

EGY KLASZTEREZŐ GLOBÁLIS OPTIMALIZÁLÁSI MÓDSZER A PARAMÉTERBECSLÉSI FELADAT MEGOLDÁSÁRA¹²

CSENDES TIBOR

József Attila Tudományegyetem, Kalmár Laboratórium

1. A paraméterbecslési feladat

A természettudományok gyakori módszere, hogy a vizsgált rendszert egy modellel közelítik. Az illető rendszer működésének megértését az jelzi, ha viselkedését jól lehet leírni, jóslni a modell viselkedése alapján. A paraméterbecslési feladat függvény formájában adott modelleknek a mérési adatokhoz való illesztését és a paraméterek meghatározását tűzi ki célul.

A paraméterbecslési feladatot a numerikus matematikához, és azon belül az optimalizáláshoz, vagy más megközelítésben az operációkutatáson belül a matematikai programozáshoz szokás sorolni. A paraméterbecslési feladat gyakorlati alkalmazásáról megemlítünk néhány közleményt, amely a dolgozatunkban szereplő eredményekhez kötődik, illetve valamely vizsgált algoritmus használatával elért eredményt tartalmaz: [9,12,14,17,18,19,22,34,35,36].

A paraméterbecslési feladat fontosságát az is jelzi, hogy a 100 legtöbb hivatkozást kapott természettudományi közlemény között az egyetlen matematikai cikk D. W. Marquardt publikációja a nemlineáris paraméterbecslésről [23]. A feladat nehézségét pedig az jellemzi, hogy például a Fermat-sejtést is meg lehet fogalmazni paraméterbecslési feladatként [25].

E dolgozatban kizárólag a determinisztikus esettel foglalkozom, amikor tehát a statisztikai jellegű megközelítéssel szemben egy konkrét adatsorhoz legjobb illeszkedést biztosító modellparamétereket keressük. A dolgozat egy klaszterező globális optimalizálási módszert ismertet, és teszteredményeket tartalmaz annak numerikus hatékonyságára vonatkozóan.

A paraméterbecslési feladat szokásos alakja a következő:

$$\min_{z \in X} f(z) \quad (1)$$

ahol $f(z) : \mathbb{R}^n \rightarrow \mathbb{R}$.

¹Beérkezett: 1991. június 17.

²A kutatások anyagi feltételeit részben az 1074/1987 és 2879/1991 sz. OTKA pályázat és a 314/108/004/8 sz. DAAD ösztöndíj biztosították

$$f(x) = \sqrt{\sum_{i=1}^m (f_i - f_{\text{mod}}(s_i, x))^2}$$

$f_i \in \mathbb{R}$ adatpont; $f_{\text{mod}}(s, x)$ pedig a modell-függvény, s annak változója, s_i -k az alappontok, $i = 1, 2, \dots, m$; $m > 0$ egész. $X \subseteq \mathbb{R}^n$ kompakt halmaz, a lehetséges megoldások halmaza. X a legtöbb esetben egy n -dimenziós intervallum: $X = \{x \in \mathbb{R}^n : a_j \leq x_j \leq b_j\}$; $a_j, b_j \in \mathbb{R}$; $j = 1, 2, \dots, n$. Ebben az esetben a lehetséges megoldások halmazát megadó feltételek egyszerű korlátok a paraméterekre. Az (1) feladat célfüggvénye a modellfüggvény optimális legkisebb négyzetes értelmű illesztését adja meg az f_i adatpontokhoz.

Az (1) feladat megoldása során gyakran feltételezik [13], hogy $f(x)$ nem multiextremális, azaz csak egy helyi minimuma van, vagy hogy legalábbis egy megfelelő kezdőpont áll rendelkezésre a helyi kereső algoritmus számára. Mivel számos olyan gyakorlati paraméterbecslési feladatot találtunk, amelynek célfüggvényének több helyi minimumpontja volt, ezért megvizsgáljuk az összefüggést az (1) nemlineáris paraméterbecslési feladat és a globális optimalizálási feladat között. Ez utóbbit a következő módon szokás definiálni:

Tekintsünk egy X kompakt halmazt az n -dimenziós valós térben, és egy $g(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ esetleg multiextremális függvényt. A feladat a $g(x)$ egy $x^* \in X$ globális minimumpontjának megkeresése, amelyre $g(x^*) \leq g(x)$ teljesül minden $x \in X$ pontra.

Azt találtuk, hogy az (1) feladat célfüggvényének szerkezete csak $f(x)$ nemnegativitását biztosítja, pontosabban érvényes a következő

1. ÁLLÍTÁS. Minden nem-negatív $g(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ valós függvényhez, m egészhez és $f_i \in \mathbb{R}$, $i = 1, 2, \dots, m$ valós számokhoz létezik olyan $f_{\text{mod}}(s, x)$ valós függvény, hogy

$$g(x) = f(x) = \sqrt{\sum_{i=1}^m (f_i - f_{\text{mod}}(s_i, x))^2}$$

minden $x \in \mathbb{R}^n$ pontra.

BIZONYÍTÁS. Legyenek például $g_i(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ valós, nem-negatív függvények ($i = 1, 2, \dots, m$) úgy, hogy

$$g(x)^2 = \sum_{i=1}^m g_i(x).$$

Ilyen függvények léteznek, mivel például a $g_i(x) = g(x)^2/m$ függvények kielégítik ezt a feltételt. Ezután az az $f_{\text{mod}}(s, x)$ modell-függvény, amelyre $f_{\text{mod}}(s_i, x) = g_i(x)^{1/2} + f_i$, elegendő tesz az 1. Állításnak. \square

Vegyük észre, hogy a $g_i(x)$ függvények csaknem teljesen szabadon választhatók meg, és ezen a módon további tulajdonságokat tudunk biztosítani az $f_{mod}(s, x)$ függvényekre. Például, ha $g(x)$ folytonos, akkor $f_{mod}(s, x)$ is választható folytonosnak.

Az 1. Állítás szerint egy nemlineáris paraméterbecslési feladat célfüggvénye bármely nem-negatív valós függvény lehet. Tehát egy nemlineáris paraméterbecslési feladatnak lehet akárhány (sőt akár kontinuum sok) helyi minimumpontja. Az $f(x)$ szerkezete, azaz a legkisebb négyzetes alak csak $f(x)$ nem-negativitását biztosítja, semmi más szabályosságot nem eredményez. Ez indokolja általános globális optimalizálási módszerek alkalmazását olyan speciális szerkezetű feladatokra is, mint (1).

2. Klaszterező globális optimalizálási módszer

Itt egy olyan algoritmust tárgyalunk, amely az előző szakaszban definiált globális optimalizálási feladat megoldására szolgál. Ez a módszer csak a célfüggvény értékét tetszőleges pontban kiszámító szubrutinra támaszkodik. Ebben az értelemben nevezzük hagyományosnak. Az ilyen eljárástól nem várhatjuk, hogy garantált megbízhatósággal mindig megtalálja a globális minimumot, de azt igen, hogy nagy valószínűséggel jó becslést adjon rá.

A globális optimalizálási feladat nehézségének egyik oldalát mutatja az, hogy még egy pont helyi minimumpont voltának ellenőrzése is NP-teljes feladat [25]. A másik gyakran idézett tétel az, hogy nem létezik olyan determinisztikus algoritmus, amely csak a célfüggvényt és annak deriváltjait kiértékelő szubrutinhívások segítségével meg tudná határozni véges számú lépésben bármely globális optimalizálási feladat megoldását [38]. A hagyományos módszerek fejlesztésének célja így az marad, hogy minél hatékonyabb és megbízhatóbb módszert találjunk. Amint látni fogjuk, a gyakorlati feladatok nagy részére létezik kielégítő megoldást nyújtó eljárás.

A globális optimalizálási módszereket célszerű a felhasznált információ jellege szerint osztályozni. A már említett hagyományos algoritmusok csak a célfüggvény és esetleg annak deriváltjai értékét használják. Ez az osztály sok érdekes módszert tartalmaz, mint a többszörös helyi keresés mintávétele után (multistart) [38], a nemlineáris szimplex módszer [30], trajektória-követés [10], vagy az alagút-módszer [38]. A Lipschitz konstans ismeretére támaszkodó algoritmusok [20,28,29] először egydimenziós feladatokra voltak alkalmazhatók, ezeket a korlátozás és szétválasztás módszerével illetve térkitöltő görbékkel általánosították. Az utóbbi években az intervallum-aritmetikával meghatározott befoglaló függvényre [7,8,15,24,31] alapozó, és a célfüggvény képletét szimbolikus manipulációval kezelő [16] eljárások is elterjedtek.

A korábbi kiterjedt alkalmazási tapasztalatok és a szakirodalom alapján Boender és munkatársai algoritmusát [4,32,33,37] implementáltuk két verzióban, kisebb változtatásokkal. A két változat közötti különbség az, hogy az egyik egy kvázi-

Newton eljárást [13], a másik egy „véletlen séta” típusú helyi kereső módszert (UNIRANDI [21,38]) használ. A következőkben ezeket A, ill. B algoritmusnak fogjuk nevezni. Mindkettő deriváltmentes, tehát a megoldáshoz nem használják a célfüggvény parciális deriváltjait. Ez utóbbiak meghatározása ugyanis nehéz, illetve lehetetlen számos paraméterbecslési feladat esetén [18,19]. UNIRANDI robusztus, de rossz hatékonyságú helyi kereső módszernek bizonyult, míg a kvázi-Newton módszer érzékeny volt az indulópontra, de pontosabbnak mutatkozott [5]. A globális optimalizálási módszer és az UNIRANDI helyi kereső eljárás implementálásánál csak az említett publikációkra támaszkodtunk.

Az algoritmus fontos része az alkalmazott klaszterezési eljárás. Ennek elhagyásával egy egyszerű multistart módszert kapnánk. A klaszterezés célja az, hogy a gyűjtött információk alapján statisztikai jellegű eszközökkel határozza meg és különítse el a helyi minimumpontok vonzáskörzetét (azon globális minimumpontban végződő \mathbb{R}^n -beli folytonos görbék pontjai halmazát, amelyek mentén a célfüggvény szigorúan monoton csökkenő). Ez teszi aztán lehetővé, hogy csak a feltétlenül szükséges számú helyi keresést hajtsa végre algoritmusunk, és egy előre adott szintű megbízhatóság fenntartása mellett az algoritmusunk a lehető legkevesebb iterációs lépést tegyen.

A tárgyalt globális optimalizálási algoritmus röviden a következő:

1. lépés: Generáljunk N darab mintapontot egyenletes eloszlással az X halmazon, és adjuk őket a C aktuális mintához. Állítsuk elő a T transzformált mintát úgy, hogy vesszük a C -ben lévő pontoknak a legkisebb célfüggvényértékkel rendelkező γ százalékát.
2. lépés: Alkalmazzuk a klaszterezési eljárást a T ponthalmazra. Ha T minden pontja hozzárendelhető valamely klaszterhez, akkor folytassuk a 4. lépésnél.
3. lépés: Indítsunk helyi keresést azokból a T -beli pontokból, amelyek nincsenek valamely klaszterhez rendelve. Ha találtunk új helyi minimumpontot, akkor folytassuk az 1. lépésnél.
4. lépés: Határozzuk meg a legkisebb helyi minimumot, ez lesz a globális minimum közelítése.

Az itt említett helyi keresés vagy a kvázi-Newton módszer, vagy az UNIRANDI eljárás volt. Az eredeti közleményben ([4]) vizsgált két klaszterezési módszer közül a single linkage eljárást választottuk, mint ígéretesebbet. A klaszterezés célja azon mintapontok megtalálása, amelyekből indulva a helyi keresés valószínűleg egy már ismert helyi minimumponthoz vezetne. A klasztereket úgynevezett magpontok (helyi minimumpontok, és olyan mintapontok, amelyekből a helyi keresés egy már ismert helyi minimumponthoz vezetett) köré építjük. Két pont $d(x, x')$ távolságát az x^* helyi minimumpont környezetében a következő képlettel definiáljuk [4]:

$$d(x, x') = \sqrt{(x - x')^T H(x^*)(x - x')}.$$

A kvázi-Newton módszer az A algoritmusban megfelelő becslést ad a célfüggvény $H(x^*)$ Hesse-mátrixára. Az UNIRANDI esetén pedig az egységmátrix pótolja a $H(x^*)$ mátrixot (vö. [4]). Egy új x pontot akkor adunk egy klaszterhez, ha annak egy x' pontjára

$$d(x, x') \leq \left[\frac{\Gamma(1 + \frac{1}{2}n) |H(x^*)|^{1/2} m(X)}{\pi^{n/2}} (1 - \alpha^{1/(N'-1)}) \right]^{1/n}$$

teljesül, ahol $|H(x^*)|$ a $H(x^*)$ mátrix determinánsát jelöli, $m(X)$ az X halmaz egy mértéke, N' az összes generált mintapont száma, és $0 < \alpha < 1$ a klaszterezési eljárás paramétere [4].

Az eredeti algoritmuson végrehajtott két legfontosabb módosítás a következő:

1. Nem használunk "legmeredekebb lejtő" szerinti lépést az aktuális minta transzformálására; ennek hatékonyságát megvizsgáltuk az implementálás során, és elhagyhatónak bizonyult.
2. A célfüggvény változóit a globális optimalizálási rutin skálázza [13] a következő képlettel:

$$x'_i = \frac{2x_i - a_i - b_i}{b_i - a_i} \quad i = 1, 2, \dots, n.$$

Ezt természetesen a célfüggvény alakjától függetlenül is meg lehet tenni. A skálázásnak ugyan nem volt kimutatható hatása a tesztfüggvények megoldásának hatékonyságára, gyakorlati problémák esetén viszont nélkülözhetetlen.

Az implementálás eredményeként egy valamivel több mint 400 soros FORTRAN szubrutint kaptunk, amely 44 kilobyte memóriát igényel (a helyi kereső rutinok nélkül). Legfeljebb 15 paraméteres globális optimalizálási feladatok megoldására szolgál, de egyszerűen bővíthető nagyobb feladatokhoz. A globális optimalizáló rutin tárigénye lineárisan nő a változószámmal, az UNIRANDI-é szintén, de nem számottevő, a kvázi-Newton eljárásé pedig $n(n+1)/2$ -vel arányos. A globális optimalizáló szubrutin a felhasználó által megadott I/O utasításokon felül önállóan dokumentálja előrehaladását, és listát ad a növekvő sorrendbe rendezett helyi minimumokról.

3. Numerikus eredmények standard globális optimalizálási tesztfeladatokon

A numerikus tesztelést ROBOTRON R55M számítógépen végeztük. A programokat szimpla pontossággal kódoltuk (ez 7 értékes tizedesjegyet biztosított). A későbbiekben használatos standard időegység, azaz az S5 tesztfüggvény 1000 kiértékeléséhez szükséges idő az $x = (4, 0, 4, 0, 4, 0, 4, 0)^T$ pontban [10,38], tíz mérés

1. Táblázat. A tesztfüggvények globális minimumai és globális minimumpontjai

Teszt-függvény	$f(x^*)$	x_1^*	x_2^*	x_3^*	x_4^*	x_5^*	x_6^*
S5	-10,153206	3,99995	4,00014	4,00011	4,00016		
S7	-10,402947	4,00061	4,00072	3,99945	3,99958		
S10	-10,536416	4,00075	4,00061	3,99967	3,99948		
H3	-3,8627815	0,1146	0,5557	0,8525			
H6	-3,3223667	0,201536	0,149909	0,476906	0,27524	0,31160	0,65735
GP	2,9996490	0,000068	-1,0001				
RCOS	0,39788723	-3,1416	12,275				
	0,39788723	3,1416	2,2750				
	0,39788723	9,425	2,4750				
SHCB	-1,0316286	0,0899	-0,7126				
	-1,0316286	-0,0899	0,7126				
RB	0,0	1,0	1,0				

átlagában 2,0 másodpercrek adódott (a standard szórás 0,15 volt). A standard globális optimalizálási tesztfeladatok [10,38] részletes leírását a Függelékben adjuk meg. Ezek a feladatok főleg a globális optimalizálási módszerek hatékonyságának mérésére alkalmasak, mivel a globális minimum elkülönítése a nagyobb helyi minimumoktól ezekben a feladatokban nem nagyon nehéz. A hatékonyság fogalmának használata az optimalizálásban nem egységes. Dolgozatunkban is különféle hatékonysági mutatókat használunk, ezek viszont valamilyen értelemben mindig a közel azonos minőségű megoldás eléréséhez szükséges számítási ráfordítást jellemzik. Az összehasonlító táblázatokban, ahol lehetséges, az eredeti cikkek adatait adjuk meg – ezek esetenként némileg eltérnek a [4,10]-ben megadottaktól. Az A és B algoritmusokkal minden tesztfeladatot tízszer oldottunk meg. Mindegyik feladatra azonos eljárás-paramétereket (N, γ, α) használtunk, úgy beállítva, hogy minden feladatra mindkét eljárás mind a tízszer megtalálta a globális minimumot.

Úgy találtuk, hogy a számítógépes ráfordítás (a szükséges CPU-idő és a függvényhívások száma) arányos a helyi minimumok igényelt pontosságával. Eszerint különböző globális optimalizálási módszerek összehasonlításakor figyelembe kell venni azok pontosságát is. Ehhez először is meg kell határozni a globális minimumok pontos helyét és értékét. Az 1. Táblázat adja meg a standard globális optimalizálási feladatok ilyen adatait, jó egyezést mutatva Price eredményeivel [30]. Meg kell jegyezni, hogy más számítógépes pontossággal persze némileg eltérő értékeket is kaphatunk. Néhány globális minimumpontot csak 4-5 értékes jegyre adtunk meg: ezek esetén a célfüggvény az optimális értéket a megadott pont kis környezetében is felveszi.

Meghatároztuk a [10]-ben hivatkozott globális optimalizálási módszerek eredmé-

2. Táblázat. Az értékes jegyek száma a globális minimum becslésében

Módszer	Tesztfüggvény								
	S5	S7	S10	H3	H6	GP	RCOS	SHCB	RB
Branin	–	–	–	–	–	–	–	–	–
Törn*	3,0	2,9	3,2	4,3	2,6	3,9	4,5	–	–
Price*	6,2	5,3	5,8	6,4	6,0	3,9	6,3	–	–
De Biase	2,9	3,4	2,0	4,7	4,7	4,8	–	6,4	–
Boender	–	–	–	–	–	–	–	–	–
A*	7,0	6,8	6,7	6,8	6,9	4,3	7,2	7,1	10,1
B*	4,9	4,0	5,2	4,3	4,3	4,4	4,1	4,5	4,7

* Ezek a módszerek nem használják a célfüggvény deriváltjait.

nyeinek pontosságát, ahol ez lehetséges volt. A globális minimum értékes jegyeinek számát a

$$-\log \frac{f(x') - f(x^*)}{|f(x^*)|}$$

képlettel definiáljuk [3], ahol $f(x^*)$ az adott tesztfüggvény globális minimuma, és $f(x')$ ennek a program által adott becslése. A Rosenbrock (RB) függvény esetén a globális minimum nulla, ekkor a $-\log f(x')$ képletet használtuk.

A globális minimumok becslései értékes jegyeinek számát a 2. Táblázat foglalja össze néhány globális optimalizálási módszerre. Az A és B eljárások helyi kereső rutinjainak paramétereit úgy próbáltuk beállítani, hogy azok hasonló pontosságot érjenek el a különböző tesztfüggvényeken. Módszerünkél a pontosság és a megbízhatóság csaknem függetlenül volt hangolható.

A globális optimalizálási módszerek által a tesztfeladatok megoldásához igényelt függvényhívások számát tartalmazza a 3. Táblázat. Mivel azok a helyi kereső eljárások, amelyek nem használják a célfüggvény deriváltjait, rendszerint gyengébb hatékonyságúak, ezért az A és B módszerek inkább csak a többi deriváltmentes módszerekkel vehetők össze. Azokat a módszereket, amelyekről ismert, hogy deriváltmentesek, csillaggal jelöltük meg a 2–4. Táblázatokban. Az adatok egy-egy futásra vonatkoznak az első négy módszer esetén, a Boender és munkatársai módszerére vonatkozó értékek négy futás átlagaként, az A és B módszerhez tartozók pedig tíz független futás átlagaként adódtak. A 3. Táblázat szerint Boender és munkatársai módszere [4] a leghatékonyabb, míg Törn [38] és Price [30] eljárása gyengébb hatékonyságú, mint az A és B algoritmus.

A 4. Táblázat a tesztfeladatok megoldásához szükséges CPU-időt adja meg standard időegységben. Ezen adatok alapján az A és B algoritmusok határozottan gyorsabbnak mutatkoznak, mint a többi deriváltmentes módszer, és az A eljárás körülbelül olyan gyors, mint Boender és munkatársaié. Gyakorlati feladatokban

3. Táblázat. A függvényhívások száma

Módszer	Tesztfüggvény								
	S5	S7	S10	H3	H6	GP	RCOS	SHCB	RB
Branin	5500	5020	4860	-	-	-	-	-	-
Törn*	3649	3606	3874	2584	3447	2499	1558	-	-
Price*	3800	4900	4400	2400	7600	2500	1800	-	-
De Biase	620	788	1160	732	807	378	597	717	-
Boender	567	624	755	235	462	398	235	-	-
A*	990	1767	2396	216	1446	436	330	233	410
B*	1083	1974	2689	697	2610	386	464	267	1524

* Ezek a módszerek nem használják a célfüggvény deriváltjait.

azonban rendszerint több CPU-idő szükséges a célfüggvény kiértékeléséhez, mint a tesztfeladatok esetén, tehát a 4. Táblázat inkább az egyes módszerek adminisztrációs (overhead) számítási igényeit jellemzi.

A hagyományos, csak függvény-kiértékelésekre épülő módszerek heurisztikus jellegűek, így a numerikus teszt látszik az egyetlen összevetési lehetőségnek. A 3. és a 4. Táblázat alapján történő összehasonlítás akkor megalapozott, ha az egyes módszerek azonos megbízhatóságúak (nagy számú független futás hasonló számú esetben adja a globális minimumot), és azonos pontosságú eredményt szolgáltatnak (l. 2. Táblázat). Sajnos ezek a követelmények ritkán teljesülnek, sőt, a személyes tisztázási kísérletek is ellentmondó információkat eredményeztek. Mindenesetre, az A algoritmus mind megbízhatóság, mind pontosság szempontjából kimutathatóan jobb, mint az összehasonlításban szereplő többi algoritmus. Ez alapozza meg a hatékonysági összevetést.

Összegezve a numerikus tesztek eredményeit megállapíthatjuk, hogy Boender és munkatársai módszerének itt vizsgált két deriváltmentes változata határozottan jobb, mint a többi deriváltmentes eljárás. Az A algoritmus hatékonysága megközelelti az eredetiét. Ez a kvázi-Newton helyi keresővel ellátott globális optimalizálási eljárás jól használható sima globális optimalizálási feladatokhoz, amelyekben a célfüggvény parciális deriváltjai nem állnak rendelkezésre, vagy nehezen számíthatók. Ugyanez a globális optimalizálási kereteljárás az UNIRANDI direkt kereső módszerrel hatékony eszköz lehet nem-sima, vagy nem-differenciálható célfüggvényekhez.

4. Globális optimalizálási módszerek megbízhatósága

Szinte mindegyik globális optimalizálási módszer csak helyi információt használ, tehát a célfüggvény és annak első- és másodrendű parciális deriváltjai értékét bizonyos pontokban. Könnyű megmutatni [38], hogy nem létezik olyan determinisztikus algoritmus, amely véges sok ilyen függvényhívás révén minden globális

4. Táblázat. A standard időegységek száma

Módszer	Tesztfüggvény								
	S5	S7	S10	H3	H6	GP	RCOS	SHCB	RB
Branin	9,0	8,5	9,5	-	-	-	-	-	-
Törn*	10,0	12,4	14,4	8,0	15,6	4,1	3,7	-	-
Price*	13,9	20,0	19,7	7,5	47,5	2,8	4,4	-	-
De Biase	26,1	23,0	33,7	17,6	23,1	16,8	15,2	23,2	-
Boender	3,5	4,5	7,0	1,7	4,3	1,5	1,0	-	-
A*	3,0	4,9	7,0	1,2	4,2	1,3	1,4	1,2	1,0
B*	3,5	6,0	8,8	1,9	14,2	1,5	1,6	1,3	1,5

* Ezek a módszerek nem használják a célfüggvény deriváltjait.

optimalizálási feladatot meg tudna oldani. Emiatt a globális optimalizálási kutatások az egyre hatékonyabb és megbízhatóbb heurisztikus eljárások fejlesztésére koncentráltak.

A globális minimum vonzáskörzetének mértéke jellemzi az adott feladat nehézségét. Ennek az a magyarázata, hogy az olyan hagyományos globális optimalizálási módszerek, mint amilyent dolgozatunkban is ismertettünk, a kisebb vonzáskörzetű helyi minimumot kisebb valószínűséggel találják meg. Ebből a szempontból a leggyakrabban használatos tesztfeladatok meglehetősen könnyűek, inkább csak az algoritmusok hatékonysága vizsgálható velük.

Az alábbiakban egy olyan tesztfeladatot tárgyalunk, amely az algoritmusok megbízhatóságát méri, azaz meghatározza azt a nehézségi szintet, amelyhez tartozó globális optimalizálási feladatot az illető algoritmus még meg tud oldani. A javasolt n -dimenziós függvény egyszerűen számítható:

$$f(x) = \sum_{i=1}^n f_i(x_i) \quad (2)$$

ahol minden $i = 1, 2, \dots, n$ -re:

$$f_i(x_i) = x_i^6(\sin(1/x_i) + 2)$$

ha $x_i \neq 0$, és $f_i(0) = 0$. Ha $x_i \neq 0$, akkor $f(x)$ gradiense

$$g_i(x) = 6x_i^5(\sin(1/x_i) + 2) - x_i^4 \cos(1/x_i),$$

Hesse-mátrixa pedig

$$H_{i,j}(x) = 0 \quad (i \neq j)$$

$$H_{i,i}(x) = 30x_i^4(\sin(1/x_i) + 2) - 10x_i^3 \cos(1/x_i) - x_i^2 \sin(1/x_i)$$

minden $i, j = 1, 2, \dots, n$ -re. Különben $g_i(x)$ és $H_{i,j}(x)$ nulla ($i, j = 1, 2, \dots, n$). A gradiens és a Hesse-mátrix mindenütt folytonos \mathbb{R}^n -ben. Mivel nyilvánvalóan

$$\sum_{i=1}^n x_i^6 \leq f(x) \leq 3 \sum_{i=1}^n x_i^6,$$

ezért $f(x)$ globális minimuma \mathbb{R}^n -ben zérus, és ezt az értéket csak az origóban veszi fel.

1. TÉTEL. Az $f(x)$ függvénynek megszámlálhatóan végtelen sok helyi minimuma és maximuma van az n -dimenziós valós térben, és minden extrémális pontja a

$$-1 \leq x_i \leq 1 \quad i = 1, 2, \dots, n \quad (3)$$

n -dimenziós intervallumban van.

BIZONYÍTÁS. Tekintsük először az $n = 1$ esetet. Tegyük fel, hogy az első derivált nulla, és $x \neq 0$. Ekkor érvényes, hogy

$$6x(\sin(1/x) + 2) = \cos(1/x). \quad (4)$$

Ezen egyenlet jobboldala -1 és 1 között változik, míg a baloldal $6x$ és $18x$ között, az első derivált tehát csak a $(-1/6, 1/6)$ intervallumban lehet nulla. A (4) egyenlet jobboldala felveszi a -1 és az 1 értékeket a

$$[(1/2k\pi), 1/(2(k+1)\pi)) \quad k = \pm 3, \pm 4, \dots \quad (5)$$

intervallumok mindegyikében, míg ugyanezen intervallumokban

$$-1 < 6x(\sin(1/x) + 2) < 1. \quad (6)$$

Ez azt bizonyítja, hogy az $f(x)$ függvénynek legalább egy helyi minimuma és maximuma van minden ilyen intervallumban, mivel az első derivált folytonos függvény. Ezen extrémumpontok különbözőek, mert mind az (5) intervallumok belsejébe esnek. Ebből következik, hogy legalább megszámlálhatóan végtelen sok szélsőérték van a (3) intervallumban.

Könnyen belátható, hogy $f(x)$ első és második deriváltja nem lehet nulla ugyanazon helyen. Mivel a második derivált folytonos, ezért minden szélsőértékhez tartozik egy nyitott intervallum \mathbb{R} -ben, amelyben az az egyetlen szélsőérték. Ebből következik, hogy $f(x)$ -nek nem lehet \mathbb{R} -ben kontinuum sok helyi extrémuma.

Az $f(x)$ függvény szeparálhatósága miatt az előbbi érvelés nyilvánvalóan kiterjeszhető bármely pozitív egész n -re. \square

Ezek szerint a feltétel nélküli feladat helyi minimumainak halmaza nem csökken a (3) korlátok alkalmazásával. A globális minimumpont nem-izolált [1] abban az

értelemben, hogy előáll helyi minimumpontok torlódási pontjaként (és most ez az egyetlen torlódási pont). A globális minimum vonzáskörzete nyilvánvalóan nullamértékű. Az $f(x)$ függvény legfontosabb tulajdonsága, hogy minél kisebb helyi minimumról van szó, annál kisebb annak vonzáskörzete; ezt a tulajdonságot a globális optimalizálási módszerek megbízhatóságának mérésére lehet használni.

Az egydimenziós változat helyi minimumpontjait a célfüggvény értéke szerint csökkenő sorrendbe lehet rendezni. Az x helyi minimumpont N_x sorszáma a következő képlettel számítható:

$$N_x = 2 \lfloor |1/x|/2\pi \rfloor - 1 + (\text{sgn}(x) - 1)/2,$$

ahol $\lfloor \cdot \rfloor$ a legnagyobb, az argumentumnál nem nagyobb egész, sgn pedig a signumfüggvényt jelöli. Az egydimenziós esetben az x helyi minimumpont vonzáskörzetének A_x méretét jól lehet becsülni a (4) egyenlet segítségével, ha x abszolút értéke kicsi. Ekkor a (4) egyenlet baloldala közel nulla, és A_x közelítőleg a jobboldal x -szel szomszédos két zérushelyének távolsága:

$$A_x \approx \frac{2\pi}{\frac{1}{x^2} - \pi^2}.$$

Az $f(x)$ függvény numerikus, számítógépes változata természetesen különbözik az analitikustól, különösen az origó közelében. Emiatt fontos a tesztfüggvény gondos programozása.

A javasolt tesztfüggvény gyorsan számítható: az egydimenziós esetben $f(x)$ 1000 kiértékeléséhez $0,306 \pm 0,006$ (SD) standard időegységre volt szükség, míg $n = 4$ esetén a megfelelő érték $0,829 \pm 0,001$ (SD). Eszerint, még a négydimenziós változat kiszámítása is valamivel kevesebb CPU-időt igényel, mint az S5 tesztfüggvényé. A (2) tesztfüggvény numerikus változatának értéke az $x_i \in (-1, 0 \cdot 10^{-13}, 1, 0 \cdot 10^{-13})$, $i = 1, 2, \dots, n$ intervallumban volt nulla. Ennek ellenére a számítógépes ábrázolás elég finom volt ahhoz, hogy több mint egymillió olyan helyi minimuma maradt, amelynek vonzáskörzete legalább száz numerikusan megkülönböztethető pontot tartalmaz.

Az A algoritmust tíz független futással teszteltük a (2) feladat egy- és négydimenziós változatával a (3) korlátokkal megadott tartományon. Az eljárás paramétereit úgy állítottuk be, hogy a globális minimumpont becslése minél közelebb legyen az origóhoz (így ezek a paraméterek eltértek a korábban használtaktól). A megbízhatósági tesztszempontról a helyi kereső választása közömbösnek bizonyult.

Az egydimenziós feladatra a legkisebb talált helyi minimum $0,523449 \cdot 10^{-52}$ volt a $0,193281 \cdot 10^{-8}$ pontban. Ez az előbb említett sorrendben a 164 687 623. helyi minimumpont, és vonzáskörzetének nagysága a képlet alapján $A_x = 0,23472 \cdot 10^{-16}$. A globális minimumra kapott legrosszabb becslés $0,319144 \cdot 10^{-23}$ -nak adódott a $-0,119009 \cdot 10^{-3}$ pontban; ez a 2673. helyi minimum és $A_x = 0,88989 \cdot 10^{-7}$. Az átlagos futás 33,5 standard időegységet és 22 137 függvényhívást igényelt.

A négydimenziós feladatra a globális minimum legjobb és legrosszabb becslése $0,272099 \cdot 10^{-8}$, illetve $0,598347 \cdot 10^{-6}$ volt. Az átlagos futás 46,1 standard időegységet és 22 020 függvényhívást használt.

Ezen eredmények alapján megállapíthatjuk, hogy a vizsgált globális optimalizálási módszer hangolható úgy, hogy alkalmas legyen a legtöbb gyakorlati feladat megbízható megoldására.

5. Numerikus tesztelés légzésmechanikai modell illesztési feladatokkal

Az előzőekben tárgyalt numerikus vizsgálatokon túl paraméterbecslési módszerünk használhatóságát világítja meg a gyakorlati feladaton végzett teszt is. A légzésmechanikai modellek illesztésére az

$$f(\mathbf{x}) = \sqrt{\frac{1}{m} \sum_{i=1}^m |z(\omega_i) - z_{mod}(\omega_i, \mathbf{x})|^2}, \quad (7)$$

célfüggvényt használtuk, ahol \mathbf{x} a modellparaméterek vektora, m az adatpontok száma, $z(\omega_i)$ az ω_i valós körfrekvencián mért komplex impedancia, és $z_{mod}(\omega_i, \mathbf{x}) : \mathbb{R}^{n+1} \rightarrow \mathbb{C}$ pedig a vizsgált modellfüggvény [5,18,19]. Könnyű megmutatni, hogy a (7)-ben szereplő komplex eltérések ellenére $f(\mathbf{x})$ felírható olyan tisztán valós függvényként, amely beilleszthető az (1) feladatba. Tesztünkben a talán leggyakrabban tanulmányozott légzésmechanikai modellt, a Mead-modellt alkalmaztuk (1. Model 1 a [2]-ben):

$$z_{mod}(s) = \frac{I_c R_2 C_1 C_2 s^3 + (I_c(C_1 + C_2) + R_c R_2 C_1 C_2) s^2 + (R_c(C_1 + C_2) + R_2 C_2) s + 1}{R_2 C_1 C_2 s^2 + (C_1 + C_2) s}, \quad (8)$$

ahol $s = i\omega$, i a képzetes egység, R_c , I_c , R_2 , C_1 , C_2 pedig pozitív valós modellparaméterek. Ez a feladat mért adatok felhasználásával gyakran vezetett több, lényegesen különböző helyi minimumhoz.

A jelen tesztben módszerünk pontosságát és hatékonyságát vizsgáltuk. A korábban említett UNIRANDI és kvázi-Newton helyi kereső eljárások mellett a kifejezetten a legkisebb négyzetes alakú célfüggvényekhez kifejlesztett Levenberg – Marquardt algoritmust is implementáltuk. A teszthez az adatokat az 5. Táblázatban megadott paraméterértékekkel számítottuk a (8) modellfüggvénnyel a 2,5, 3,0, ..., 10,5 Hz frekvenciákon, (azaz a szokásos alacsonyfrekvenciás mérések ω_i értékein). Mivel globális optimalizálási módszerünk sztochasztikus, ezért minden helyi keresővel hétszer futtattuk ugyanazon a feladaton. A globális minimumot minden esetben megtalálták az egyes eljárások. Az 5. és 6. Táblázatban lévő adatok ezen futások átlagaként értelmezendők.

Az 5. Táblázat az egyes helyi keresőkkel kiegészített globális optimalizálási algoritmus talált optimális paraméter-értékei átlagát és variációs koefficiensét (szórás

5. Táblázat. A három helyi kereső eljárás hatása a modellparaméterek reprodukálásának pontosságára

Helyi kereső		Paraméter				
		R_c	I_c	C_1	R_2	C_2
Eredeti értékek:		4,00	0,02	0,005	7,00	0,1
UNIRANDI	átlag	4,0335	0,01944	0,005082	6,9738	0,1105
	c.v.	0,0041	0,0144	0,0092	0,0019	0,0760
Levenberg-Marquardt	átlag	4,0195	0,01969	0,005055	6,9838	0,1044
	c.v.	0,0128	0,0381	0,0218	0,0044	0,1073
kvázi-Newton	átlag	4,0003	0,02000	0,005001	6,9998	0,1001
	c.v.	0,0003	0,0007	0,0004	0,0001	0,0023

osztva az átlaggal, jelölése c.v.) foglalja össze. Bár mindegyik módszer pontossága kielégítő, a kvázi-Newton eljárással kiegészített messze a legpontosabb. Meg kell jegyezni, hogy a kis c.v. érték csak akkor mutatja az illető módszer pontosságát, ha csak egyetlen globális minimumpont van, és az szeparált; míg ha a globális minimumpontok egy alteret alkotnak, akkor a kis c.v. érték azt jelzi, hogy a vizsgált algoritmus nem képes ezt a redundanciát tükrözni (vö. [2]). Az 5. Táblázatban a C_2 paraméterre közel egy nagyságrenddel nagyobb variációs koefficiensek adódtak, mint a többi paraméterekre. Ez arra utal, hogy a célfüggvény kevésbé érzékeny erre a paraméterre, ami összhangban van mérési adatokkal végzett hasonló vizsgálatokkal [2].

A 6. Táblázat a három eljárásra vonatkozó hatékonysági eredményeket foglalja össze. Ebben f^* jelöli a talált globális minimumot, CPU a megoldáshoz használt CPU-időt másodpercben, NFE pedig a szükséges függvényhívások számát. Az első három oszlopban szereplő értékek ugyanarra a hét futásra vonatkozó átlagértékek, mint amelyek az 5. Táblázat adatait szolgáltatatták. Az első három oszlop alapján a kvázi-Newton módszer a leghatékonyabb, de az első két módszert nehéz összevetni. Ezért az utolsó két oszlopban olyan mutatókat szerepeltetünk, amelyek jellemzik az illető eljárás hatékonyságát: a nagyobb érték nagyobb hatékonyságot jelez. Ezek a mutatók is megerősítik a kvázi-Newton módszer kiemelkedő voltát, valamint azt, hogy feladatunkon a Levenberg-Marquardt módszer valamivel jobb hatékonyságú, mint az UNIRANDI.

Ezek az eredmények némileg meglepőek, hiszen a feladat szerkezete alapján a Levenberg-Marquardt módszernek kellene a leghatékonyabbnak lennie. Valószínűleg a viszonylag kevés mintapontra való illesztés és a célfüggvény simasága miatt bizonyult a kvázi-Newton módszer hatékonnyabbnak. A teszteredmények összhangban vannak az ilyen modellillesztési feladatokra vonatkozó numerikus tapasztalatokkal (pl. [2,12,26,27]). Egy cikket találtunk [11], amelyben összehasonlításra alkalmas

6. Táblázat. A három paraméterbecslési eljárás hatékonysága

Helyi kereső	100 f^*	CPU	NFE	$\frac{1}{\text{CPU } f^*}$	$\frac{1}{\text{NFE } f^*}$
UNIRANDI	0,8100	646,7	10957	0,00191	0,000113
Levenberg-Marquardt	1,2676	209,0	3632	0,00377	0,000217
kvázi-Newton	0,0219	124,7	2324	0,36690	0,019693

adat van légzésmechanikai modellillesztési eljárás hatékonyságáról. Ott több mint 47 000 függvényhívásra volt szükség a (8) modell azonosításához 31 impedancia-adat alapján. Ezzel szemben globális optimalizálási módszerünk a kvázi-Newton eljárással ugyanilyen feladatot mért adatokon az esetek nagy többségében 10 000-nél kevesebb függvényhívással oldott meg, és tesztiünkben átlagosan csak 2324 függvényhívásra volt szükség. Eljárásunkat [5,6,18] publikálása óta számos neves kutatócsoport használja (pl. [12,14,22,26,27,38]).

Függelék: Standard globális optimalizálási tesztfeladatok

Ebben a függelékben a dolgozatban több helyen is említett standard globális optimalizálási tesztfeladatokat ismertetjük. Ezeket a szokásos nemlineáris optimalizálási tesztfeladatoktól az különbözteti meg, hogy a lehetséges megoldások halmazán rendszerint több helyi minimumpont van, és így egy helyi kereső eljárás nem feltétlenül tudja a globális minimum helyét meghatározni. A tesztfeladatok használata tipikus [4,6,8,10,24,32,33,37,38]; inkább kivételnek számít, ha egy globális optimalizálási módszer numerikus vizsgálatát más feladatokon végzik el [39].

A dolgozatban tárgyalt algoritmusokhoz jobban kötődő adatokat (a globális minimum értéke és helye) összefoglaltuk az 1. Táblázatban, itt az egyes feladatok leírását adjuk meg.

1. Shekel feladatok. A célfüggvény 4-változós:

$$f(x) = - \sum_{i=1}^m \frac{1}{(x - a^i)^T (x - a^i) + c^i}$$

ahol a^i valós vektor, c^i valós szám ($i = 1, \dots, m$), a lehetséges megoldások halmaza $0 \leq x_j \leq 10$ ($j = 1, 2, 3, 4$). A feladathoz tartozó adatok:

	i									
	1	2	3	4	5	6	7	8	9	10
a_1^i	4	1	8	6	3	2	5	8	6	7,0
a_2^i	4	1	8	6	7	9	5	1	2	3,6
a_3^i	4	1	8	6	3	2	5	8	6	7,0
a_4^i	4	1	8	6	7	9	5	1	2	3,6
c^i	0,1	0,2	0,2	0,4	0,4	0,6	0,3	0,7	0,5	0,5

Az $m = 5, 7, 10$ eseteket az S5, S7, S10 rövidítésekkel szokás jelölni. Ezek rendre 5, 7, illetve 10 helyi minimumponttal rendelkeznek, amelyek helyét közelítőleg az a^i vektorok adják meg, az egyes helyi minimumok értéke pedig $-1/c^i$ körül van. A feladatok jellemzője, hogy (különösen nagy m esetén) a globális minimum vonzáskörzete kicsi a lehetséges megoldások halmazához képest, ezért az egyes módszerek gyakran nem találják meg a globális minimumot.

2. Hartman feladatok. A célfüggvény alakja:

$$f(x) = - \sum_{i=1}^m c_i \exp\left(- \sum_{j=1}^n a_{i,j} (x_j - p_{i,j})^2\right)$$

ahol $p_{i,j}$, közelítőleg az i . helyi minimumpont, és c_i közelítőleg a megfelelő helyi minimum. A lehetséges megoldások halmaza $0 \leq x_i \leq 1$, ahol $i = 1, 2, \dots, n$. A feladathoz tartozó adatok $n = 3$ esetén:

i	$a_{i,j}$			c_i	$p_{i,j}$		
1	3,0	10,	30,	1,0	0,3689	0,1170	0,2673
2	0,1	10,	35,	1,2	0,4699	0,4387	0,7470
3	3,0	10,	30,	3,0	0,1091	0,8732	0,5547
4	0,1	10,	35,	3,2	0,03815	0,5743	0,8828

Az $n = 6$ esetre az adatok:

i	$a_{i,j}$						c_i
1	10,	3,0	17,	3,5	1,7	8,	1,0
2	0,05	10,	17,	0,1	8,0	14	1,2
3	3,0	3,5	1,7	10,	17,	8,	3,0
4	17,	8,0	0,05	10,	0,1	14	3,2

i	$P_{i,j}$					
1	0,1312	0,1696	0,5569	0,0124	0,8283	0,5886
2	0,2329	0,4135	0,8307	0,3736	0,1004	0,9991
3	0,2348	0,1451	0,3522	0,2883	0,3047	0,6650
4	0,4047	0,8828	0,8732	0,5743	0,1091	0,0381

A globális minimumpontok körül a célfüggvény rendkívül lapos, és ez nem kedvez az intervallum-aritmetikán alapuló módszereknek. A Hartman feladatok rövid jelölése H3, illetve H6.

3. Goldstein-Price feladat. A kétváltozós célfüggvény

$$f(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)],$$

és a lehetséges megoldások halmaza $-2 \leq x_i \leq 2$ ($i = 1, 2$). Ezen négy helyi minimumpont található. A célfüggvény bonyolultsága miatt a természetes intervallumkiterjesztésen alapuló befoglaló függvény általában több nagyságrenddel szélesebb, mint a megfelelő értékészlet. Jelölése GP.

4. Branin feladat. A kétváltozós célfüggvény alakja

$$f(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10 \left(1 - \frac{1}{8\pi}\right) \cos x_1 + 10,$$

a lehetséges megoldások halmaza $-5 \leq x_1 \leq 10$, $0 \leq x_2 \leq 15$. Ezen három globális minimumpont van. Jelölése RCOS.

5. Six-hump-camel-back feladat. A célfüggvény ismét kétváltozós:

$$f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4,$$

a $-5 \leq x_1 \leq 5$ lehetséges megoldási halmazzal. A függvény szimmetrikus az origóra, és három pár helyi minimumpontja van. Jelölése SHCB.

6. Rosenbrock feladat. Az

$$f(x) = 100(x_1^2 - x_2)^2 + (1 - x_1)^2$$

függvényt globális optimalizálási tesztfeladatként a $-1 \leq x_i \leq 5$ korlátokkal szokás használni. Rövidítése RB.

IRODALOM

1. ARCHETTI, F., G.P. SZEGŐ (1980): Global optimization algorithms. In: Nonlinear Optimization — Theory and Algorithms (L.C.W. Dixon, E. Spedicato, G.P. Szegő, eds.) Birkhäuser, Boston, 42^o-469.
2. AVANZOLINI, G., P. BARBINI (1982): Comment on "Estimating Respiratory Mechanical Parameters in Parallel Compartment Models". IEEE Trans. Biomed. Eng. 29, 772-774.
3. BAHVALOV, N.SZ. (1977): A gépi matematika numerikus módszerei. Műszaki Könyvkiadó, Budapest.
4. BOENDER, C.G.E., A.H.G. RINNOOY KAN, G.T. TIMMER, L. STOUGIE (1982): A stochastic method for global optimization. Math. Programming 22, 125-140.
5. CSENDES, T., B. DARÓCZY, Z. HANTOS (1986): Nonlinear parameter estimation by global optimization: comparison of local search methods in respiratory system modelling. In: System Modelling and Optimization (Lecture Notes in Control and Information Sciences, No. 84, A. Prékopa and B. Straziczky, eds.), Springer, Berlin, 188-192.
6. CSENDES, T. (1988): Nonlinear parameter estimation by global optimization — efficiency and reliability. Acta Cybernetica 8, 361-370.
7. CSENDES, T. (1989): An interval method for bounding level sets of parameter estimation problems. Computing 41, 75-86.
8. CSENDES, T. (1990): Interval method for bounding level sets: revisited and tested with global optimization problems. BIT 30, 650-657.
9. DARÓCZY, B., Z. HANTOS (1990): Generation of optimum pseudorandom signals for respiratory impedance measurements. Int. J. Biomed. Comp. 25, 21-31.
10. DIXON, L.C.W., G.P. SZEGŐ (eds.) (1978): Towards Global Optimisation 2. North-Holland, Amsterdam.
11. EYLES, J.G., R.L. PIMMEL (1981): Estimating respiratory mechanical parameters in parallel compartment models. IEEE Trans. Biomed. Eng. 28, 313-317.
12. FARRÉ, R., R. PESLIN, E. OOSTVEEN, B. SUKI, C. DUVIVIER, D. NAVAJAS (1989): Human respiratory impedance from 8 to 256 Hz corrected for upper airway shunt. J. Appl. Physiol. 67, 1973-1981.
13. GILL, P. E., W. MURRAY, M.H. WRIGHT (1981): Practical Optimization. Academic Press, London.
14. GROMA, G.I., F. RAKSI, G. SZABÓ, G. VARÓ (1988): Picosecond and nanosecond components in bacteriorhodopsin light-induced electric-response signal. Biophysics J. 54, 77-80.
15. HANSEN, E.R. (1980): Global optimisation using interval analysis — the multi-dimensional case. Numer. Math. 34, 247-270.

16. HANSEN, P., B. JAUMARD, S.H. LU (1989): An analytical approach to global optimization. RUTCOR Research Report 4-89, Rutgers University, New Brunswick.
17. HANTOS, Z., G. GALGÓCZY, B. DARÓCZY, T. CSENDES, J. KLEBNICZKI (1982): A lumped parameter model of the upper airway. *Bull. Eur. Physiopath. Resp.* 18, 48-49.
18. HANTOS, Z., B. DARÓCZY, B. SUKI, G. GALGÓCZY, T. CSENDES (1986): Forced oscillatory impedance of the respiratory system at low frequencies. *J. Appl. Physiol.* 60, 123-132.
19. HANTOS, Z., B. DARÓCZY, T. CSENDES, B. SUKI, S. NAGY (1990): Modelling of low-frequency pulmonary impedance in the dog. *J. Appl. Physiol.* 68, 849-860.
20. HORST, R. (1988): Deterministic global optimization with partition sets whose feasibility is not known: application to concave minimization, reverse convex constraints, DC-programming, and Lipschitzian optimization. *J. Optim. Theory Appl.* 58, 11-37.
21. JÄRVI, T. (1973): A random search optimizer with an application to a maximin problem. Publications of the Institute for Applied Mathematics, University of Turku, No. 3.
22. LUTCHEN, K.R., JACKSON, A.C. (1987): Reliability of parameter estimates from models applied to respiratory impedance data. *J. Appl. Physiol.* 62, 403-413.
23. MARQUARDT, D.W. (1963): An algorithm for least-squares estimation of nonlinear parameters. *SIAM J. Appl. Math.* 11, 431-441.
24. MOORE, R.E., H. RATSCHKE (1988): Inclusion functions and global optimization II. *Math. Programming* 41, 341-356.
25. MURTY, K.G., S.N. KABADI (1987): Some NP-complete problems in quadratic and nonlinear programming. *Math. Programming* 39, 117-130.
26. OOSTVEEN, E., R. PESLIN, C. GALLINA, A. ZWART (1989): Flow and volume dependence of respiratory mechanical properties studied by forced oscillation. *J. Appl. Physiol.* 67, 2212-2218.
27. PESLIN, R., C. DUVIVIER, B. SUKI, R. FARRE, E. OOSTVEEN, C. GALLINA (1990): Respiratory impedance to ambient pressure changes at low frequencies. *J. Appl. Physiol.* 68, 665-671.
28. PINTÉR, J. (1986): Extended univariate algorithms for n-dimensional global optimization. *Computing* 36, 91-103.
29. PINTÉR, J. (1988): Branch-and-bound algorithms for solving global optimization problems with Lipschitzian structure. *Optimization* 19, 101-110.
30. PRICE, W.L. (1978): A controlled random search procedure for global optimization. In: *Towards Global Optimisation 2.* (L.C.W. Dixon, G.P. Szegő, eds.), North-Holland, Amsterdam, 71-84.
31. RATSCHKE, H., J.G. ROKNE (1987): Efficiency of a global optimization algorithm. *SIAM J. Numer. Anal.* 24, 1191-1201.
32. RINNOOY KAN, A.H.G., G.T. TIMMER (1987): Stochastic global optimization methods. Part I: Clustering methods. *Math. Programming* 39, 27-56.
33. RINNOOY KAN, A.H.G., G.T. TIMMER (1987): Stochastic global optimization methods. Part II: Multi-level methods. *Math. Programming* 39, 57-78.

34. ROTGER, M., R. PESLIN, C. DUVIVIER, D. NAVAJAS, C. GALLINA (1988): Density dependence of respiratory input and transfer impedances in humans. *J. Appl. Physiol.* 65, 928-933.
35. SUKI, E., R. PESLIN, C. DUVIVIER, R. FARRÉ (1989): Lung impedance in healthy humans measured by forced oscillations from 0.01 to 0.1 Hz. *J. Appl. Physiol.* 67, 1623-1629.
36. SUKI, B., T. CSENDES, B. DARÓCZY (1990): Mechanical impedance of the canine diaphragm II. theoretical model and parameter estimation. *Med. & Biol. Eng. & Comp.* 28, 367-373.
37. TIMMER, G.T. (1984): Global optimization: a stochastic approach. Ph.D. Thesis, Erasmus University, Rotterdam.
38. TÖRN, A., A. ŽILINSKAS (1989): Global Optimization. (Lecture Notes in Computer Science No. 350, G. Goos and J. Hartmanis, Eds.) Springer, Berlin.
39. WALSTER, G.W., E.R. HANSEN, S. SENGUPTA (1985): Test results for a global optimization algorithm. In: Numerical optimization 1984 (P.T. Boggs, R.H. Byrd, R.B. Schnabel (ed.), SIAM, Philadelphia) pp. 272-287.

A CLUSTERING GLOBAL OPTIMIZATION METHOD FOR PARAMETER ESTIMATION PROBLEMS

In this paper we first show that the objective function of a least squares type nonlinear parameter estimation problem can be any non-negative real function, and therefore this class of problems corresponds to global optimization. Two non-derivative implementations of a global optimization method are presented; with nine standard test functions applied to measure their efficiency. A new nonlinear test problem is then presented for testing the reliability of global optimization algorithms. This test function has a countable infinity of local minima and only one global minimizer. The region of attraction of the global minimum is of zero measure. The results of efficiency and reliability tests are given, and three local search procedures are compared by means of a real life problem.

