlab környezetben végeztem.

A kutatómunka célja, hogy a felütköző sugár hely szerint változó hőátadási tényezőjét meghatározzuk a sugárral melegített korong hátulján mért hőmérsékletek alapján. Az így adódó inverz hővezetési problémát neurális hálózattal oldottam meg. Megvizsgáltam továbbá különböző függvény-reprezentációk

hatékonyságát, illetve összehasonlítottam a zajos és zajmentes szimulált mérésen alapuló eredményeket.

2. Elméleti áttekintés

2.1. Inverz hővezetési feladatok megoldása lágy számítási eljárásokkal

A bevezetésben már tettem említést az inverz hővezetési feladatokról és azok megoldásának nehézségeiről. A megoldási módszereket két nagy csoportba oszthatjuk, a hagyományos módszerek és a lágy számítási módszerek csoportjába. A hagyományos módszerek döntően valamilyen gradiens alapú iteratív eljárásan alapulnak [1], míg a lágy számítási eljárások (*soft computing methods*) valamely mesterséges intelligencia algoritmust alkalmaznak a probléma megoldására. A leggyakoribbak ezek közül jelenleg a neurális hálózatok [2-4], melyeket már én is sikeresen alkalmaztam korábban [5] és a genetikusan algoritmusok [6-8]. A lágy számítási módszerek alkalmazhatósága már bizonyított, a mai napig is folynak kutatások e téren.

Alkalmazhatóság szempontjából a legnagyobb különbség a neurális hálózatok és a genetikusan algoritmusok között, hogy előbbinek szüksége van egy tanítási ciklusra, ami a feladat bonyolultságától függően néhány perctől kezdve akár órákig is eltarthat, ám ezek után a számítás minden esetre a másodperc tört része alatt megy végbe, míg utóbbinak nincs szüksége ilyen jellegű előkészítésre, de minden egyes számítás iterációs futtatást igényel, melynek időtartama a néhány perces, órás tartományba esik. Tehát a neurális hálózatokat egy-egy célfeladatra érdemes alkalmazni, akár folyamatos online vezérlésre is, ezzel szemben a genetikusan algoritmusok általánosabb problémákra nyújtanak megoldást, lassabb reakcióidővel. Ha párhuzamot akarunk vonni a gyártórendszerek és a két algoritmus között, a neurális hálók feleltethetőek meg a tömeggyártásnak, a genetikusan algoritmusok pedig az egyedi vagy sorozatgyártásnak.

Mindkét eljárás kimenete valamilyen digitális formában tárolt numerikus érték, illetve értékek, éppen ezért analitikus, azaz tökéletesen pontos eredményt nem adnak. Ez a tény meghatározta a korábbi kutatások stratégiáját is, a neurális hálót alkalmazó publikációkban szinte kivétel nélkül az idő-, hely-, vagy hőmérséklet-függő paramétereket táblázatos formában keresték, azaz az ismeretlen függvényt diszkrét pontokban határozták meg, a pontok között lineáris interpolációt alkalmazva. Ez pedig a megoldásba további numerikus hibákat visz. A pontosság növelhető, ha egyre több diszkrét ponttal írjuk le a keresett függvényt, ami viszont nagyobb számítási időt eredményez. Dolgozatomban éppen ezért próbálkozom a hagyományostól eltérő függvény-megadási módszerekkel. Ez a neurális háló számára jóval bonyolultabb transzformációt fog eredményezni, éppen ezért magasabb tanulási időre lehet számítani, és a siker sem garantált.

2.2. A neurális hálózatokról röviden

A természetes neurális hálózatok neuronok, azaz idegsejtek összekapcsolódásából állnak. Ezek a közel azonos elemek nagy mennyiségben olyan hálózatot alkotnak, melyek alkalmasak bonyolult műveletek végrehajtására. A mesterséges neurális hálózatok pontosan ezt a viselkedést próbálják meg matematikailag leírni, hogy tanulásra, és ezután problémamegoldásra alkalmasak legyenek. A modellel elsőként 1943-ban McCulloch és Pitts állt elő [9], ám a neurális hálózatok elterjedésére még várni kellett a 80-as évekig [10]. Az első modellek óta már számos hálóstruktúrát és tanítási algoritmust dolgoztak ki, magyar nyelven talán az egyik legmélyebb könyv a Neurális hálózatok, melyet Varga Gábor szerkesztett, számos író társával együtt [11].

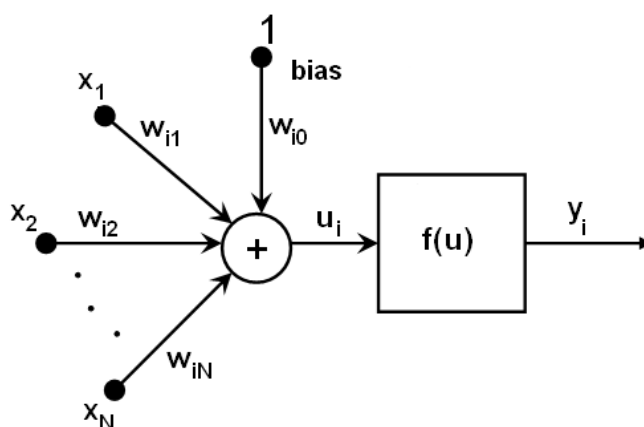
Bár tanítási algoritmusból számtalan létezik, mégis három fő-típusra oszthatjuk őket, ellenőrzött tanításra, nem ellenőrzött tanításra és analitikus tanításra.

Ellenőrzött tanítás során rendelkezésünkre áll az adott bemenethez tartozó elvárt kimenet véges számú esetre, melyek alapján a hálózat az ismeretlen esetek megoldását próbálja megtanulni. A dolgozat során ilyen jellegű tanulásra van szükség.

Ezzel szemben a nem ellenőrzött tanítás során a bemenetekhez tartozó kimenet ismeretlen. Az ilyen tanítás osztályozásra és adattömörítésre alkalmas.

Az analitikus tanítás elve hasonló az ellenőrzött tanításhoz, de a tudást nem iterációkon keresztül fogadja be, hanem a rendelkezésekre álló adatok alapján matematikai összefüggéseket keres.

A hálózatokat feloszthatjuk struktúrájuk alapján is; lehetnek előrecsatoltak, visszacsatoltak, rétegekbe szervezettek, és ezek kombinációi is, így könnyen belátható, hogy számtalan eset állhat elő, ami viszont szinte változatlan az a neuronok felépítése (1. ábra).



1. ábra - Az elemi neuron felépítése

A hálózatok minden eleme kivétel nélkül egy-egy neuron, melyek a teljes hálózathoz képest egy-egy igen egyszerű műveletet hajtanak végre. Véges számú

bemenettel rendelkeznek, melyek közül az egyik értéke konstans 1 (ezt nevezzük *bias*-nak, azaz torzításnak), a többi bemeneti értéket (az ábrán $x_1...x_N$) vagy a háló bemenete adja meg, vagy egy korábbi rétegben elhelyezkedő neuron kimenete. Ezeket az értékeket a neuron felszorozza a hozzájuk tartozó súly-tényezővel (az ábrán $w_{i0}...w_{iN}$), majd az így kapott értékeket összeadja. Az eredményt (u) egy aktivációs függvénnyel ($f(u)$) transzformálja át. Az aktivációs függvény általában a tangens-szigmoid, egy nem-lineáris, de egyértelmű 0 és 1 közötti kimenettel rendelkező függvény. A kezdeti hálók ezzel a nem-linearitással nem rendelkeztek és emiatt azok nem voltak képesek nem-lineáris függvények megtanulására. Többek között ezért is terjedtek el olyan lassan a neurális hálózatok [11].

2.3. A hőátadási tényező meghatározása

A hőátadási tényező egy igen nehezen kézben tartható mennyiség, számos változótól függ, a lokális geometriai méretektől, a környező áramlás Reynolds-számától, tehát az áramlási turbulenciáktól is, és akkor még nem is beszéltünk a halmazállapot-változásokról és a több-fázisú rendszerekről.

A mérnöki gyakorlatban három elterjedt módszer áll rendelkezésünkre, a fél-empirikus képletek, a CFD-szimulációk és az inverz módszeren alapuló mérések.

A fél-empirikus képletek állnak a legközelebb az analitikusan számolt értékekhez, de még így is jelentős hibákkal rendelkeznek számos esetben, illetve egy hely, vagy idő szerint változó hőátadási tényezőt már nem lehet velük kezelni, nem beszélve a komplex geometriákról.

A CFD-szimulációkat a korszerű számítás-technika tette lehetővé. Közel tetszőlegesen bonyolult geometriára változó áramlási peremfeltételekkel meg tudják határozni a hőátadási tényezőt a felület diszkrét pontjaiban, ahol a hálózás lehetővé teszi azt. Hátrányuk viszont az, hogy megfelelő szakértelemre van szükség ezeknek az amúgy is költséges programoknak a helyes használatához, és a probléma komplexitásától függően a számítási idő hatalmas lehet.

A harmadik megoldás az inverz megközelítés, mely a hőmérséklet-eloszlásból következtet a hőátadási tényező értékére. Hatalmas előnye, hogy az áramlástani problémákat teljességgel mellőzi, viszont, mint azt korábban már kifejtettem, a mérés kiértékelése még így is komplex feladat. Mindenképp szükség van hozzá mérésre, vagy hővezetési szimulációra, de ezek a szimulációk jóval kisebb számítási-igénnyel rendelkeznek, mint egy hasonló nagyságrendű CFD-szimuláció, ugyanakkor szintén képesek kezelni a nem-konstans értékeket is.

2.4. Felütköző sugár vizsgálata inverz módszerrel

Nem titkolt tény, hogy e dolgozat Arkadiusz Ryfa, lengyel kutató munkásságából merítette alapötletét. Ryfa 2009-ben publikálta első cikkét, melyben egy korongot melegítettek felütköző sugárral, fűtőközegnek olajt használtak. Ekkor hely szerint lineárisan vagy lépcsőzetesen változó hőátadási tényezőt keresett. Mérésében

szenzorként termo-festéket alkalmazott, és a kritikus hőmérséklet elérésének sebességéből következtetett a hőátadási tényező nagyságára. Az inverz feladat megoldására módosított Levenberg-Marquardt algoritmust használt [12].

Az első 2011-ben megjelent publikációjában már hőkamerával mérte a hátoldali hőmérséklet-eloszlást, és fűtőközegnek nagynyomású forró levegőt használt olaj helyett, valamint a hőátadási tényezőt lineáris függvények kombinációjaként kereste [13].

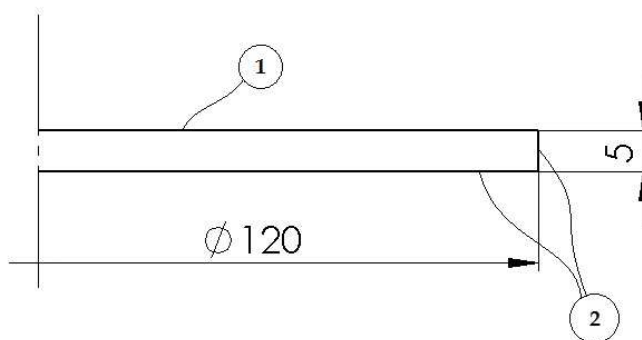
A második 2011-es cikkében a korábban alkalmazott módszerek hatékonyságát és megbízhatóságát igazolta különböző fúvóka átmérőkkel és a fúvóka testtől vett távolságának változtatásával [14].

2012-es cikkében inverz módszerének eredményeit CFD-szimulációval vetette össze, melyben megállapította, hogy módszere bár nem képes tökéletesen visszaadni a szimuláció eredményeit, de a számítási idő nagyságrendekkel rövidebb. [15]

Összességében tehát Ryfa munkája rengeteg részletre kiterjed, ám az általa alkalmazott hagyományos módszeren alapuló megoldási algoritmus nem bizonyul kellően hatékonynak. A legfontosabb eltérés az ő munkája, és e dolgozat tárgya között az inverz feladat megoldásának módszere, illetve, hogy e dolgozatban nem szerepelnek valós mérési eredmények, egy véges differencia algoritmus gondoskodik azok szimulálásáról.

3. A probléma vizsgálatának koncepciója

A 2. ábrán látható a vizsgált geometria, egy 120 mm átmérőjű, 5 mm vastag korong, melynek anyaga rozsdamentes acél. A korong felső lapját éri a felütköző sugár (1-es számmal jelölt felület), a 2-es számmal jelölt felületek a környező, nyugvó levegővel vannak kapcsolatban.

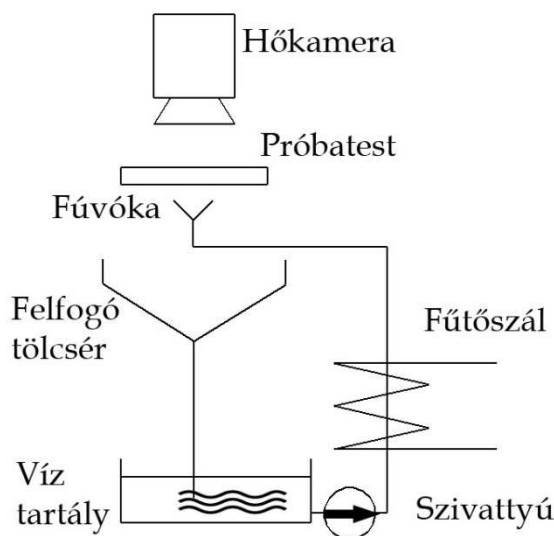


2. ábra - A vizsgált modell

Ahogy a bevezetésben már említettem, a paraméterezés úgy lett megválasztva, hogy az eredményeket egy későbbi méréssel ellenőrizni lehessen. Így a felütköző sugár hőmérséklete konstans 90°C , hőátadási tényezője pedig 1000 és $10000 \text{ W}/(\text{m}^2\cdot\text{K})$ között változhat a sugár mentén. (Ez a valós mérésben egyszerű vízsugárral megvalósítható.) A korong kezdeti hőmérséklete megegyezik a levegő hőmérsékletével. A környező levegő 20°C -os, a hőátadási tényező a nyugvó levegő és a korong között $5 \text{ W}/(\text{m}^2\cdot\text{K})$ értékű. A rozsdamentes acél sűrűsége $7810 \text{ kg}/\text{m}^3$, fajhője $480 \text{ J}/(\text{kg}\cdot\text{K})$, hővezetési tényezője pedig $16,3 \text{ W}/(\text{m}\cdot\text{K})$.

3.1. A tervezett mérés

A TDK keretében nem volt lehetőség valós mérést végezni, de Ryfa cikkei alapján a 3. ábrán látható koncepció már kidolgozásra került. Az elgondolás szerint a próbatestet alulról érné egy fúvókán keresztül a vízsugár, így a víz egy felfogó tölcsérben landol, mely visszavezeti azt a tartályba. A megfelelő nyomást egy szivattyú szolgáltatná, az állandó közeghőmérsékletet pedig egy fűtőszál biztosítaná. A hőmérsékletmérést a próbatest hátlapját figyelő hőkamera végezné.



3. ábra – A tervezett mérés sematikus ábrája

3.2. A feladat megoldás menete

Minden inverz feladat megoldásához szükség van az egyenes feladat megoldására is, a neurális hálózat alkalmazása sem jelent kivételt. Míg a legtöbb megoldási módszer az iterációs lépéseknél igényli az egyenes feladat megoldását minden egyes számításnál, a neurális hálózatoknak csak a tanítás során van szüksége rá, azután soha többé. A tanítás során a hálózat az általunk kiválasztott ki- és bemenő értékek között fennálló függvénykapcsolatot tanulja meg. Esetünkben a bemenetet a hátlapon mérhető hőmérséklet-eloszlás, a kimenetet pedig a felütköző sugár hőátadási tényezője jelenti. A betanított hálózat akkor fog megfelelően működni, ha a tanítás során megfelelően sok eltérő függvény-párból tud tanulni. Mivel mért adat nem áll rendelkezésre, ezért csak szimulációval kaphatunk használható adatokat. A legkézenfekvőbb megoldás egy véges differencia-módszeren alapuló algoritmust készíteni, mivel így tetszőleges számú tanításra (és tesztelésre) használható függvény-párt lehet generálni.

Tehát összegezzük a megoldás menetét:

- Kijelöljük az inverz feladatot.
- A neurális hálózat tanításához az egyenes feladat megoldásával megfelelő mennyiségű adatot nyerünk.
- A legenerált adathalmazzal betanítjuk a hálót, mely így már alkalmas az inverz feladat megoldására.
- Alkalmazás előtt a háló eredményeit validáljuk és kiértékeljük.

4. Az egyenes feladat megoldása

Az egyenes feladatot leíró modell körszimmetrikus közelítéssel él, mivel a mérési koncepció szerint a sugár keresztmetszete kör alakú és merőlegesen éri a korong középpontját. Így hővezetési szempontból kétdimenziós tranziens feladatról beszélünk.

Kiinduló egyenletem a hővezetés általános differenciálegyenletének hengerkoordinátás alakja volt, belső hőforrás és szög menti változás nélkül (1).

$$\frac{\delta T}{\delta \tau} = a \cdot \left[\frac{\delta^2 T}{\delta r^2} + \frac{1}{r} \cdot \frac{\delta T}{\delta r} + \frac{\delta^2 T}{\delta z^2} \right] \quad (1)$$

ahol a a hőfokvezetési tényező. Megoldásom során explicit módszert alkalmaztam, mely hely szerint centrális (2-3), idő szerint előrelepő közelítést alkalmaz (4).

$$\frac{\delta T}{\delta r} = \frac{T(r + \Delta r, z, \tau) - T(r - \Delta r, z, \tau)}{2 \cdot \Delta r} \quad (2)$$

$$\frac{\delta T}{\delta z} = \frac{T(r, z + \Delta z, \tau) - T(r, z - \Delta z, \tau)}{2 \cdot \Delta z} \quad (3)$$

$$\frac{\delta T}{\delta \tau} = \frac{T(r, z, \tau + \Delta \tau) - T(r, z, \tau)}{\Delta \tau} \quad (4)$$

A továbbiakban bevezetem i, j és n indexet a jobb átláthatóság céljából. $i=1,2,..I$ jelöli a sugárirányú, $j=1,2,..J$ a vastagság menti koordinátát, $n=1,2,..N$ pedig az időbeli lépést. Továbbá $\Delta r = \Delta z$ egyszerűsítéssel élek. Így megkapjuk az (1) egyenletet véges differencia alakban (5).

$$\frac{T_{i,j}^{n+1} - T_{i,j}^n}{\Delta \tau} = a \cdot \left[\frac{T_{i+1,j}^n + T_{i-1,j}^n + T_{i,j+1}^n + T_{i,j-1}^n - 4T_{i,j}^n}{\Delta r^2} + \frac{1}{r_i} \cdot \frac{T_{i+1,j}^n - T_{i-1,j}^n}{2\Delta r} \right] \quad (5)$$

További egyszerűsítéshez érdemes bevezetni p differencia modulust(6).

$$p = \frac{a \cdot \Delta \tau}{\Delta r^2} \quad (6)$$

Az explicit módszer szerint, az (5) egyenletet $T_{i,j}^{n+1}$ -re kell rendezni(7).

$$T_{i,j}^{n+1} = p \cdot \left[T_{i+1,j}^n + T_{i-1,j}^n + T_{i,j+1}^n + T_{i,j-1}^n - 4T_{i,j}^n \right] + \frac{p}{2} \cdot \frac{\Delta r}{r_i} \left[T_{i+1,j}^n - T_{i-1,j}^n \right] + T_{i,j}^n \quad (7)$$

Ugyanakkor fontos megjegyezni, hogy $r=0$ -ban a (7) egyenletnek szinguláris pontja van, így határértékkel határozhatjuk meg e pontokban a hőmérsékletet (8).

$$\lim_{r \rightarrow 0} \left(\frac{1}{r} \frac{\partial T}{\partial r} \right) = \frac{\partial^2 T}{\partial r^2} \quad (8)$$

Ezáltal lokálisan módosul a (7) egyenlet. (9)

$$T_{1,j}^{n+1} = p \cdot \left[2 \cdot T_{2,j}^n + 2 \cdot T_{0,j}^n + T_{1,j+1}^n + T_{1,j-1}^n - 6T_{1,j}^n \right] + T_{1,j}^n \quad (9)$$

Az differencia egyenlet megoldásához szükség van perem- és kezdeti feltételekre is. Ezek folytonos alakját adják meg a (10-14) egyenletek.

$$-\lambda \frac{\partial T}{\partial r} \Big|_{r=0} = 0 \quad (10)$$

$$-\lambda \frac{\partial T}{\partial r} \Big|_{r=R} = \alpha_2 (T(r=R, z, \tau) - T_{\infty 2}) \quad (11)$$

$$-\lambda \frac{\partial T}{\partial z} \Big|_{z=0} = \alpha_2 (T(r, z=0, \tau) - T_{\infty 2}) \quad (12)$$

$$-\lambda \frac{\partial T}{\partial z} \Big|_{z=Z} = \alpha_1(r) (T(r, z=Z, \tau) - T_{\infty 1}) \quad (13)$$

$$T(r, z, \tau=0) = T_{\infty 2} \quad (14)$$

A kezdeti- és peremfeltételi egyenletek véges differencia alakban (a tartományon kívül eső, úgynevezett virtuális pontok hőmérsékletére rendezve) a következők. (15-19)

$$T_{0,j}^n = T_{2,j}^n \quad (15)$$

$$T_{l+1,j}^n = -\frac{2\alpha_2 \Delta r}{\lambda} (T_{l,j}^n - T_{\infty 2}) + T_{l-1,j}^n \quad (16)$$

$$T_{i,0}^n = -\frac{2\alpha_2 \Delta r}{\lambda} (T_{i,1}^n - T_{\infty 2}) + T_{i,2}^n \quad (17)$$

$$T_{i,j+1}^n = -\frac{2\alpha_1(r) \Delta r}{\lambda} (T_{i,j}^n - T_{\infty 1}) + T_{i,j-1}^n \quad (18)$$

$$T_{i,j}^1 = T_{\infty 2} \quad (19)$$

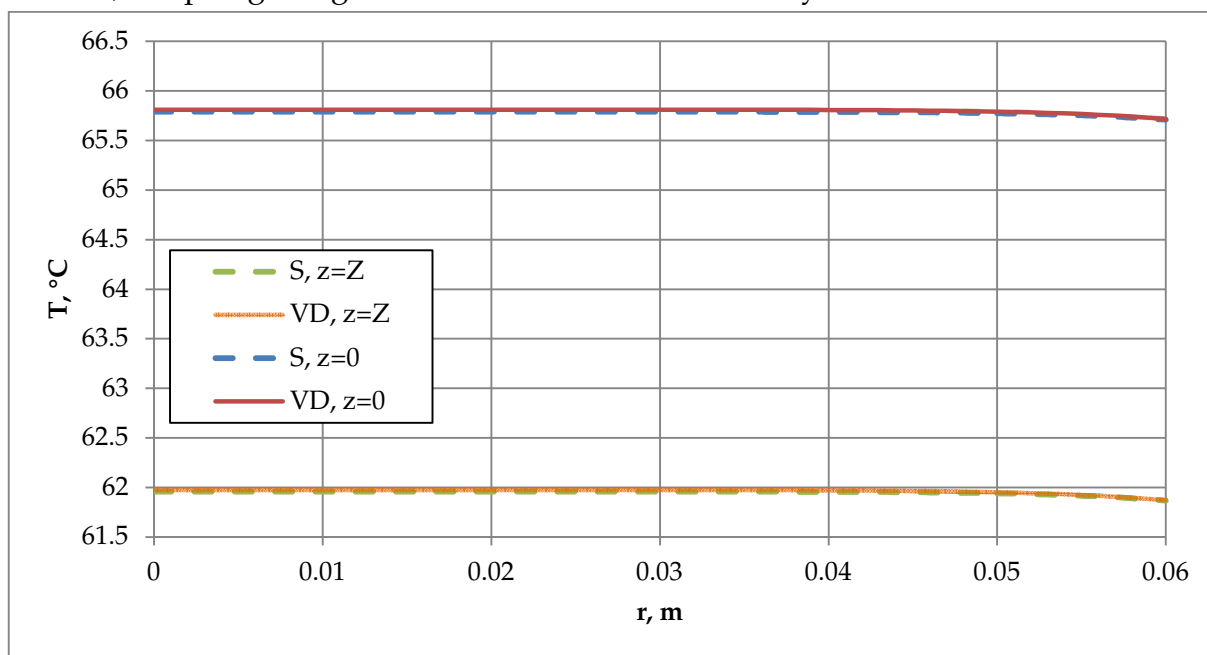
A (7,9,15-19) egyenletek felhasználásával a megadott elrendezés tranziens hőmérséklet-eloszlása meghatározható.

4.1. Az egyenes feladat megoldásának validálása

Ahhoz, hogy megbizonyosodjunk a megoldás pontosságáról, illetve helyességéről célszerű összevetni az eredményeket valamilyen más alapon működő megoldással. Jelen esetben ez a SolidWorks CAD-rendszerbe beépített Simulation végeselemes programcsomaggal történt.

A vizsgált modell egy korong, a 2. ábrán látható méretekkel, és természetesen minden paraméter megegyezik az ismertetett konfigurációéval. Mivel sajnálatos módon a Simulation nem alkalmas hely szerint folytonosan változó hőátadási tényező megadására, így egy konstans esetben hasonlítom össze az eredményeket, amikor a felütköző sugár hőátadási tényezője minimális, tehát $1000 \text{ W}/(\text{m}^2 \cdot \text{K})$. A szimuláció tranziens. A hely szerinti hőmérséklet-eloszlást a 20. másodpercben ábrázoltam az előlapon ($z=Z$), illetve a hátlapon ($z=0$). Az tranziens hőmérséklet görbéket pedig $r=R$ -nél, a korong elő- ($z=Z$) és hátlapjának ($z=0$) peremén hasonlítom össze.

A hely szerinti eloszlás-ábrákon jól látható (4. ábra), hogy a két numerikus módszer jellegre azonos görbét adott ki, a legnagyobb eltérés $0,02 \text{ }^\circ\text{C}$ nagyságrendű. Erről pontos számértékeket az 1. táblázat tartalmaz. S jelöli a Simulation által számított értékeket, VD pedig a véges differencia módszer eredményeit.



4. ábra – Hőmérséklet-eloszlás különböző eljárásokkal, $\tau=20 \text{ s}$

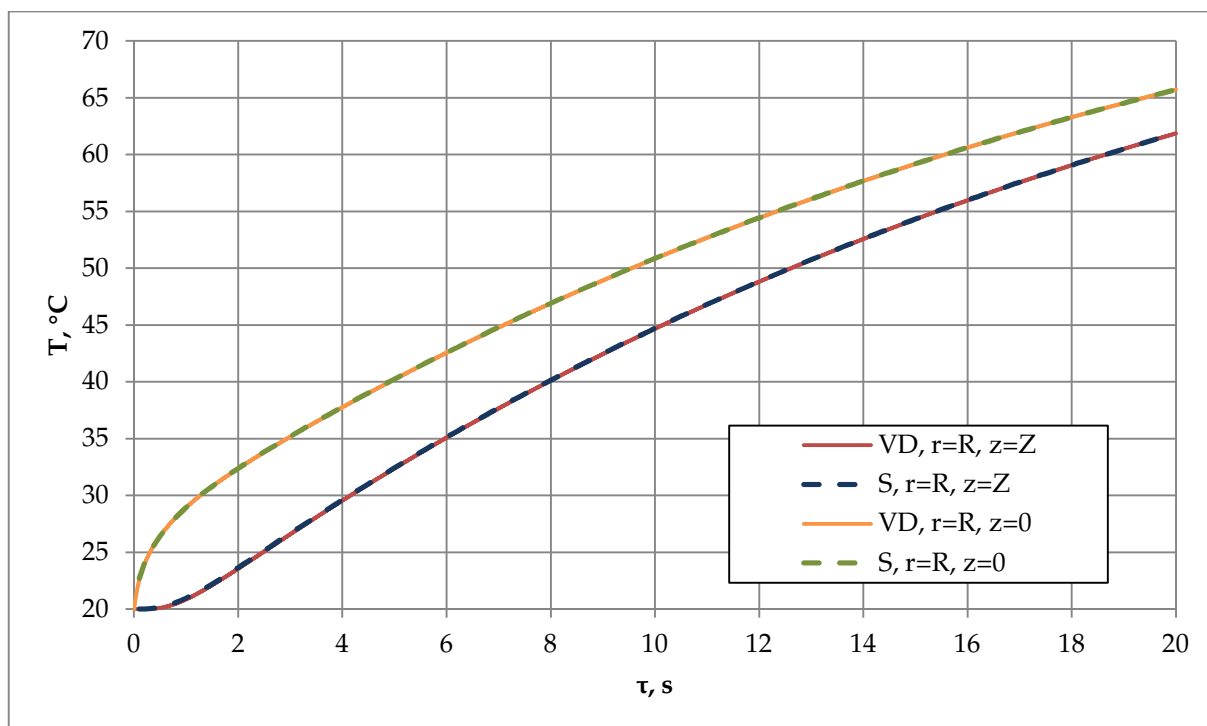
1. táblázat - Hőmérséklet-eloszlás különbségek, $\tau=20$ s

Előlap, $z=Z$				Hátlap, $z=0$		
r, m	$S, ^\circ C$	$VD, ^\circ C$	$\Delta T, ^\circ C$	$S, ^\circ C$	$VD, ^\circ C$	$\Delta T, ^\circ C$
0	65.790	65.812	0.022	61.958	61.975	0.017
0.01	65.790	65.812	0.022	61.958	61.975	0.017
0.02	65.790	65.812	0.022	61.958	61.975	0.017
0.03	65.790	65.811	0.021	61.958	61.974	0.016
0.04	65.788	65.808	0.020	61.955	61.971	0.016
0.05	65.775	65.792	0.017	61.940	61.952	0.012
0.06	65.709	65.717	0.008	61.868	61.870	0.002

A tranziens összehasonlítás eredményét az 5. ábra mutatja, a görbék gyakorlatilag teljesen fedik egymást, a számértékeket a 1. táblázat 2. táblázat tartalmazza. Az eltérés kezdetben megközelíti a $0,05\text{ }^\circ\text{C}$ -os értéket, de a későbbiekben jóval alacsonyabb eltérést tapasztalunk.

2. táblázat - Tranziens hőmérséklet-eloszlás különbségek

Előlap pereme, $r=R, z=Z$				Hátlap pereme, $r=R, z=0$		
τ, s	$S, ^\circ C$	$VD, ^\circ C$	$\Delta T, ^\circ C$	$S, ^\circ C$	$VD, ^\circ C$	$\Delta T, ^\circ C$
5	40.223	40.250	0.027	32.388	32.436	0.048
10	50.851	50.862	0.011	44.678	44.702	0.024
15	59.183	59.183	0.000	54.314	54.323	0.009
20	65.717	65.709	0.008	61.870	61.868	0.002



5. ábra - Tranziens hőmérséklet- eloszlás

Az összevetés alapján az alkalmazott véges differencia algoritmus kellően pontos eredményeket ad, így megfelel az inverz feladat megoldásának további lépéseéhez.

5. Az inverz feladat megoldása

5.1. Érzékenységi vizsgálat

Érzékenységi vizsgálat alatt azt értjük, hogy az általunk keresett függvény egyes paramétereinek megváltozása mennyivel változtatja meg a hőmérséklet-eloszlást, mind idő, mind hely szerint. Erre azért van szükség, mert alacsony érzékenység esetén az egyes eseteknek alacsony információtartalma van a neurális hálózat számára, tehát nehezebben tudja megtanulni azt, vagy éppen egyáltalán nem képes rá az adott paraméterek mellett.

Az érzékenységi tényező (B_k) definíció szerint (20) a hőmérséklet-eloszlás függvény parciális deriváltja minden keresett paraméterre (P_k), ahol $k=1,2,..K$, a keresett függvényt leíró paraméter sorszáma [16].

$$B_k = \frac{\partial T(r, \tau)}{\partial P_k} \quad (20)$$

Mivel nem minden esetben lesz a paraméterek dimenziója azonos, nem minden esetben lehet összehasonlítani az így kapott tényezőket, ezért célszerű bevezetni a relatív érzékenységi tényezőt (21).

$$B_k^* = \frac{\partial T(r, \tau)}{\partial P_k} \cdot P_k \quad (21)$$

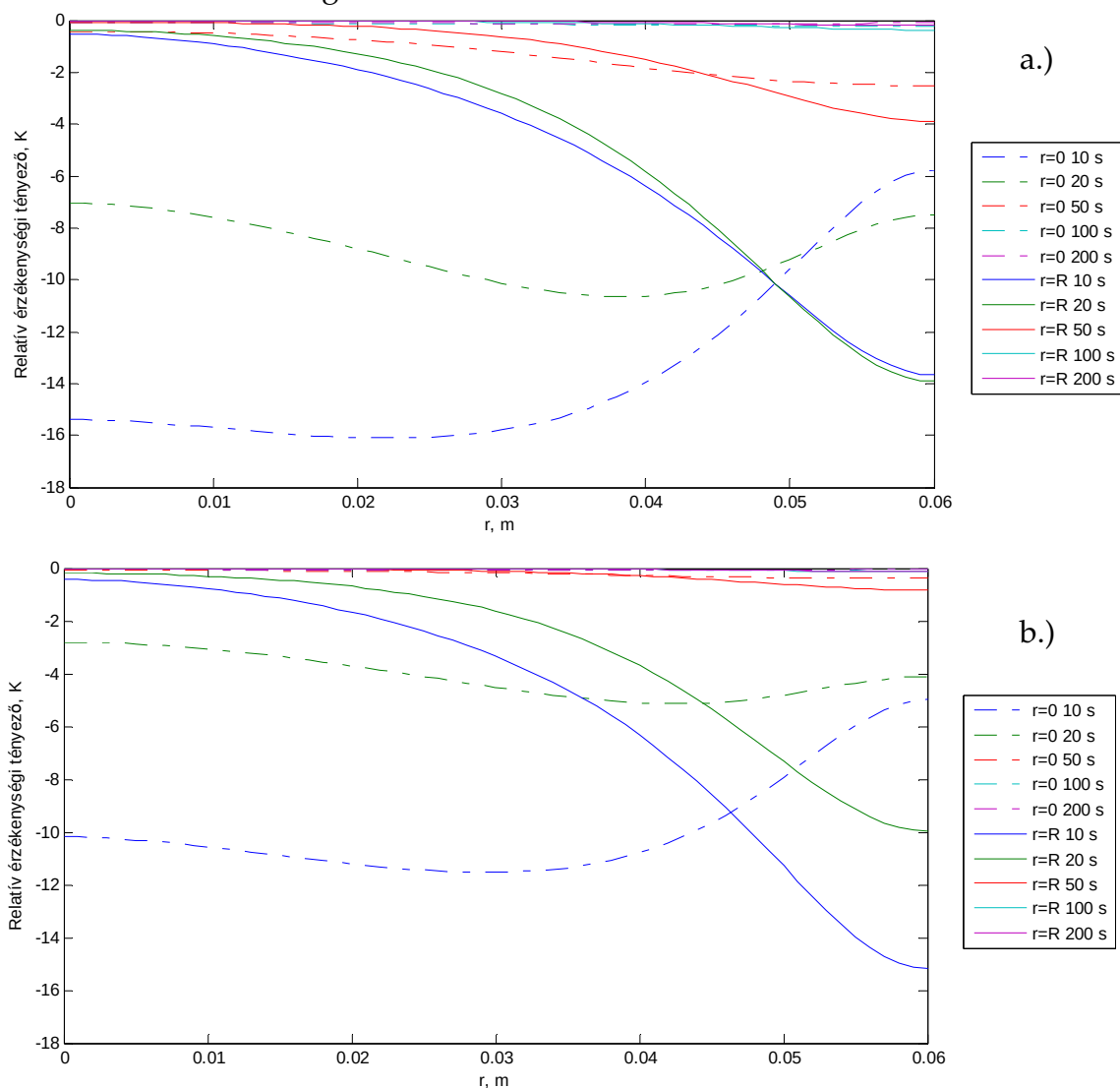
A numerikus megoldás viszont lehetetlenné teszi a differenciálszámítást, így numerikus, előrelépő véges differencia módszert alkalmazok, ahol $\varepsilon=10^{-4}$ (22).

$$B_k^* = \frac{T_{(P_1, P_2, \dots, P_k + \varepsilon P_k, \dots, P_K)}(r, \tau) - T_{(P_1, P_2, \dots, P_k, \dots, P_K)}(r, \tau)}{\varepsilon P_k} \cdot P_k \quad (22)$$

Az érzékenységi vizsgálat elsődleges célja az volt, hogy kijelöljem, mely időpontok legyenek a neurális hálózat bemenetei, úgy, hogy a valós mérés is megvalósítható legyen. A vizsgálat során két esetben néztem az érzékenységet. Mindkét esetben lineárisan csökkent a hőátadási tényező a középponttól. Első esetben 5000-ről 1000 W/(m²·K)-re csökkent, második esetben pedig 8000-ről 2000 W/(m²·K)-re. A vizsgálat során a lineáris függvényt két paraméterrel definiáltam, $r=0$ -nál, illetve

$r=R$ -nél határoztam meg a függvény értékeit, így e két változóra vizsgáltam az érzékenységet.

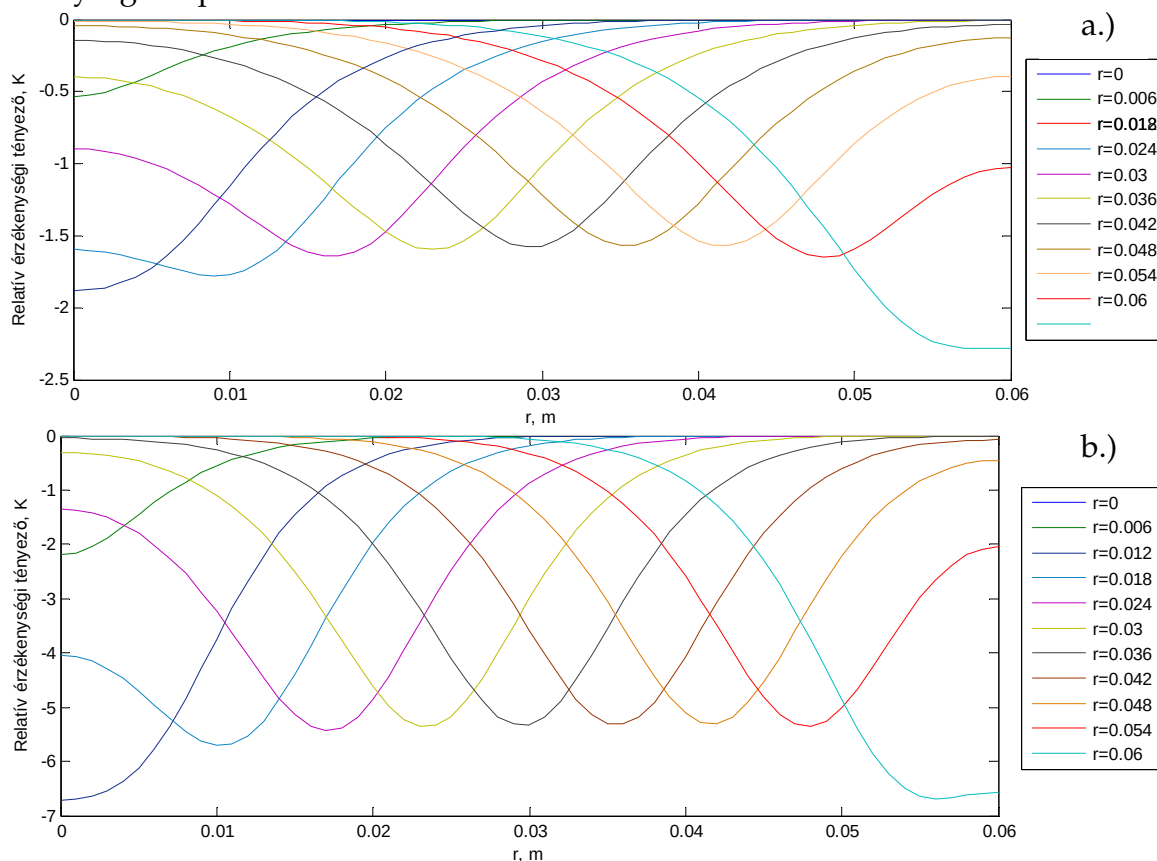
Az eredményeket a 6. ábra mutatja. Mindkét esetben egyértelmű, hogy az érzékenység a korai időpillanatokban jóval nagyobb, mint a későbbiekben, ezért a neurális hálózat bemeneteit 10 és 20 másodpercnél határoztam meg, melyek valós mérésnél is jól időzíthetőek. Továbbá az is leolvasható az ábrákról, hogy a nagyobb hőátadási tényező kisebb érzékenységi tényezőt eredményez. Ez azzal magyarázható, hogy a hőátadási folyamat gyorsabban megy végbe magasabb hőátadási tényező esetén, tehát hamarabb közelít az állandósul állapothoz, ez által kevesebb információt szolgáltatván a tranzienshatásról.



6. ábra - lineáris érzékenységi vizsgálatok, a.) 5000-1000 W/(m²·K)-, b.) 8000-2000 W/(m²·K)-

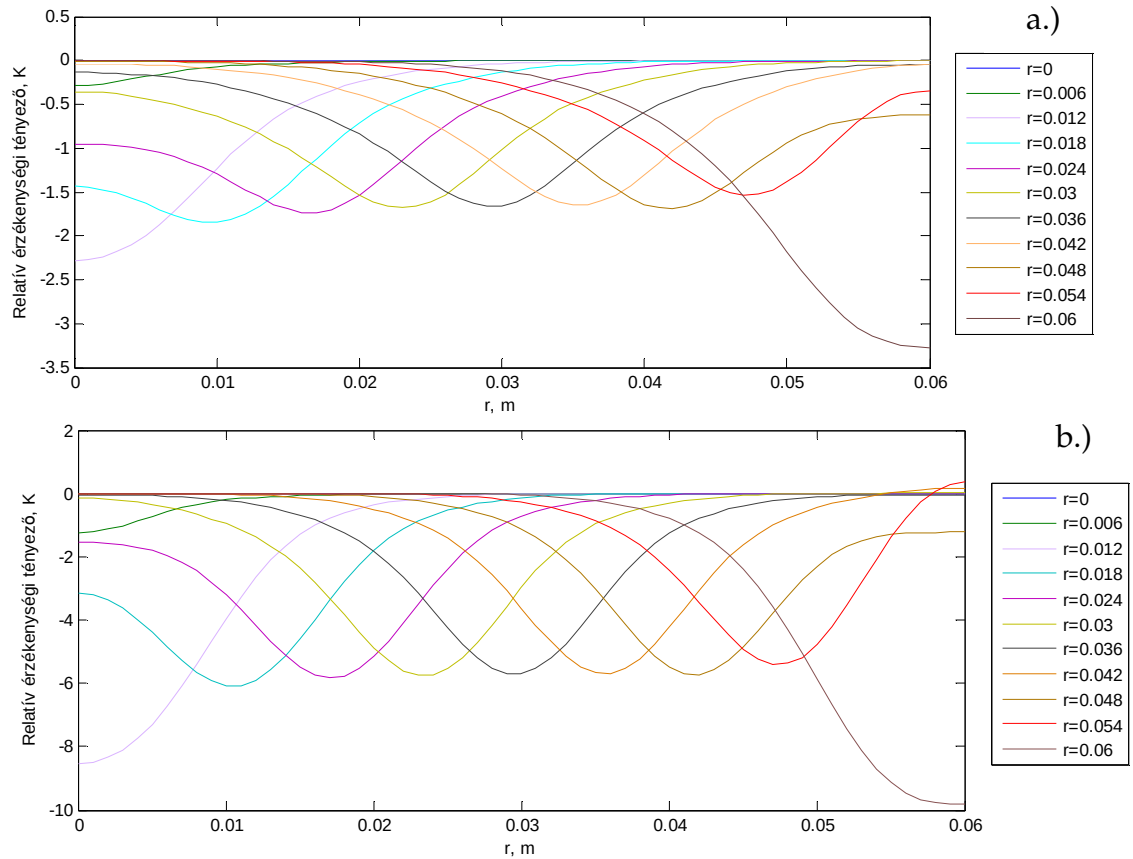
A bemeneti időpontok kiválasztása után érdemes megvizsgálni a keresett függvények érzékenységét is. Mivel táblázatos megadás esetén végtelen sok függvény-alak létezik, így a sugár mentén állandó 5000 W/(m²·K)-es eseten végeztem érzékenységi vizsgálatot.

A lineáris interpolációs megközelítés érzékenységet a 7. mutatja. A görbék az adott ponton megváltoztatott hőátadási tényezők hatását mutatják. Az egyes paraméterek hatása közel megegyezik, ám a széleken valamivel nagyobb az érzékenység. Ezenkívül itt is megfigyelhetjük, hogy a korábbi időpillanatban jóval nagyobb érzékenységet tapasztalunk.



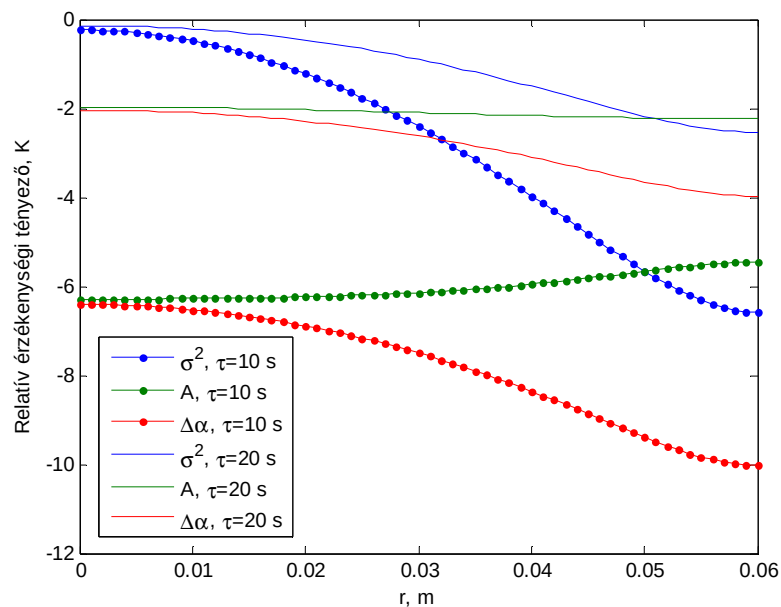
7. ábra - érzékenységi vizsgálat lineáris interpoláció esetén, a.) $\tau=20$ s-, b.) $\tau=10$ s

Hasonló, bár nem azonos eredményeket kapunk a spline interpolációt alkalmazó megközelítésnél is (8. ábra), az értékek nagyságrendje megegyezik a lineáris interpolációnál látottakkal. Így hasonló szintű megoldást várunk az inverz feladat megoldására, mint az előző esetbenél.



8. ábra - érzékenységi vizsgálat spline interpoláció esetén, a.) $\tau=20$ s-, b.) $\tau=10$ s

Gauss-függvény megadással csak három paramétert kell vizsgálnunk, ám eltérő mennyiségekről van szó, így az érzékenységi görbék is más jelleget vesznek fel. A vizsgált paraméterezés: $\sigma^2=0,05$, $A=3000$, $\Delta\alpha=3000$. A korábbiakhoz hasonlóan a korábbi időpillanat itt is nagyobb érzékenységgel rendelkezik (9. ábra). A σ^2 és $\Delta\alpha$ jóval nagyobb érzékenységgű, mint σ^2 , így a szórás becslése várhatóan kevésbé lesz pontos.



9. ábra - érzékenységi vizsgálat Gauss-görbe esetén

A polinomiális megközelítéshez nem készült érzékenységi vizsgálat, mivel ez a függvénymegadási módszer értékelhető eredményt a kutatás alatt nem adott, ezt részletesebben a 6.4 alfejezetben taglalom.

5.2. Az alkalmazott neurális hálózat

Az inverz feladat megoldásához egy előrecsatolt neurális hálózatot (*Feedforward neural network*) alkalmaztam, melyet Levenberg-Marquardt algoritmust alkalmazó hiba-visszaterjesztéses algoritmussal (*Levenberg-Marquardt backpropagation*) tanítottam.

Az előrecsatolt háló felépítését tekintve két jellemzővel bír: a hálózat neuronjai csak a következő réteg számára adnak információt, tehát visszacsatolás egyáltalán nem történik, valamint minden egyes neuron összeköttetésben van a soron következő réteg összes neuronjával. Háromrétegű hálózatot alkalmaztam, az első réteg a bemeneti réteg, mely a bemeneteket kezeli, majd továbbadja a rejtett rétegnek, mely számításait a kimeneti rétegnek adja le, amely végül a hálózat kimeneteit képezi. A rejtett réteg aktivációs függvénye tangens-szigmoid (23), míg a kimeneti rétegé lineáris. Előbbi a hálózat nem-lineáris leképezéseit teszi lehetővé, utóbbi pedig a kimenetek tetszőleges skálázását.

$$f(x) = \frac{2}{1 + e^{-2x}} - 1 \quad (23)$$

A hálózat rejtett rétege 10, 20, illetve 40 neuront tartalmazott a vizsgálatok során, ez a paraméter határozza meg a háló számítási kapacitását, ám több rejtett neuron a tanítási időt is jelentősen megnöveli.

A hiba-visszaterjesztéses algoritmus az előrecsatolt hálózatoknál leggyakrabban alkalmazott ellenőrzött tanítási módszer, Levenberg-Marquardt algoritmussal kombinálva igen gyors és hatékony tanulást eredményez.

A hálózat bemenetei a hátlap diszkrét pontjain mért hőmérséklet-értékek. Mivel a kísérlet során hőkamerát használunk, így feltételezhetjük, hogy bármely pontban ismerjük a hőmérsékletet. Mivel az inverz feladatot a neurális hálózat csak véges számú bemenettel tudja megoldani, ezért bemenetként 11 pontot határoztam meg, melyek egymástól 6-6 mm-re helyezkednek el, ezzel 10 egyenlő szakaszra bontva a hőkamerával mért felület sugarát. Továbbá érdemes két eltérő időpontban mért hőmérséklet-eloszlást alkalmazni, mivel a tranzienshatás információt tartalmaz a háló számára a hőátadási tényező nagyságáról. Ezeknek az időpillanatoknak a kiválasztásával az érzékenységi vizsgálat során foglalkoztam.

A hálózat kimenete értelemszerűen a sugár mentén változó hőátadási tényező, ám a különböző függvényreprezentációk módosítják annak formáját. Négy megközelítést vizsgáltam: a táblázatos megadást lineáris és spline interpolációval, illetve Gauss-görbével és polinomiális együtthatókkal való megadást. Táblázatos megadás esetén a hőátadási tényezőt a mért hőmérsékletekkel azonos sugár

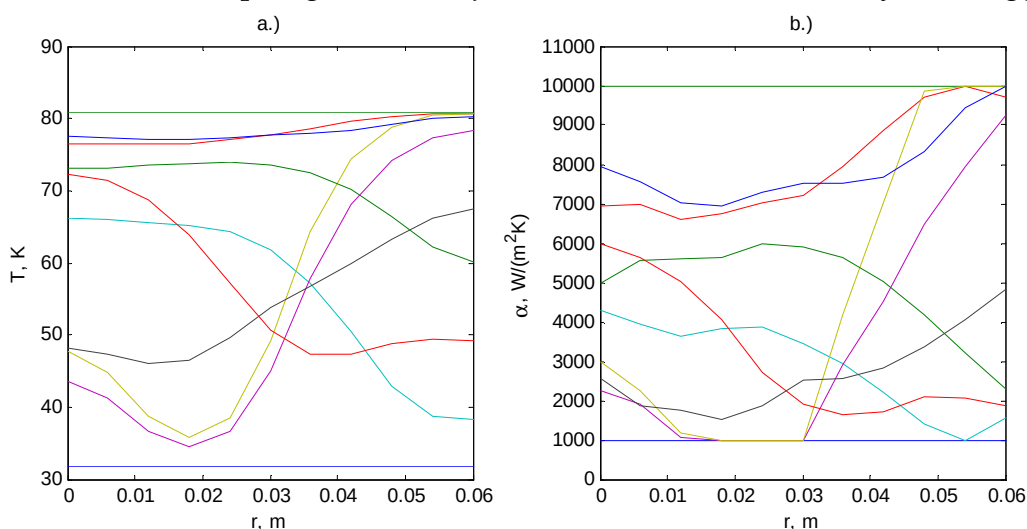
koordinátáknál keresem, egyik esetben a pontok között lineáris, másik esetben spline interpolációval, mindkét esetben 11-11 kimenete lesz a hálónak. Gauss, vagyis haranggörbés megadásnál a (24) egyenlet által megadott formában keressük a hőátadási tényező-függvényt. Így a függvényt 3 paraméterrel le tudjuk írni: A haranggörbe maximális értéke, σ^2 a függvény szórása, és $\Delta\alpha$ a görbe függőleges eltolása. Ez a megközelítés jóval kevesebb paramétert jelent, úgy, hogy a függvényt folytonosan le tudjuk írni, viszont a hálózat számára bonyolultabb a megtanulandó függvénykapcsolat, ezenfelül korlátozza a hőátadási tényező függvény formáját is.

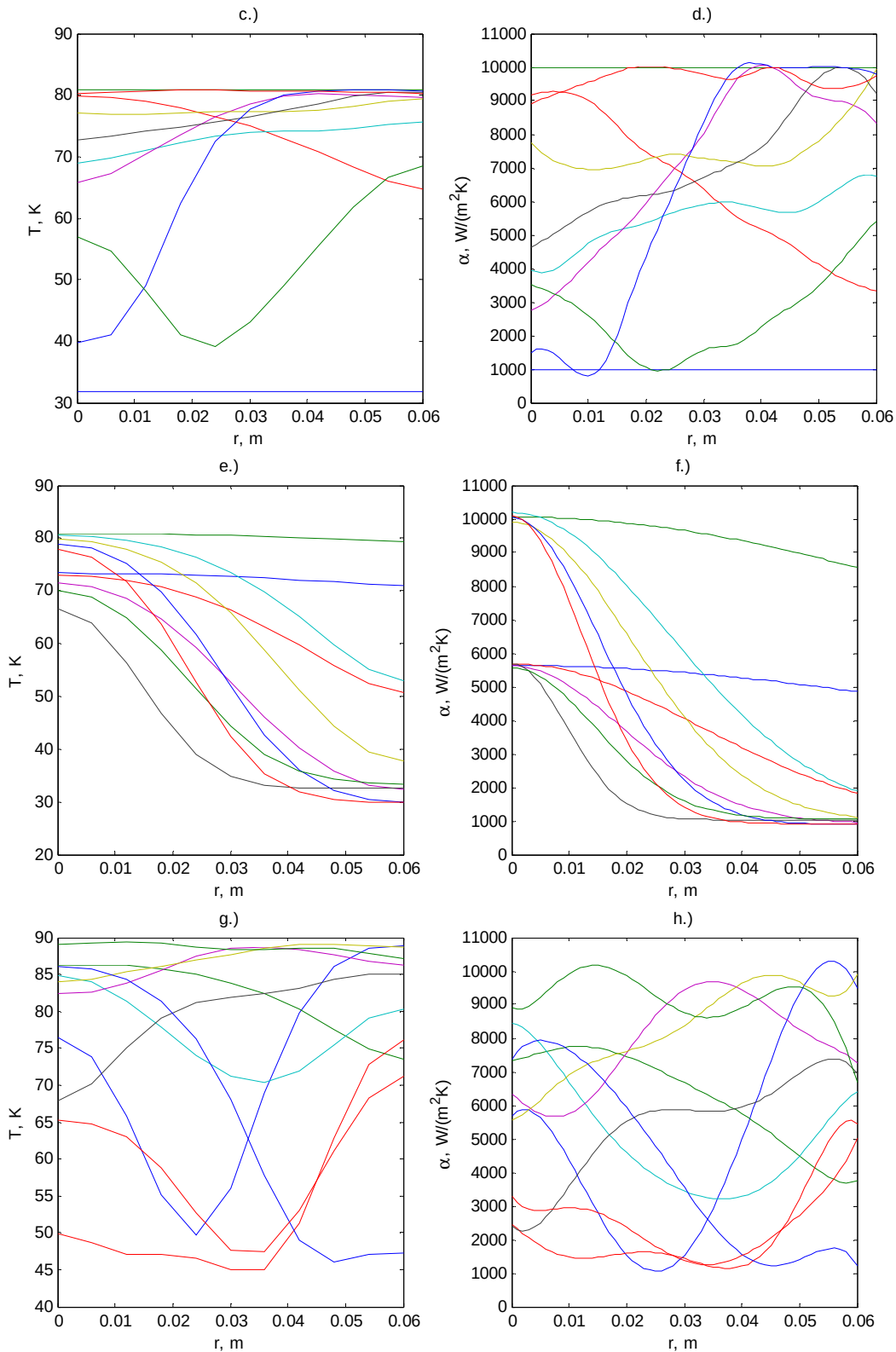
$$\alpha(r) = A \cdot e^{-\frac{r^2}{2\sigma^2}} + \Delta\alpha \quad (24)$$

A polinomiális reprezentáció szerint a hőátadási tényezőt a (25) egyenlet alakjában keressük, melyben az ismeretlenek az $a_0, \dots, a_n$ együtthatók. A polinom foka meghatározza, hogy összesen hány minimuma és maximuma lehet a függvénynek, azaz, milyen bonyolult formákat tud leírni. A hatodfokú polinom már elég változóval rendelkezik, hogy jól közelítse a táblázatos formában 11 pontban leírt függvényeket, így 7 kimenetű lesz az alkalmazott hálózat. Ez a reprezentáció közel tetszőleges alakú függvényt tud véges számú paraméterrel folytonosan leírni, viszont a hálózat számára ez jelenti a legbonyolultabb függvénykapcsolatot, illetve az együtthatók közti eltérés több nagyságrendű.

$$\alpha(r) = a_0 + a_1 r + a_2 r^2 + \dots + a_n r^n \quad (25)$$

A hálózat bemeneti és kimeneti függvényeit szemlélteti a 10. ábra 10-10 példafüggvénnyel, baloldalt (a,c,e,g) a bemeneti hőmérsékletek láthatóak $\tau=10$ s-nál, jobb oldalt (b,d,f,h) pedig a hozzájuk tartozó hőátadási tényező függvény.

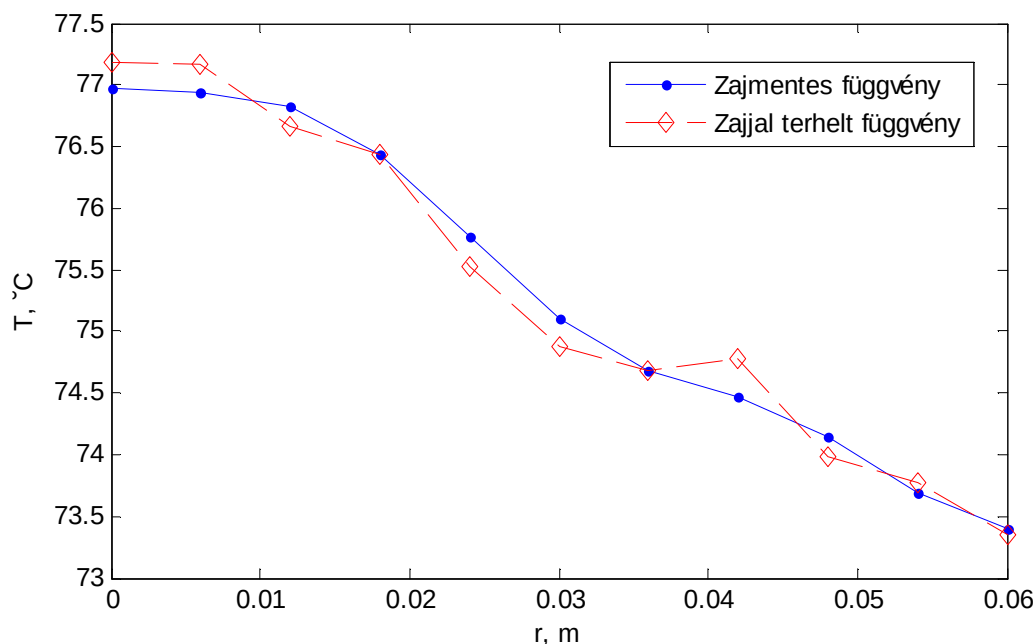




10. ábra- Be- és kimeneti függvény példák a,b.) táblázatos megadás lineáris interpolációval, c,d.) táblázatos megadás spline interpolációval, e,f.) Gauss-görbe, g,h.) polinomiális megadás

6. Eredmények

A módszer megvalósítása szinte biztosan mérési bizonytalanságokkal fog együtt járni. Ezért a neurális hálózatokat zajos és zajmentes bemeneti értékekkel is tanítottam, illetve kiértékelésük során is mindkét esetet figyelembe vettem. A modellezett zaj véletlenszerű, szórása 0,5 °C, eloszlása normális. Egy zajmentes bemeneti függvényt és annak zajjal terhelt változatát mutatja be a 11. ábra.



11. ábra – Zajmentes és zajjal terhelt bemeneti függvény, $\tau=10$ s

A neurális hálózatok kiértékelése az esetek döntő többségében négyzetes középérték-hibával történik (26).

$$RMS = \sqrt{\frac{1}{N} \sum_{o=1}^N (d_o - y_o)^2} \quad (26)$$

d_o az adott, elvárt kimeneti érték, y_o a háló által számított érték, N az összehasonlított értékek száma..

A hálózatokat a tanítás során minden egyes iterációs lépés alatt négyzetes középérték-hibával értékeli ki a tanítási algoritmus (E). Így nyomon követhető a tanítás állapota, a hiba növekedése a legtöbb esetben problémát jelent. A kiértékelés során a tanításhoz alkalmazott függvényekkel való tesztelést elvlasztjuk a háló számára ismeretlen függvényektől, mivel a tanításhoz alkalmas függvények hibája az azokhoz való idomulást mutatja, míg az ismeretlen függvények hibája a keresett függvény-kapcsolat megtanulásának minőségét fejezi ki. Az effajta vizsgálatok során a tetszőleges dimenziójú mennyiségek szerepelhetnek a hibaszámításban, így E értéke dimenziótlan.

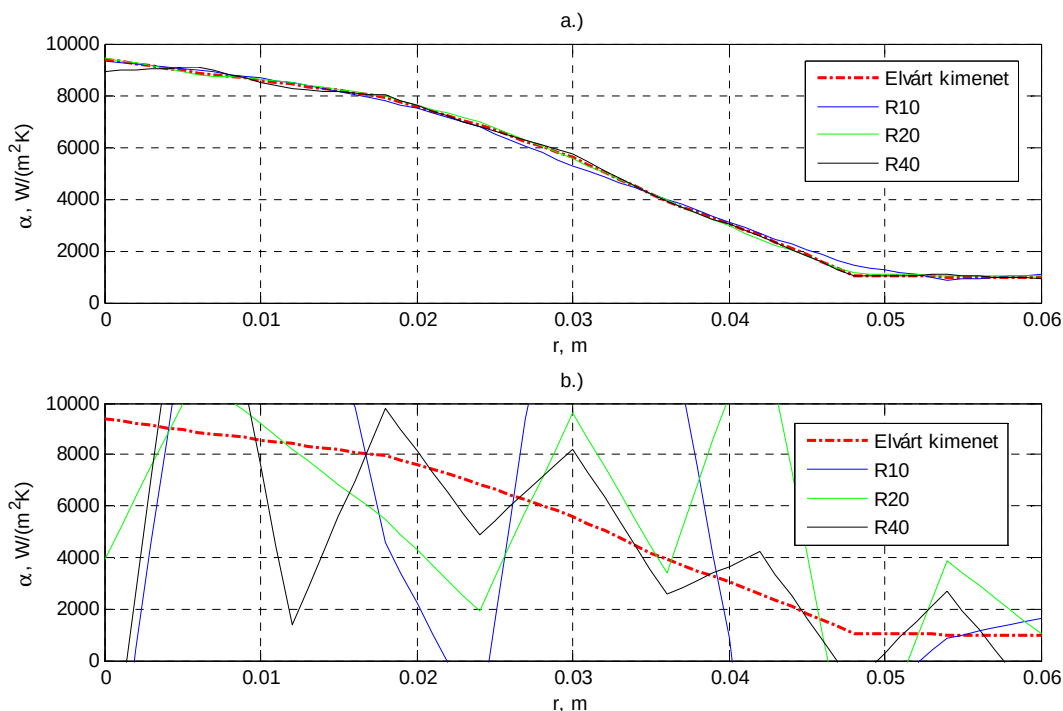
A betanított hálózatok összehasonlítását egy átlagolt, statisztikus hibaértékkel ($\bar{E}$) végeztem. A hálózatok által számított hőátadási függvényeket szintén négyzetes középérték-hibával hasonlítottam össze a sugár mentén 11 pontban, melyek egymástól egyenlő távolságra, 6-6 mm-re helyezkedtek el. A hibákat 100-100 darab a hálózat számára addig ismeretlen függvényekre számítottam ki és ezek átlaga határozta meg $\bar{E}$ értékét. Mivel itt már kizárólag hőátadási tényezőket hasonlítok össze, így $\bar{E}$ mértékegysége $W/(m^2 \cdot K)$.

Az eredmények során bemutatott példa-függvények a hálózat számára ismeretlenek voltak, azaz a tanítás során nem lettek felhasználva.

6.1. Táblázatos függvény-megadás lineáris interpolációval

Az 5.2-es alfejezetben ismertetettek alapján ez a függvény-megadási módszer 22 bemeneti és 11 kimeneti ponttal rendelkező hálózatot igényel. A zajmentes függvényekkel való tanítás minden hálómérettel 1000 iterációs lépésen át tartott, 200 darab eltérő tanítási függvénnyel, melyek közül 1-1 volt a vizsgált tartomány alsó és felső határát jelentő konstans 1000, illetve 10000 $W/(m^2 \cdot K)$ -es hőátadási tényező függvény és az ahhoz tartozó hőmérséklet-görbe. Az E hibaérték az egyes iterációk során folyamatosan csökkent, így az 1000 iterációs lépés során helyesen tanulta a függvénykapcsolatot, többre pedig az eredmények ismeretében nem volt szükség.

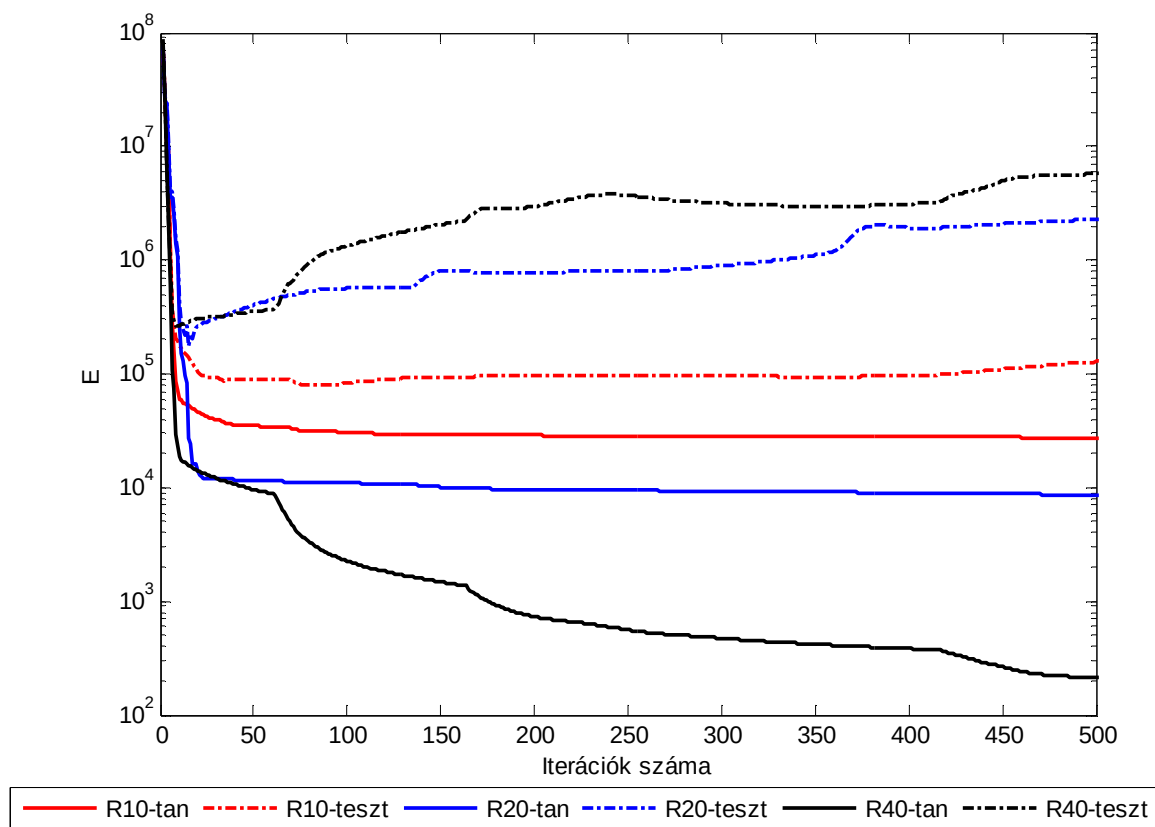
Az így kapott hálózatok a zajmentes bemenetre igen jó választ adtak, ám zajjal terhelt bemenetre nem voltak felkészülve, így ezekre használhatatlan eredményeket adtak. Ezt szemlélteti a 12. ábra. Az R10, 20 és 40 jelölés a rejtett rétegben található neuronok számát jelöli.



12. ábra –Táblázatos megadás lineáris interpolációval, zajmentes tanítás a.) zajmentes bemenet b.) zajos bemenet

A hálózatok zajra való felkészítéséhez a korábban használt tanítási függvények bemeneti hőmérsékleteit zajjal terheltem, és így tanítottam a hálózatokat. Itt azonban E értéke a háló számára ismeretlen tesztfüggvényeknél bizonyos iterációs szám után elkezdett növekedni, miközben a tanításhoz felhasznált függvényeknél pedig továbbra is monoton csökkent. (13. ábra). Az R10, 20 és 40 jelölés a rejtett rétegben található neuronok számát jelöli, a „tan”, illetve a „teszt” pedig a tanulásra felhasznált és a tanulása fel nem használt (teszt) függvényekkel történt futtatást jelöli.

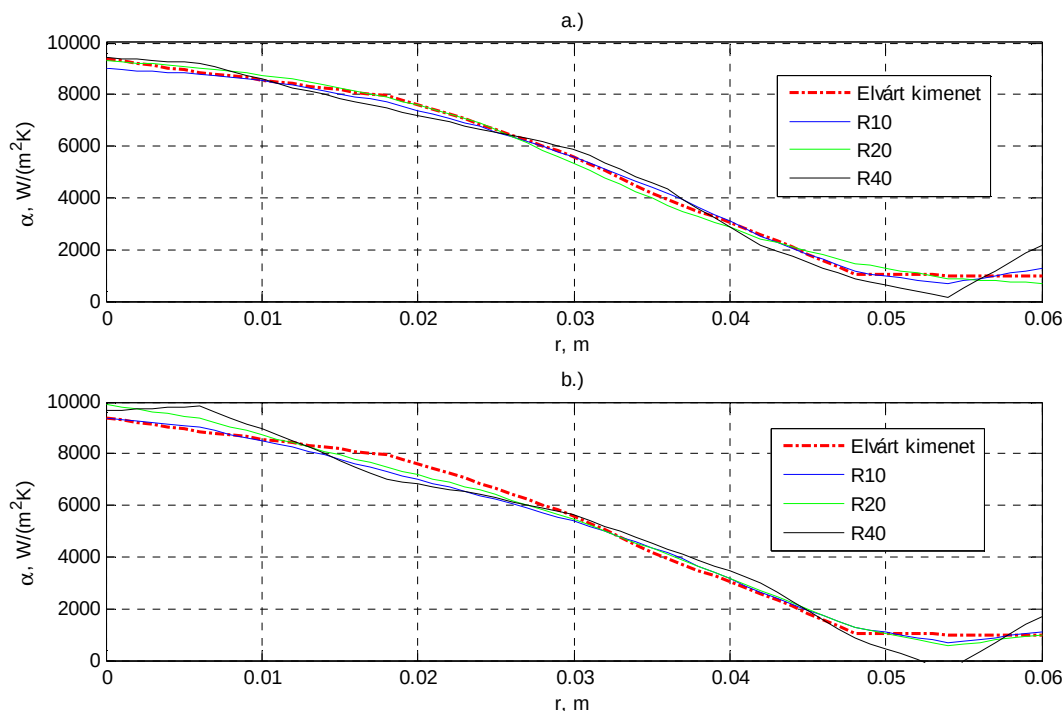
Ezt a jelenséget nevezzük túltanulásnak. Túltanuláskor a hálózat a tanítás során felhasznált függvényekhez túlságos ragaszkodást mutat, és emiatt az ismeretlen függvényeket nem tudja helyesen megoldani. A túltanulást esetünkben három tényező befolyásolhatja: a tanítási iterációk száma, a rejtett réteg mérete és a bemenetek zajossága. A tanítási iterációk száma és a rejtett réteg mérete adja meg úgy mond a hálózat intelligencia szintjét, hasonlatosan az biológiai tanuláshoz: minél tovább tanulunk egy specifikus témakört, annál otthonosabban mozgunk benne, más témakörök tudásának rovására, illetve agyi kapacitásunk határozza meg, hogy mennyire vagyunk képesek az adott témakört elsajátítani. A bemenet zajossága pedig minden egyes esetet eltérővé tesz, így a megtanult esetek jóval kevésbé tudják lefedni a teljes tudásterületet. E megfontolások alapján a túltanulást elkerülhetjük kevesebb iterációs lépéssel, illetve kisebb rejtett réteggel, mivel a zaj szórásának csökkentése a gyakorlatban nem minden esetben lehetséges.



13. ábra – Tanulási görbék, táblázatos megadás, lineáris interpoláció, zajjal terhelt tanítás

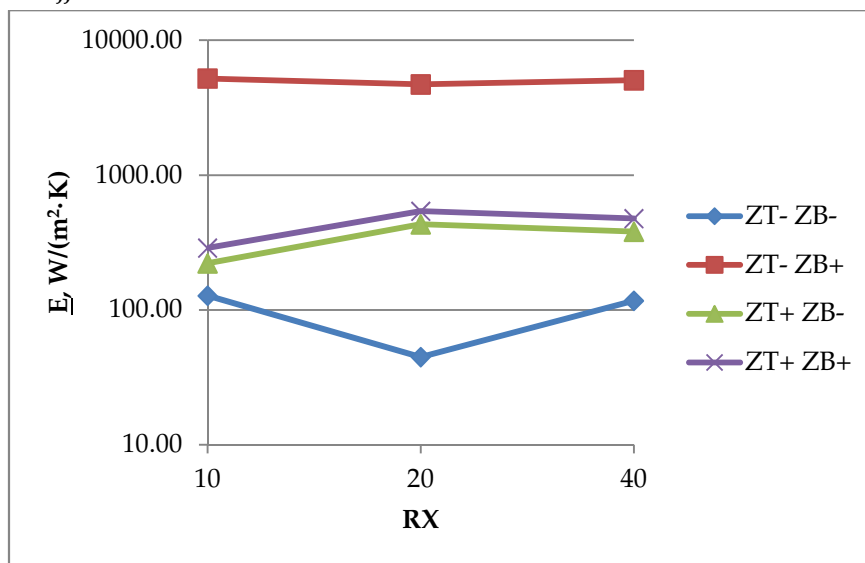
A 13. ábra tanulsága alapján a 10 rejtett neuronnal rendelkező hálózatot 500 iterációs lépésen keresztül tanítottam, míg a 20 és 40 rejtett neuronnal rendelkező hálózatokat pedig csak 50-50 lépésen át.

Az ilyen módon tanított hálózatok pontosságát mutatja be egy példán keresztül a 14. ábra.



14. ábra – Táblázatos megadás lineáris interpolációval, zajjal terhelt tanítás a.) zajmentes bemenet b.) zajos bemenet

A létrehozott hálózatok statisztikus hibaértéke leolvasható a 15. ábráról, pontos értékeit a 3. táblázat tartalmazza. RX jelöli a rejtett neuronok számát, ZT- és ZT+ a zajmentes, illetve zajjal terhelt tanítást jelöli, ezzel analóg módon ZB- és ZB+ jelöli a zajmentes és zajjal terhelt tesztelési, azaz a hálózat számára ismeretlen bemeneteket.



15. ábra – Statisztikus hibaértékek, táblázatos megadás, lineáris interpoláció

3. táblázat - Statisztikus hibaértékek, táblázatos megadás, lineáris interpoláció

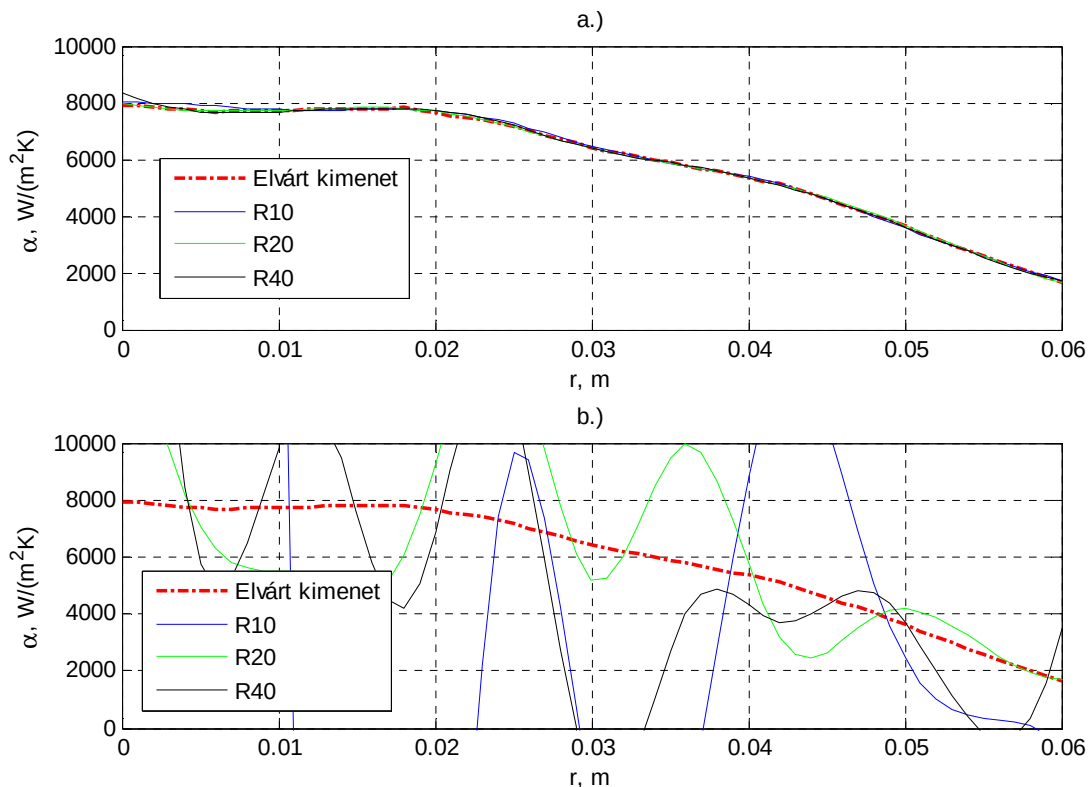
$\bar{E}, W/(m^2 \cdot K)$	ZT-	ZT-	ZT+	ZT+
RX	ZB-	ZB+	ZB-	ZB+
10	127	5205	221	287
20	44	4707	431	539
40	116	5060	380	476

Az adatokból levonhatjuk a tanulságokat, melyek analógak a példákon megfigyelhető jelenségekkel. A zajmentes esetre felkészített hálózat zajmentes bemenetekre adja a legjobb eredményeket, ahogy ez várható volt, viszont zajos bemenetekre a hálózat használhatatlan. Zajjal terhelt tanítás után a hálózat veszített a pontosságából a zajmentes bemenetek esetén, viszont egy nagyságrenddel pontosabb eredményt tud adni a zajos bemenetekre. Ezenfelül a rejtett réteg neuronjainak száma is meghatározó. Amennyiben teljesen eltekintünk a zajtól, 20 neuron esetén kapjuk a legjobb eredményt, míg zaj esetén 10 neuron alkalmazása a célszerű a három vizsgált hálóméret közül. Ebből azt a következtetést vonom le, hogy a túltanulás zajmentes esetben is fennáll 40 rejtett neuron alkalmazásakor, illetve zajjal való terheléskor nem sikerült tökéletesen kiküszöbölni a túltanulást 20 rejtett neuronnal, de valószínűsítem, hogy nem is lehet.

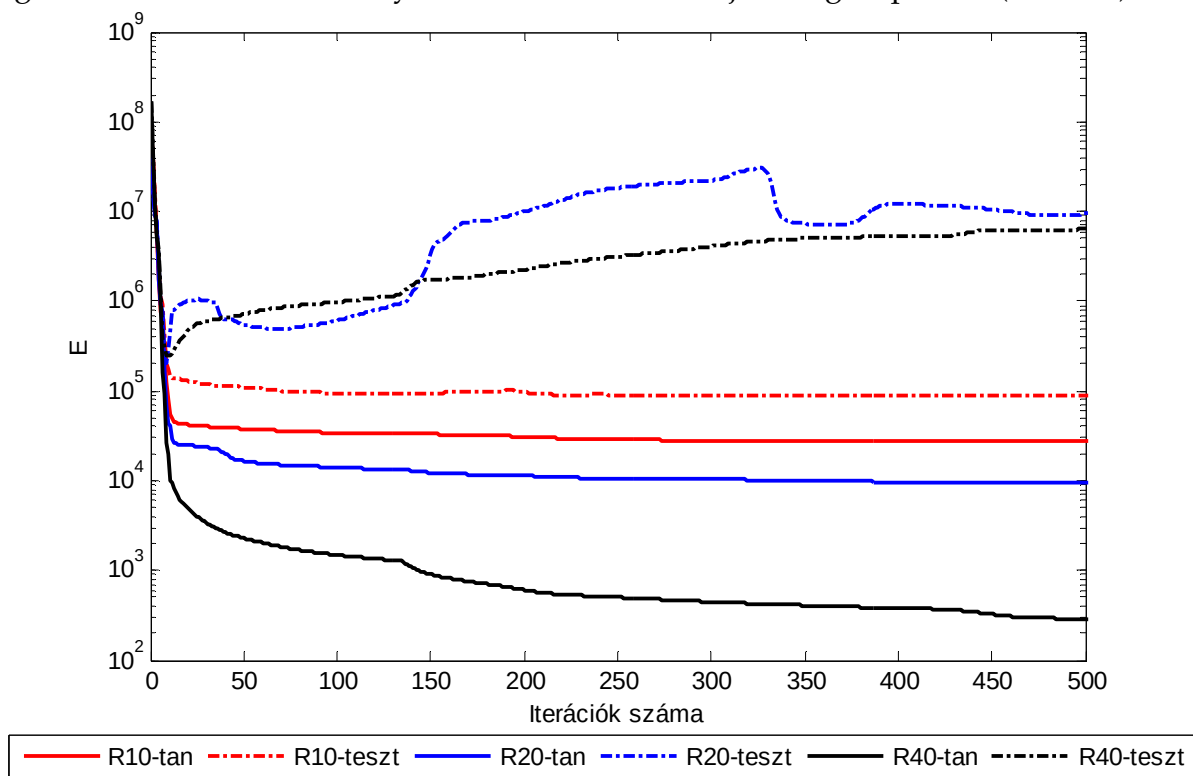
Összességében a hálózat az inverz feladatot ezzel a megadási formával sikeresen megoldotta, a zajjal terhelt eset pontosabb megoldása pedig további kutatómunkát igényel.

6.2. Táblázatos függvény-megadás spline interpolációval

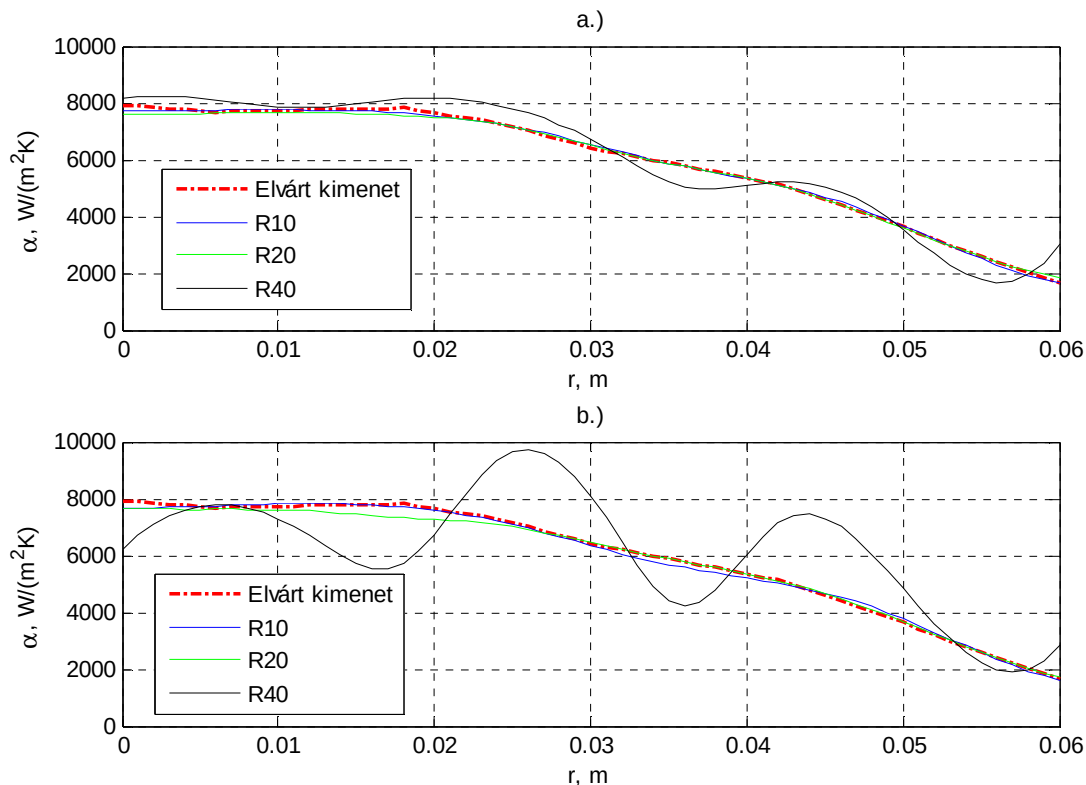
A korábbi esethez képest ez annyiban tér el, hogy ismert, diszkrét pontok között spline görbéket alkalmazunk, így folytonosabb görbéket kapunk a keresett paraméterek számának megnövelése nélkül. A hálózatokat szintén 200 darab eltérő függvénnyel tanítottam, melyek szintén tartalmazták a vizsgált tartomány alsó és felső határát. A tanítás paraméterezése is igen hasonló az előző esetéhez képest, így a zajmentes tanítás minden hálóméretnél 1000 iterációs lépésen át zajlott. Az így betanított hálózatok eredményére mutat egy példát a 16. ábra.



A zajmentes tanítás a hálózatot szintén nem készítette fel a zajjal terhelt bemenetek megoldására, így a korábban alkalmazott tanítási függvényeket zajjal terhelve megismételtem a tanítást, melyben ismét a túltanulás jelensége lépett fel (17. ábra).

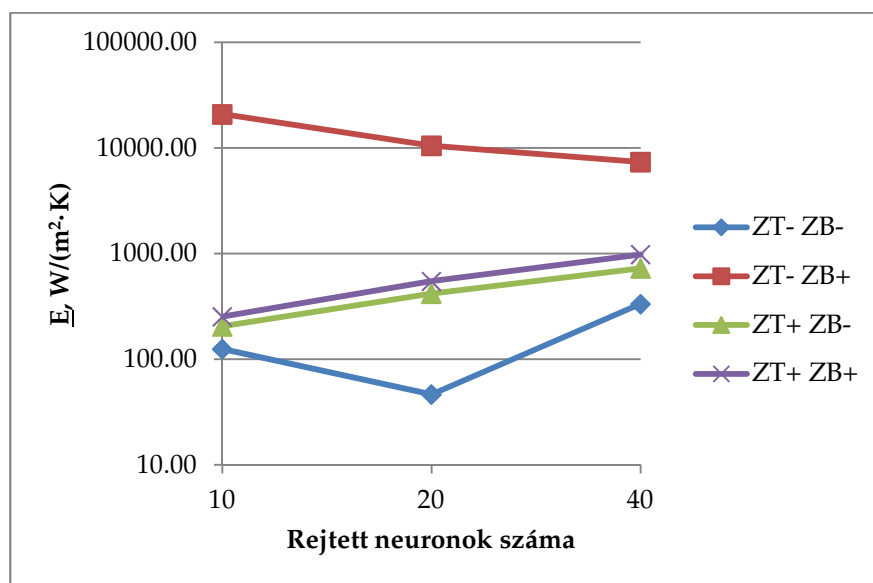


A görbék alapján a 10 rejtett neuronnal rendelkező hálózatot 500 lépésen át iteráltattam, a 20 és 40 rejtett neuronnal rendelkezőket pedig 50 lépésen át, csakúgy, mint a lineáris interpoláció alkalmazásakor. A zajjal terhelt tanítás eredményét egy példában a 18. ábra szemlélteti.



18. ábra- Táblázatos megadás spline interpolációval, zajjal terhelt tanítás a.) zajmentes bemenet b.) zajos bemenet

Az spline interpolációs, táblázatos függvéymegadásra betanított hálózatok statisztikus hibái leolvashatóak a 19. ábra a pontos számadatokat pedig a 4. táblázat tartalmazza.



19. ábra – Statisztikus hibaértékek, táblázatos megadás, spline interpoláció

4. táblázat - Statisztikus hibaértékek, táblázatos megadás, spline interpoláció

$\bar{E}$, W/(m ² ·K)	ZT-	ZT-	ZT+	ZT+
RX	ZB-	ZB+	ZB-	ZB+
10	125	20958	205	253
20	46	10563	420	548
40	334	7392	730	985

Az eredmények hasonlóak a lineáris interpolációnál tapasztaltakhoz, mind jellegre, mind nagyságrendben, bár a hibaértékek valamivel nagyobbak, ez egyértelműen a spline görbe hatása, ugyanis, mint ahogy a példákön látszik, a kimenetek hibáit felnagyítja az interpoláció oszcillációs hatása. Az adatok különösebb magyarázatot az előzmények ismeretében nem igényelnek, a legpontosabb eredményt szintén a teljesen zajmentes eset adta, míg a zajjal terhelt bemenetek esetén elengedhetetlen a hálózat zajjal terhelt függvényekkel való tanítása.

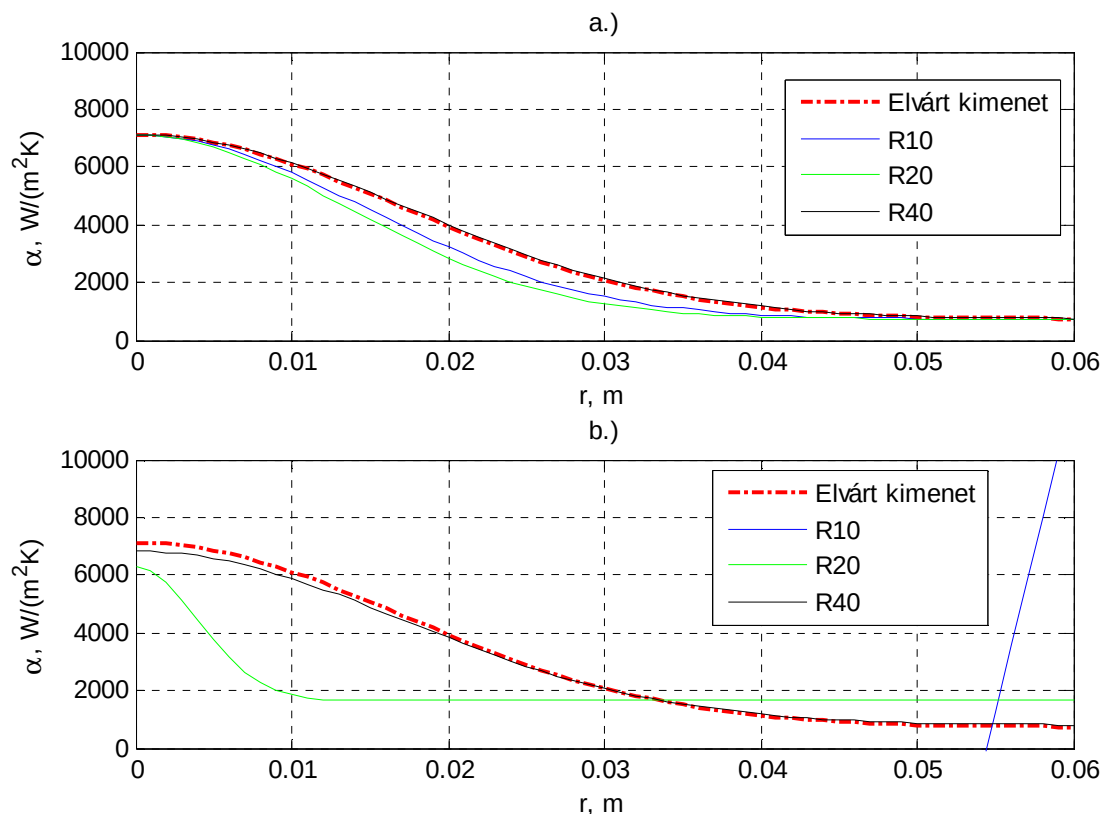
A teljesen zajmentes esetben a görbék a 16. ábrán gyakorlatilag fedik egymást, ez adja a legjobb megoldást, míg a zajokra fel nem készített hálózat kimenete teljesen használhatatlan, melyen a spline görbe okozta oszcilláció tovább ront. A zajra felkészített hálózatok követik az elvárt görbét, ám a 40 rejtett neuronnal rendelkező hálózat kisebb oszcillációba kezd. Viszont a 10 rejtett neuronnal rendelkező hálózat igen pontos megoldást ad.

Mindent összevetve a hálózat alkalmas ennek az inverz feladatnak a megoldására ezzel a függvénymegadási móddal is, mind zajmentes, mind zajjal terhelt esetben. A nagyobb pontosság elérését szintén lehetségesnek tartom.

6.3. Gauss-görbével való megadás

Ennél a megadási formánál is 22 bemeneti érték áll a háló rendelkezésére, ám csak 3 kimenete van, mely a korábban ismertetett (24) függvény 3 paramétere. A haranggörbe maximumának értéke 900 és 9000, függőleges eltolása 100 és 900, szórása pedig 0,01 és 0,1 között lett meghatározva. A tanításhoz e paraméterezéssel 1000 darab tanítási függvényt használtam fel.

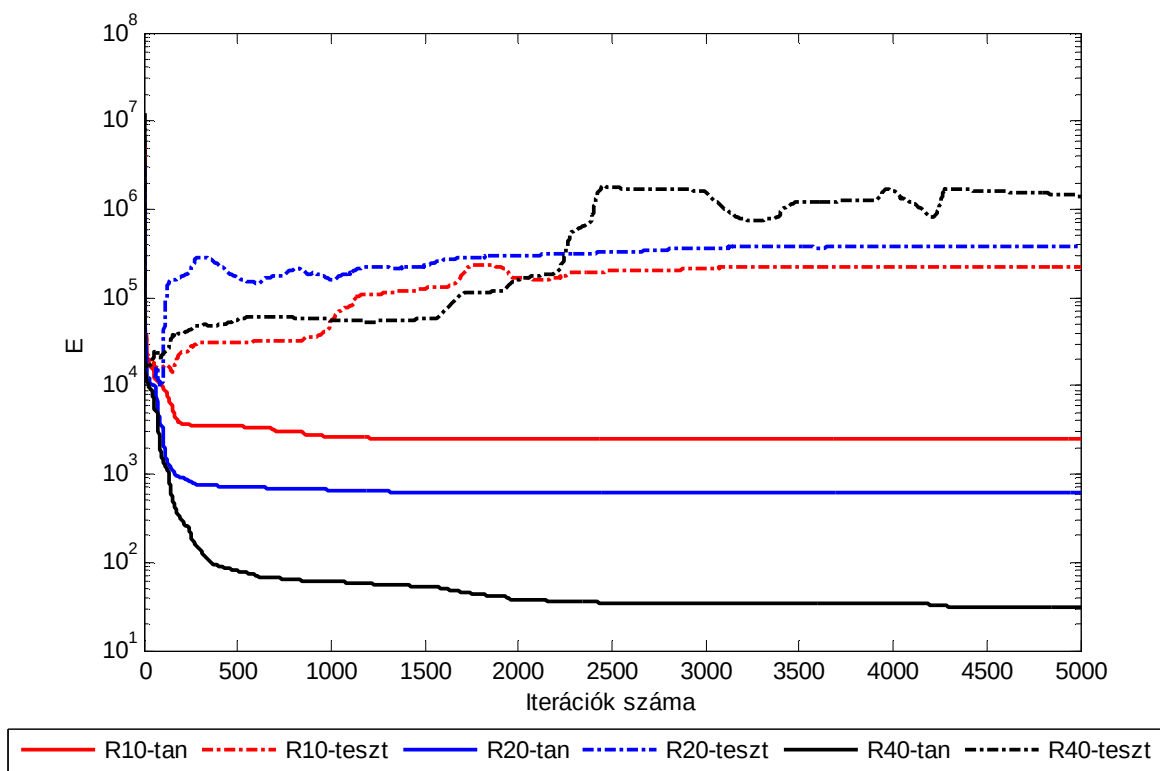
A zajmentes tanítás minden hálómérettel 5000 iterációs lépést igényelt. Ez jóval több tanítási függvényt és iterációs lépést jelent, mint korábban, ám ezeket a függvény-kapcsolat bonyolultsága szükségessé tette. Az így kapott hálózatok eredményeit egy példában a 20. ábra mutatja be. A hálózat 40 rejtett neuronnal még a zajos bemenetre is kellően pontos választ adott, ám a 10 rejtett neuronnal rendelkező hálózat használhatatlan eredményt szolgáltatott.



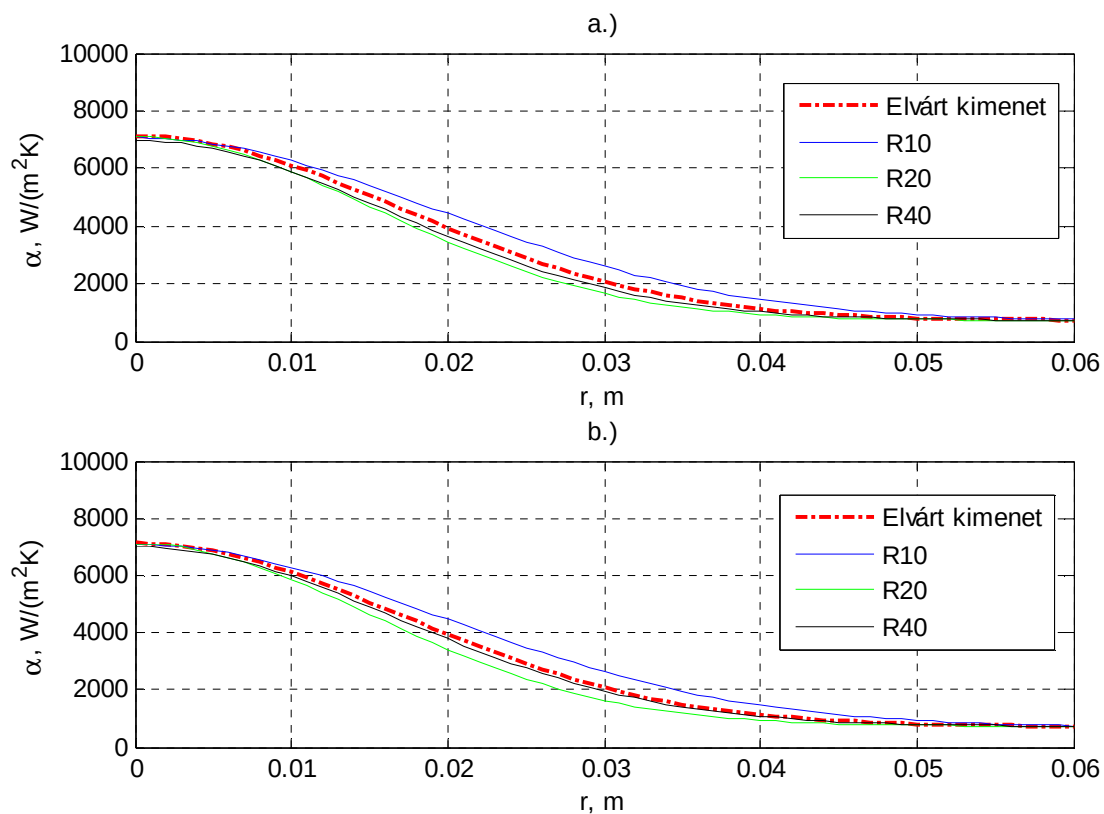
20. ábra- Gauss-görbés megadás, zajmentes tanítás a.) zajmentes bemenet b.) zajos bemenet

Továbbá ennél a függvényt megadási módnál is végeztem zajjal terhelt függvényekkel való tanítást, bár az eredmények kevésbé indokolják ennek szükségességét. Így egy érdekes jelenséggel találkoztam. Bár a tanulási hiba (E) túltanulás jeleit mutatta (21. ábra), az alacsonyabb iterációs tanítások nem voltak célravezetők, a hálózatnak szüksége volt 5000 iterációs lépésre jelen esetben is. Feltételezésem szerint ezt az eltérő nagyságrendű paraméterekkel lehet magyarázni.

A hibával terhelt tanítás eredményeit a 22. ábra mutatja be, mely látszólag nagyobb hibát mutat zajjal terhelt teszteléskor is, mint a zajmentes tanítás során elkészült hálózatok.

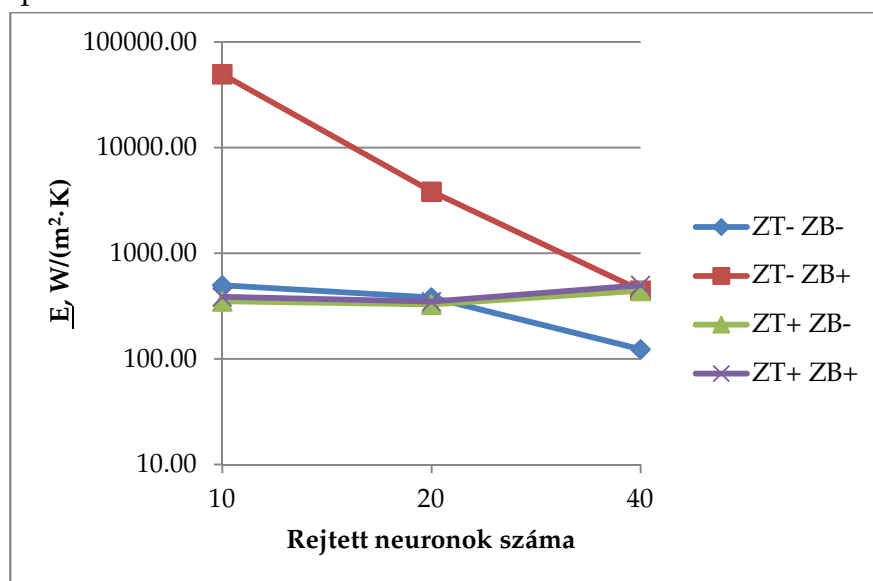


21. ábra - Tanulási görbék, Gauss-görbés megadás, zajjal terhelt tanítás



22. ábra - Gauss-görbés megadás, zajjal terhelt tanítás a.) zajmentes bemenet b.) zajos bemenet

A Gauss-görbés megadásra betanított hálózatok statisztikus hibaértékét a 23. ábra mutatja be, a pontos számadatokat az 5. táblázat tartalmazza.



23. ábra - Statisztikus hibaértékek, Gauss-görbés megadás

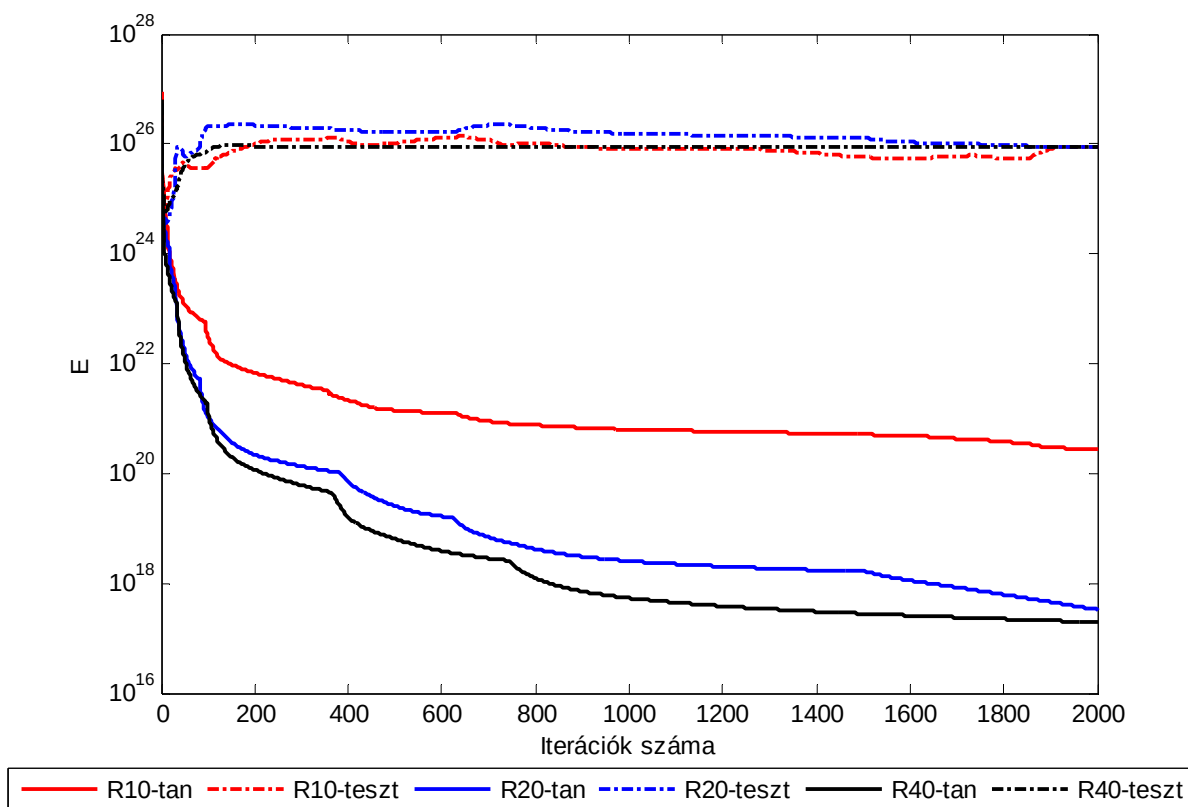
5. táblázat - Statisztikus hibaértékek, táblázatos megadás, spline interpoláció

$\bar{E}, W/(m^2 \cdot K)$	ZT-	ZT-	ZT+	ZT+
RX	ZB-	ZB+	ZB-	ZB+
10	500	49747	351	387
20	383	3848	328	349
40	123	445	443	500

Az adatok alapján ez a megközelítés nagyszámú rejtett neuront igényel a pontos megoldáshoz, a zajjal terhelt bemenetekre egy nagyobb háló gyakorlatilag érzéketlen, ami kifejezetten alkalmassá teszi valós mérések során való alkalmazásra, amennyiben a mérések során, vagy CFD szimulációval bebizonyosodik, hogy egy felütköző sugár hőátadási tényezőjének eloszlása sugár mentén valóban leírható Gauss-görbével.

6.4. Polinomiális függvénymegadás

Ebben az esetben 7 kimenete volt a hálózatnak, mellyel egy hatodfokú polinomot paraméterez, míg 22 darab bemenettel rendelkezik, a korábbi módszerekkel megegyezően. A hálózatok 200 darab zajmentes tanítási függvény alapján tanultak, 2000 iterációs lépéssel. A tanulási hibák (E) láthatóak a 24. ábrán. Bár a hálózat a számára ismert függvények felé monoton csökkenő hibát mutat mindhárom hálómérettel, az ismeretlen függvények felé semmiféle érdemleges változást nem mutat, tehát a tanulás sikertelen. Nagyobb számú iterációval sem érdemes tanítani a hálózatot, mivel az ismert függvények felé mutatott hiba csökkenése is minimálissá vált a 2000 iterációs lépés végére.

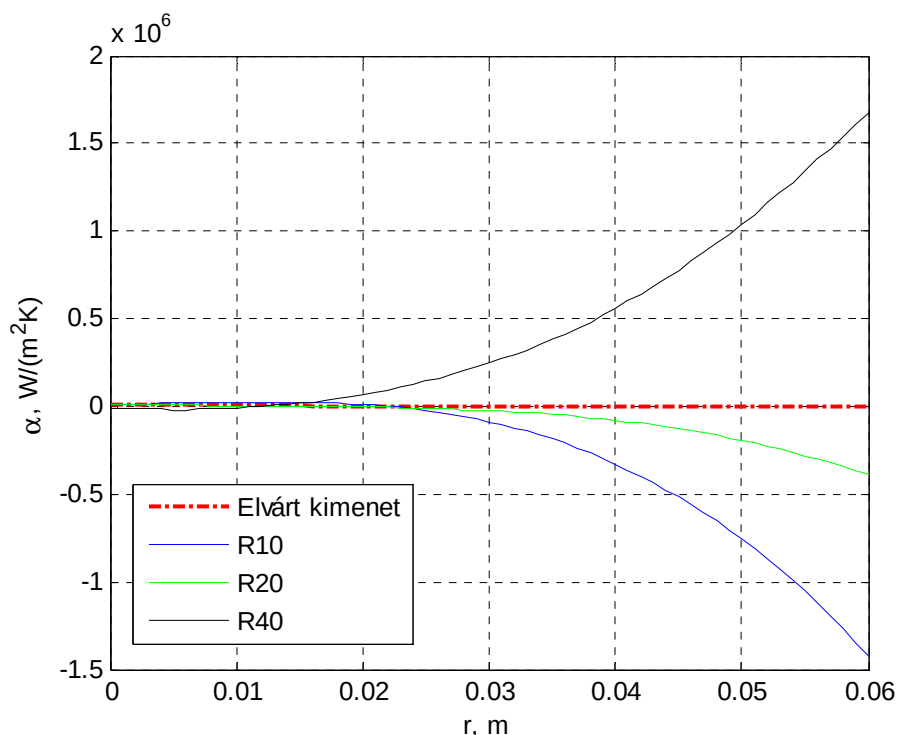


24. ábra - Tanulási görbék, polinomiális megadás, zajmentes tanítás

A tanulás sikertelenségének számos oka lehet. A függvénykapcsolat a jóval bonyolultabb, mint bármely korábbi függvénymegadásnál, ezenfelül a kimeneti paraméterek közül a legnagyobb 10^{13} -es nagyságrendbe esett, míg a legkisebb 10^2 -esbe, arról nem is beszélve, hogy az egyes paraméterek minimális megváltozása is jelentősen módosíthatja a függvény alakját.

A megoldás sikertelenségét egy példában a 25. ábra szemlélteti, ahol az elvárt függvénytől a hálózat kimenetei két nagyságrendnyi hibát vétenek, mely egyértelművé teszi, hogy ezt a függvény-megadási módszer az alkalmazott hálózatok képtelenek megtanulni.

Összességében ezt a megközelítést alkalmatlannak vélem az inverz feladat megoldására.



25. ábra – Polinomiális megadás, zajmentes tanítás

7. További kutatási lehetőségek

A 3.1 alfejezetben leírt mérési összeállítás megvalósítása az elsődleges továbblépési irány. A dolgozatban bemutatott eljárás alkalmazása valós, mért értékekkel magasabb szintre emeli az egész kutatást, valamint pontos képet kaphatunk a felütköző sugár hőátadási tényezőjének sugár-menti eloszlásáról.

Továbbá a hálózatok pontosságának határait jobban megismerhetjük, ha több hálómérettel, eltérő iterációs lépéssel tanítjuk a neurális hálókat. Más hálózat típusok alkalmazásával is érdemes lehet kísérletezni.

A Gauss-görbés függvény-megadás a mérési zajra érzéketlennek bizonyult, ezzel bizonyítva, hogy speciális függvények használatával a mérési zajt kiküszöbölhetjük, ezért a valós mérések ismeretében érdemes lehet további függvény-megadási módszerekkel kísérletezni.

Az eredményeket egy anyaggal és egy geometriával mutattam be, ám ez korántsem elég gyakorlati alkalmazásokra. Ehhez egyéb anyagokra és geometriákra is bizonyítani kell a módszer alkalmazhatóságát.

8. Összefoglalás

A dolgozat során egy körszimmetrikus felütköző sugár hőátadási tényezőjének eloszlását határoztam meg a célfelület hátoldalán mért zajos és zajtalan hőmérséklet-eloszlásból. Ehhez neurális hálózatokat alkalmaztam különböző hálóméretekkel.

A kutatás során valós mérésre nem került sor, így véges differencia-algoritmuson alapuló szimulált méréseket végeztem. Az algoritmus felelt a neurális hálózatok betanítására és tesztelésére alkalmazott függvény-párok létrehozásáért.

Négyféle függvénymegadási módszert vizsgáltam, a táblázatos megadást lineáris és spline interpolációval, a Gauss-görbés megadást és a polinomiális megadást. A táblázatos megadási módszerek alkalmazása sikeres megoldást jelentett, bár a zajjal terhelt mérések hatására érzékelhető hibákat vétettek. A Gauss-görbés megadás pontos eredményeket szolgáltatott mind zajtalan, mint zajos mérések esetén is, bár a függvény alakját jelentősen korlátozza ez a módszer. A polinomiális megadás nem mutatott alkalmazható eredményeket, így ez a módszer sikertelen volt.

Mindent összevetve elmondható, hogy a neurális hálózatok a kijelölt feladat megoldására alkalmasak, a további kutatások várhatóan olyan eredményeket fognak felmutatni, melyek a gyakorlatban is alkalmazhatóak.

9. Irodalomjegyzék

- [1] M. NECATI ÖZISIK, HELCIO R.B. ORLANDE: *Inverse Heat Transfer: Fundamentals and Applications*, Taylor and Francis, 2000
- [2] RAUDENSKY M., WOODBURY K. A., KRAL J., BREZINA T.(1995) : *Genetic algorithm in solution of inverse heat conduction problems*. Numerical Heat Transfer Part B: Fundamentals, 28/3
- [3] RAUDENSKY M., HORSKY J., KREJSA J., SLAMA L. (1996): *Usage of artificial intelligence methods in inverse problems for estimation of material parameters*. International Journal of Numerical Methods for Heat & Fluid Flow, 6/8
- [4] LIU G. R., LEE J. H., PATERA A. T., YANG Z. L., LAM K. Y. (2005): *Inverse identification of thermal parameters using reduced-basis method*. Computer Methods in Applied Mechanics and Engineering, 194/27-29
- [5] G. ÁCS, B. CZÉL (2011): *Estimation of the Heat Transfer Coefficient by Neural Network*, Proceedings of 10th International Conference on Heat Engines and Environmental Protection, Balatonfüred, Hungary, May 23-25, 2011.
- [6] GUO Q., SHEN D., GUO Y., LAI C.H. (2007): *Parallel genetic algorithms for the solution of inverse heat conduction problems*. International Journal of Computer Mathematics, 84/2
- [7] CZÉL B., GRÓF GY. (2012): *Inverse identification of temperature-dependent thermal conductivity via genetic algorithm with cost function-based rearrangement of genes*, International Journal of Heat and Mass Transfer, 55/15-16. .
- [8] CZÉL B., GRÓF GY. (2009): *Genetic Algorithm-Based Method for Determination of Temperature-Dependent Thermophysical Properties*. International Journal of Thermophysics, 30/6
- [9] MCCULLOCH, W. AND PITTS, W. (1943): *A logical calculus of the ideas immanent in nervous activity*. Bulletin of Mathematical Biophysics, 7:115 - 133.
- [10] BEN KRÖSE, PATRICK VAN DER SMAGT: *An Introduction to Neural Networks*, University of Amsterdam, 8th edition, 1996
- [11] ALTRICHTER MÁRTA, HORVÁTH GÁBOR, PATAKI BÉLA, STRAUZS GYÖRGY, TAKÁCS GÁBOR, VALYON JÓZSEF: *Neurális hálózatok*, Panem könyvkiadó, 2006
- [12] RYFA, A., BIALECKI, R., FACCHINI, B. AND TARCHI, L.(2009): *Application of the inverse analysis for boundary condition retrieval*, Inverse Problems in Science and Engineering, 17:6, 829 — 853
- [13] RYFA, A., BIALECKI, R. (2011): *International Journal of Heat and Fluid Flow*, International Journal of Heat and Fluid Flow 32 (2011) 1024–1035
- [14] RYFA, A. AND BIALECKI, R.(2011) *The heat transfer coefficient spatial distribution reconstruction by an inverse technique*, Inverse Problems in Science and Engineering, 19: 1, 117 — 126

- [15] A. RYFA, D. INGHAM & R. A. BIALECKI (2012): *Direct and Inverse Methods for an Air Jet Impingement, Numerical Heat Transfer, Part A: Applications: An International Journal of Computation and Methodology*, 61:7, 547-568
- [16] ÖZISIK, M., ORLANDE, H. R. B.: *Inverse Heat Transfer: fundamentals and applications*, Taylor & Francis, New York, 2000