



MISKOLCI EGYETEM
GÉPÉSZMÉRNÖKI ÉS INFORMATIKAI KAR



Analízis Tanszék

TUDOMÁNYOS DIÁKKÖRI DOLGOZAT

Megoldási módszerek az általánosított Sakiadis-feladatra

Csáti Zoltán

I. éves MSc gépészmérnök hallgató

Konzulens:

Vadászné dr. Bognár Gabriella

egyetemi docens

Gépészmérnöki és Informatikai Kar

Miskolc

2013

Tartalomjegyzék

1. Bevezetés	3
2. Matematikai modell	4
3. Iteratív transzformációs módszer	8
4. Véges differencia módszer	13
5. Chebyshev pszeudospektrál módszer	17
6. Összefoglalás, további célok	22
7. Függelék	23
8. Hivatkozások	28

1. Bevezetés

Prandtl 1904-ben ismertette a róla elnevezett határréteg elméletet [18], a folyadékmechanika ilyen irányú vizsgálatait ezt követően rohamos fejlődésnek indultak. Az elmélet szerint egy sűrűdésos folyadéktérben lévő test esetén elegendő a testet körülvevő – az egész térhez képest jóval kisebb térfogatú részben – vékony rétegben viszkózusként kezelni a közeget; attól bizonyos távolságra a folyadék jó közelítéssel ideális folyadéknak tekinthető. Abban az időben, amikor számítógépek hiányában analitikus megoldást kellett keresni az áramlási jellemzők meghatározására, ez az egyszerűsítés nagyon hasznos volt. A határréteg feltételezéseket azonban a mai napig alkalmazzák, mert így a folyadékmozgást leíró matematikai modellek egyszerűbbek. Ennek nagy jelentősége van a kohászati, petrokémiai és műanyagfeldolgozási alkalmazásokban. Egy nyugvó folyadékban mozgó test által okozott áramlás fontos szerepet játszik sok anyagfeldolgozó eljárásnál, ú.m. meleg hengerlés, fémmegmunkálás és folytonos öntés (lásd pl. [1], [13], [21]). Newtoni folyadékban állandó sebességgel mozgó síklap határréteg áramlását Sakiadis [20] vizsgálta analitikusan. Eredményeit Tsou és társai [24] kísérletileg igazolták. Sakiadis polimer lemezek folytonos extrudálását vizsgálta a düznitől a felcsévélő hengerig. A nyílás és a felcsévélő henger egymástól véges távolságra van elhelyezve [19], [20]. A vizsgálat során feltételezte, hogy a folyamat elindulása után bizonyos idő elteltével állandósult állapot alakul ki. Tsou, Sparrow és Goldstein [24] megmutatták mind analitikus, mind kísérleti eredményeket tartalmazó tanulmányaikban, hogy a newtoni folyadékokra analitikus megoldással nyert lamináris sebességmező nagyon jó egyezést mutat a mért adatokkal. Ez igazolja, hogy folyamatosan mozgó felületen kialakuló határrétegben a modell fizikailag megvalósuló áramlást ír le. Célunk, hogy nyugvó nem-newtoni folyadékban állandó sebességgel mozgó át nem eresztő síklap határréteg problémáját leíró egyenleteket oldjuk meg. Mivel zárt alakú egzakt megoldás az általunk vizsgált esetre nem adható meg, ezért numerikus módszereket alkalmazunk a megoldások meghatározására és elemzésére. Meghatározzuk a sebességeloszlást a határrétegben, a határréteg vastagságát és az ellenállástényezőt különböző hatványtörvény kitevőkre.

2. Matematikai modell

Vizsgálatunkban egy nem-newtoni folyadékban elhelyezett impermeábilis síklapot tekintünk. Az összenyomhatatlan, nyugvó folyadékban a síklap vízszintes irányban állandó sebességgel mozog. Célunk meghatározni a határrétegben a sebességeloszlást stacionárius áramlást feltételezve.

A határréteg elmélet feltételezései a következők [15]:

- síkbeli áramlással számolunk, azaz a sebességre fennáll $\vec{v}(x, y) = u(x, y)\vec{i} + v(x, y)\vec{j}$,
- a falra merőleges sebességkomponens lényegesen kisebb a fallal párhuzamos komponensnél: $v \ll u$,
- a sebességkomponensek falra merőleges változása lényegesen nagyobb a fallal párhuzamos változásnál: $\partial u / \partial x \ll \partial u / \partial y$, $\partial v / \partial x \ll \partial v / \partial y$,
- az áramlás stacionárius,
- a ρ sűrűség állandó, a közeg összenyomhatatlan,
- a térerősséggel nem számolunk,
- a nyomás gradienssel sem számolunk, így $\partial p / \partial x \approx 0$, $\partial p / \partial y \approx 0$.

A kontinuitási egyenlet ekkor síkbeli Descartes-féle koordináta-rendszerben a következőképpen írható:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad (2.1)$$

ahol u és v a vízszintes és függőleges irányú sebességkomponenst jelöli. A Navier-Stokes-egyenlet a határréteg feltételezések figyelembevételére után az alábbi módon áll elő:

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y}, \quad (2.2)$$

ahol τ_{yx} az x normálisú síkon y irányban ébredő nyírófeszültség. A dolgozatban a nem-newtoni folyadékokra az Ostwald-de Waele-féle hatványtörvényt alkalmazzuk, amelynél

$$\tau_{yx} = K \left| \frac{\partial u}{\partial y} \right|^{n-1} \frac{\partial u}{\partial y}, \quad (2.3)$$

ahol K a konzisztencia index, n pedig a hatványtörvény kitevője. Ha $n < 1$, a folyadék pszeudoplasztikus, ha $n > 1$, akkor dilatáló, ha $n = 1$, akkor newtoni. A (2.3) anyagegyenletet a (2.2) mozgásegyenletbe helyettesítve a

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = \frac{\partial}{\partial y} \left(\gamma \left| \frac{\partial u}{\partial y} \right|^{n-1} \frac{\partial u}{\partial y} \right), \quad (2.4)$$

egyenletet kapjuk, ahol $\gamma = K/\rho$. A (2.1)-(2.4) nemlineáris parciális differenciálegyenlet-rendszerhez tartozó peremfeltételeket a következő gondolatmenet segítségével adjuk meg. Ha felület menti csúszást nem feltételezünk, akkor a lap felszínével érintkező folyadékreszek sebessége megegyezik a lap mozgási sebességével. A lap nem áteresztő, így a lap felszínén nincs a $\vec{v}$

sebességvektornak függőleges irányú komponense. A síklaptól megfelelő távolságban kialakul a zavartalan áramlás, amely jelen esetben a nyugvó közeg. Így a három peremfeltétel:

$$u(x, 0) = U_w, \quad v(x, 0) = 0, \quad \lim_{y \rightarrow \infty} u(x, y) = 0. \quad (2.5)$$

A parciális differenciálegyenlet-rendszert közönséges harmadrendű differenciálegyenletté alakítjuk hasonlósági változók bevezetésével. A levezetés menete hasonló a [7]-ban megadott módszerhez, ugyanis a Blasius-feladat és a Sakiadis-feladat csak a peremfeltételekben tér el egymástól. Vezessük be a ψ áramfüggvényt $u = \partial\psi/\partial y$, $v = -\partial\psi/\partial x$ alapján. Ezt behelyettesítve a (2.4) egyenletbe, a következőt kapjuk:

$$\psi_y \psi_{yx} - \psi_x \psi_{yy} = \gamma \left(|\psi_{yy}|^{n-1} \psi_{yy} \right)_y, \quad (2.6)$$

ahol az alsó indexek az adott változó szerinti parciális deriválást jelölik. Látható, hogy az áramfüggvénnyel a folytonossági egyenlet automatikusan teljesül. Vezessük be az η hasonlósági változót és keressük az áramfüggvényt a következő alakban:

$$\eta = a \frac{y}{x^\beta},$$

$$\psi = bx^{-\alpha} f(\eta),$$

ahol a, b, α, β konstansok, melyeket az alábbiakban ismertetett módon választunk meg. Állítsuk elő a (2.6) egyenlethez a deriváltakat.

$$\begin{aligned} \psi_x &= -bx^{-\alpha-1}[\alpha f + \beta \eta f'] \\ \psi_y &= abx^{-\alpha-\beta} f' \\ \psi_{yy} &= a^2 bx^{-\alpha-2\beta} f'' \\ \psi_{yx} &= -abx^{-\alpha-\beta-1}[(\alpha + \beta)f' + \beta \eta f''], \end{aligned}$$

ahol f', f'' az f függvény η változó szerinti első illetve második deriváltját jelöli. Ezeket felhasználva a (2.6) differenciálegyenlet egyszerűsítés után az alábbi alakot ölti:

$$a^2 b^2 x^{-2\alpha-2\beta-1} [a f f'' - (\alpha + \beta) f'^2] = \gamma (a^2 b)^n x^{(-\alpha-2\beta)n} \frac{a}{x^\beta} [|f''|^{n-1} f'']'.$$

Ha mindkét oldalon az x kitevői megegyeznek, akkor a

$$(2n - 1)\beta + (n - 2)\alpha = 1 \quad (2.7)$$

egyenletet kapjuk, ha pedig a konstans szorzók egyenlőségét tekintjük, akkor a

$$a^2 b^2 = \gamma (a^2 b)^n a \quad (2.8)$$

egyenlet nyerhető. A peremfeltételeket a ψ áramfüggvényre felírva további kapcsolatot létesíthetünk a paraméterek között.

A (2.5) harmadik peremfeltételében ha $y \rightarrow \infty$, akkor $\eta \rightarrow \infty$:

$$\lim_{y \rightarrow \infty} \psi_y(x, y) = \lim_{\eta \rightarrow \infty} f'(\eta) = abx^{-\alpha-\beta} f'(\eta) = 0.$$

Válasszuk $f'(\infty) = \lim_{\eta \rightarrow \infty} f'(\eta) = 0$ -nak, továbbá legyen

$$\alpha + \beta = 0. \quad (2.9)$$

Innen következik, hogy

$$ab = U_w. \quad (2.10)$$

Azonban a (2.7)-ből és (2.9)-ből

$$\beta = -\alpha = \frac{1}{n+1}.$$

A (2.5) második peremfeltételének új alakja:

$$-\psi_x(x, 0) = \frac{b}{n+1} x^{-\frac{n}{n+1}} [\eta f'(\eta) - f(\eta)] = 0.$$

A fenti egyenlet $\eta = 0$ esetén, tetszőleges x -re fennáll, ha

$$f(0) = 0.$$

Végül a (2.5) első peremfeltételét átírva kapjuk meg a mozgásegyenlet hasonlósági alakjához tartozó harmadik peremfeltételt:

$$\psi_y(x, 0) = abx^{-\alpha-\beta} f'(0) = -U_w.$$

Felhasználva a (2.9)-at és (2.10)-et

$$f'(0) = 1$$

adódik. A fentiek alapján összefoglaljuk a paraméterek kiszámolt értékeit:

$$\begin{aligned} \alpha &= -\frac{1}{n+1}, & \beta &= \frac{1}{n+1}, \\ a &= \gamma^{-\frac{1}{n+1}} U_w^{\frac{2-n}{n+1}}, & b &= \gamma^{\frac{1}{n+1}} U_w^{\frac{2n-1}{n+1}}. \end{aligned}$$

Az eredeti sebességösszetevők és a dimenziómentes η és f hasonlósági változók között az összefüggés:

$$u(x, y) = U_w f'(\eta) \quad (2.11)$$

$$v(x, y) = \frac{U_w}{n+1} Re_x^{-\frac{1}{n+1}} (\eta f'(\eta) - f(\eta)), \quad (2.12)$$

$$\eta = Re_x^{\frac{1}{n+1}} \frac{y}{x}, \quad Re_x = \frac{U_w^{2-n} x^n}{\gamma}, \quad (2.13)$$

ahol Re_x a lokális Reynolds-szám. A hasonlósági megoldással kapott peremérték-feladat:

$$\left(|f''|^{n-1} f''\right)' + \frac{1}{n+1} f f'' = 0 \quad (2.14)$$

$$f(0) = 0, \quad f'(0) = 1, \quad \lim_{\eta \rightarrow \infty} f'(\eta) = 0 \quad (2.15)$$

A megoldást az $f \in C^3[0, \infty)$ függvények halmazán keressük. Célunk meghatározni a (2.14)-(2.15) peremérték-feladat megoldását, ugyanis ekkor megkapjuk a megoldást az eredeti sebességmezőre is (2.11)-(2.12) alapján, továbbá meghatározni a határréteg vastagságot és $f''(0)$ -t, amely a

$$C_D = (n+1)^{\frac{1}{n+1}} Re^{-\frac{n}{n+1}} |f''(0)|^{n-1} f''(0)$$

ellenállás-tényező és a

$$\tau_w(x) = \left[\frac{\varrho^n K U_w^{3n}}{x^n} \right] |f''(0)|^{n-1} f''(0)$$

fal menti nyírófeszültség kiszámításához szükséges.

A (2.14)-(2.15) peremértékfeladat megoldásának számos nehézsége van. A differenciálegyenlet erősen nemlineáris, az értelmezési tartomány a $[0, \infty)$ félig végtelen intervallum, így zárt alakban analitikus megoldás nem adható meg. A dolgozatban három különböző numerikus módszert mutatunk be a feladat megoldására. A (2.14)-(2.15) peremértékfeladatot newtoni folyadéokra ($n = 1$) többen vizsgálták. Bataller [3] például belövéses módszerrel jutott eredményre. Ishak [14] véges differencia módszerrel határozta meg a közelítő megoldást és Fazio [11] az iteratív transzformációs módszert alkalmazta. Az 1. táblázat összefoglalja a különböző szerzők által kapott eredményeket.

Szerzők	$-f''(0)$
Sakiadis [20]	0,44375
Tsou és társai [24]	0,444
Takhar és társai [22]	0,4439
Pop és társai [17]	0,4445517
Pantokratoras [16]	0,4438
Andersson [2], $\eta_\infty = 32$ -re	0,4437483
Bataller [3]	0,44374733
Ishak [14]	0,4438
Fazio [11]	0.443806

1. táblázat. $f''(0)$ értékei $n = 1$ esetén

3. Iteratív transzformációs módszer

A nem-iteratív és iteratív transzformációs módszert – melynek alapja a skálázási elv – Fazio vezette be [9]. Ezzel a módszerrel egy peremérték feladat kezdetiérték feladattá transzformálható, ami az esetünkben azért különösen előnyös, mert így a végtelenben előírt peremfeltétel helyett kezdeti feltételt kell figyelembe vennünk. A transzformáció során többféle csoportot alkalmazhatunk, például translációs, spirál, vagy skálázási csoportot. Ebben a feladatban a skálázási csoporttal jutunk eredményre. Vegyük az η független változót, és az $f(\eta)$ függő változót. Vezessünk be az

$$g = \lambda^\alpha f, \quad \eta^* = \lambda^\beta \eta \quad (3.1)$$

skálázási transzformációt, ahol λ a csoportparaméter, az α, β kitevők pedig konstansok, melyeket úgy határozunk meg, hogy teljesüljön az egyenletre és a peremfeltételekre az invariancia. Az invariancia azt jelenti, hogy ha $f(\eta)$ a vizsgált differenciálegyenlet megoldása, akkor $g(\eta^*) = \lambda^\alpha f(\lambda^\beta \eta)$ is megoldása a differenciálegyenletnek bármely λ paraméter esetén. Nem iteratív transzformációs módszer nem alkalmazható még newtoni folyadék esetén sem, mert bár a differenciálegyenlet és a (2.15) első peremfeltétele invariánssá tehető, a másik két peremfeltétel nem. Vegyünk egy h paraméterrel bővített skálázási csoportot a (3.1)-nek megfelelően és a bővített csoporttól követeljük meg az invarianciát, azaz

$$g = \lambda^\alpha f, \quad \eta^* = \lambda^\beta \eta, \quad h^* = \lambda^\gamma h. \quad (3.2)$$

Tehát, ha $f(\eta; h)$ megoldás, akkor $g(\eta^*) = \lambda^\alpha f(\lambda^\beta \eta; \lambda^\gamma h)$ is megoldás. Mivel a feladat fizikai tartalmából következik, hogy τ_w -nek negatívnak kell lennie, ezért $f''(0) < 0$. Ennek figyelembevételével a (2.14) differenciálegyenletben elvégezve a deriválást:

$$n |f''|^{n-1} f''' + \frac{1}{n+1} f f'' = 0. \quad (3.3)$$

Bevezetjük a (3.2)-ben általánosan felírt skálázási csoportot, f^* helyett g -t írva:

$$g = \lambda^\alpha f, \quad \eta^* = \lambda^\beta \eta, \quad h^* = \lambda^\gamma h. \quad (3.4)$$

Állítsuk elő a deriváltakat:

$$\begin{aligned} f'(\eta) &= \frac{df}{d\eta} \frac{d\eta^*}{d\eta} = \lambda^{\beta-\alpha} g'(\eta^*) \\ f''(\eta) &= \frac{d}{d\eta} f'(\eta) = \lambda^{2\beta-\alpha} g'' \\ f'''(\eta) &= \lambda^{3\beta-\alpha} g''', \end{aligned}$$

ahol f' az η , g' az η^* szerinti deriváltat jelöli. Behelyettesítve ezeket a (3.3)-be, rendezés után a következő egyenletet kapjuk:

$$n \lambda^{(n-1)(2\beta-\alpha)+3\beta-\alpha} |g''|^{n-1} g''' + \frac{1}{n+1} \lambda^{2\beta-2\alpha} g g'',$$

mely akkor invariáns, ha a kitevőkre teljesül az alábbi egyenlet:

$$n(2\beta - \alpha) = \beta - 2\alpha. \quad (3.5)$$

A (2.15) első peremfeltétele minden esetben invariáns, ugyanis

$$f(0) = \lambda^{-\alpha} g(0) = 0$$

csak $g(0) = 0$ esetén teljesül. Viszont ha a (2.15) második feltételét nézzük, akkor

$$f'(0) = \lambda^{\beta-\alpha} g'(0) = 1$$

egyenletből nem következik általános esetben $g'(0) = 1$. Módosítsuk a (2.15) peremfeltételeit az alábbi módon:

$$f(0) = 0, \quad f'(0) = h^p, \quad \lim_{\eta \rightarrow \infty} f'(\eta) = 1 - h^p, \quad (3.6)$$

ahol p egy szabályozó paraméter, melynek a későbbiekben a numerikus megoldás során lesz szerepe. Most a (3.6) második egyenletéből következik, hogy

$$\lambda^{\beta-\alpha} g'(0) = (\lambda^{-\gamma} h^*)^p,$$

azaz

$$\lambda^{\beta-\alpha+p\gamma} g'(0) = h^{*p},$$

így az invariancia feltétele:

$$\beta - \alpha + p\gamma = 0. \quad (3.7)$$

Álljon fenn a keresett $f''(0)$ -ra, hogy $f''(0) = -\sigma$, ahol $\sigma > 0$ és legyen a hiányzó kezdeti érték $g''(0) = -1$. Ez utóbbiból azt kapjuk, hogy

$$\lambda^{\alpha-2\beta} f''(0) = -1.$$

Legyen $\lambda = \sigma$, ekkor az invariancia akkor teljesül, ha

$$\alpha - 2\beta + 1 = 0. \quad (3.8)$$

A (3.5), (3.7) és (3.8) lineáris egyenletek megoldásával megkapjuk az α , β és γ paraméterek értékeit

$$\alpha = \frac{1-2n}{3}, \quad \beta = \frac{2-n}{3}, \quad \gamma = -\frac{1}{p} \frac{1+n}{3}. \quad (3.9)$$

Fontos megemlíteni, hogy mivel megköveteljük, hogy $f''(\eta) < 0$ legyen az értelmezési tartomány minden pontjában, ezért $f'(\eta)$ szigorúan monoton csökkenő függvény és nullához tart. A számítás során ki fog derülni, hogy ha σ -t nemnegatívnak vesszük, akkor az eljárás nem működik. Ugyanez a helyzet $n = 1$ esetén is [11]. Szükséges még meghatároznunk a λ csoportparaméter értékét, melyet a (3.6) harmadik peremfeltételéből számíthatunk ki:

$$\lambda^{\beta-\alpha} g'(\infty) = 1 - (\lambda^{-\gamma} h^*)^p.$$

Átrendezés után, $\lambda = \sigma$ figyelembevételével

$$\sigma = (g'(\infty) + h^{*p})^{-\frac{3}{1+n}} \quad (3.10)$$

adódik. Tehát a $g(\eta^*)$ függvényre vonatkozó kezdetiérték-feladat:

$$\left(|g''|^{n-1} g''\right)' + \frac{1}{n+1} g g'' = 0 \quad (3.11)$$

$$g(0) = 0, \quad g'(0) = h^{*p}, \quad g''(0) = -1. \quad (3.12)$$

A (3.11)-(3.12) kezdetiérték-feladatnak egy megoldása van $0 < n < 2$ esetén. A megoldás menete a következő:

1. **input:** $h^{*(0)}, n, p, tol$
2. (3.11)-(3.12) megoldása egy kezdeti $(h^*)^{(0)}$ -val
3. $\Gamma(h^*) = \left(g'(\infty) + h^{*(0)p}\right)^{-\frac{1+n}{3p}} h^{*(0)} - 1$
4. (a) **if** $|\Gamma(h^*)| < tol$
 $g(\eta^*) \rightarrow f(\eta)$
return
- (b) **else**
while $|\Gamma(h^*)| > tol$
 (3.11)-(3.12) megoldása
 $\Gamma(h^*) = (g'(\infty) + h^{*p})^{-\frac{1+n}{3p}} h^* - 1$
 $h^* = \text{gyökkereső_algoritmus}(h^*)$
end
 $g(\eta^*) \rightarrow f(\eta)$
end

A $\Gamma(h^*) \rightarrow 0$ azt fejezi ki, hogy $h^{*p} \rightarrow 1$; azaz ekkor megkapjuk a (2.14)-(2.15) peremérték-feladathoz tartozó megoldást.

A (3.11)-(3.12) kezdetiérték feladat numerikus megoldását néhány n paraméterre a MATLAB R2011a verziójának beépített Adams-Bashforth-Moulton megoldójával végeztük el a pontossági és adaptivitási paramétereket alapértelmezetten hagyva. Az $\eta_{\max}$ határréteg vastagsággal arányos szám a definíció szerint könnyen meghatározható.¹ A $\{\Gamma(h^*)\}$ gyöksorozat egy nem-lineáris egyenletet határoz meg, amelynek a gyökét intervallumfelezéssel állítottuk elő. Azért használtuk ezt a csupán első rendben konvergens módszert, mert egyrészt könnyű programozni, másrészt – feltéve, hogy a $\Gamma(h^*)$ a h^* folytonos függvénye – a képzett intervallumsorozat mindig ráhúzódik a keresett gyökre. A leállási feltétel $|\Gamma(h_a^*) - \Gamma(h_b^*)| < 10^{-4}$ volt, ahol h_a^* és h_b^* az adott iterációban a h^* -hoz tartozó intervallum alsó illetve felső határa. A 2. táblázatban összefoglaltuk a különböző n -ekre számolt $f''(0)$ értékeket.

¹A határréteg vastagság a fal és a határréteg azon pontja közötti távolság, ahol sebesség eléri az állandósult sebesség nagyságának 99%-át.

n	0.96	0.97	0.98	0.99	1	1.1	1.2
$f''(0)$	-0.443438	-0.443438	-0.443286	-0.443193	-0.443726	-0.436611	-0.409834

n	1.3	1.4	1.42	1.45	1.46	1.47	1.48	1.49
$f''(0)$	-0.361843	-0.282487	-0.259810	-0.217167	-0.199204	-0.178086	-0.151836	-0.115155

2. táblázat. $f''(0)$ értékei különböző n esetén

Számításaink $n = 1$ esetben Fazio eredményeivel egyeznek meg [11].

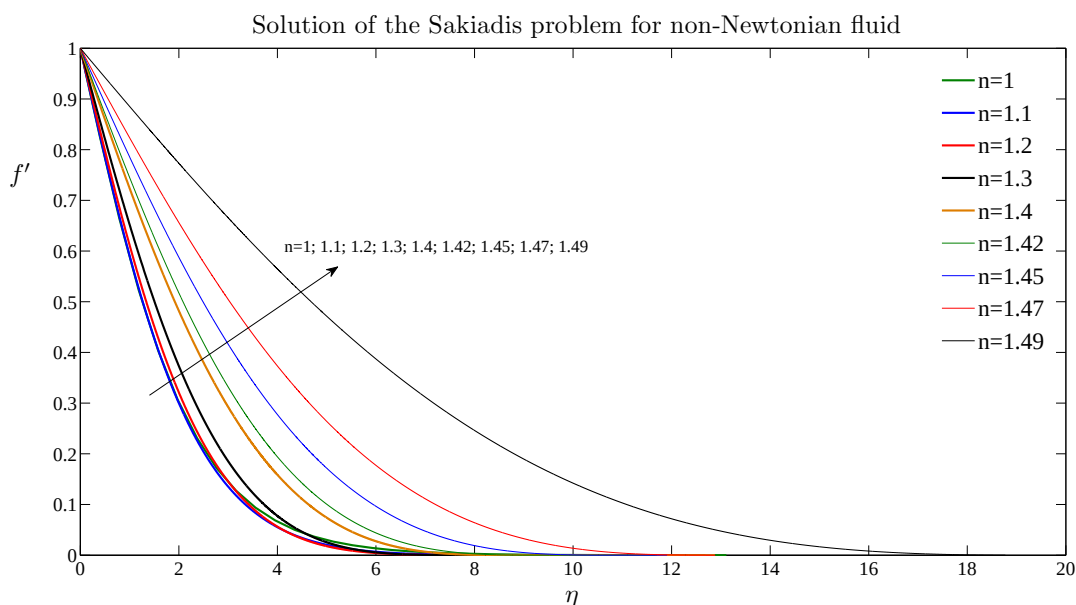
Az iteratív transzformációs módszer előnyei:

- Nem kell kezelni a végtelent.
- Kezdetiérték-feladatok megoldására stabil módszerek állnak rendelkezésre.
- A kezdetiérték feladat és a peremérték-feladat között kölcsönösen egyértelmű megfeleltetés van, így ahány megoldása van a kezdetiérték-feladatnak, pontosan annyi megoldása létezik a peremérték-feladatnak is [10], ezért ez egy alkalmas módszer peremérték-feladatok egzisztenciájának és unicitásának vizsgálatára.

Az iteratív transzformációs módszer hátránya:

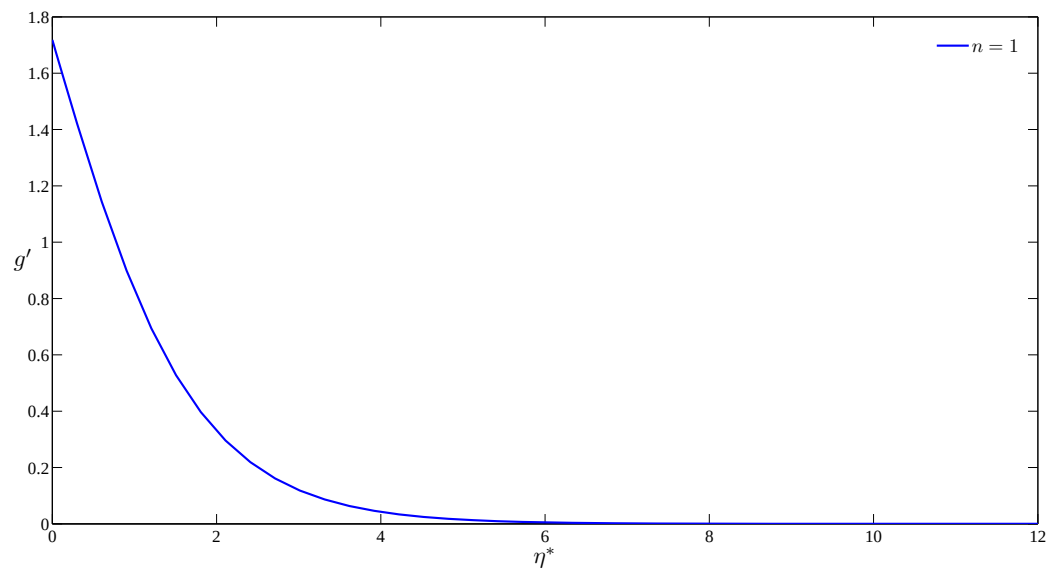
- Többször meg kell oldanunk kezdetiérték-feladatot, ezáltal a számításigény nagyobb.
- $n < 0,96$ -ra nem kaptunk megoldást intervallumfelezéssel.

Az 1. ábra az $f'(\eta)$ függvényeket mutatja különböző n értékeknél, amely a (2.11) szerint arányos az u sebességeloszlással. A 2. ábra $n = 1$ -re mutatja a $g'(\eta^*)$ függvényt.



1. ábra. Az $f'(\eta)$ függvény különböző n hatványtörvény kitevőkre

Látható, hogy n növelésével nő a határréteg vastagság. A módszer nem működik az $n < 1$ esetekre. Ha $n > 1,49$, akkor $f''(0)$ annyira közel esik 0-hoz, hogy az eljárás nem megbízható.



2. ábra. Az $f'(\eta)$ függvény $n = 1$ esetén

Az iteratív transzformációs módszerrel a megoldás hibájára nem tudtunk becslést adni, ezért felmerül a kérdés, hogy ezek a megoldások valóban kielégítőek-e.

4. Véges differencia módszer

Ebben a fejezetben véges differencia módszerrel határozzuk meg a (2.14)-(2.15) feladat közelítő megoldását. Megoldási nehézség, hogy az $f(\eta)$ függvény egy végtelen intervallumon van értelmezve. Ezt kétféleképpen szokás kezelni: a $[0, \infty)$ intervallumot csonkítjuk egy megfelelően nagy $[0, L]$ intervallumra és ott számítjuk ki a peremérték-feladat közelítő megoldását vagy pedig az eredeti intervallumon dolgozunk. Az első esetben figyelembe kell venni a csonkításból származó hibát, amit a gyakorlatban úgy oldanak meg, hogy addig növelik az L értéket, amíg az adott L -hez tartozó megoldásfüggvénynek az előző L -nél kapott megoldáshoz képesti eltérése kisebb egy meghatározott értéknél. A véges differencia módszer alkalmassá válhat a $[0, \infty)$ -on való közvetlen számításokra, ha kvázi-egyenközű rácsot alkalmazunk. Ilyenkor az utolsó csomópontot a végtelenbe helyezzük és így a végtelenben előírt peremfeltétel közvetlenül teljesíthető. A numerikus módszer sémáját Fazio [12] cikkében ismertetett módon építjük fel. A továbbiakban az η változó helyett az x -et vezetjük be, a MATLAB-ben írt programok könnyebb áttekinthetősége miatt. Minden fejezetben egymástól független jelölést használunk.

Osszuk fel a $\xi \in [0, 1]$ intervallumot N egyenlő részre: $\xi_0 = 0$ és $\xi_{n+1} = \xi_n + h$ ($n = 0 \dots N-1$), ahol $h = 1/N$. A kvázi-egyenközű felosztáshoz vezessük be az $x(\xi)$ függvényt az alábbi alakban:

$$x = -c \ln(1 - \xi), \quad (4.1)$$

ahol $c > 0$ a leképezést szabályozó paraméter. Ha a nem egész számú csomópontokat az

$$x_{n+\alpha} = x \left(\xi = \frac{n + \alpha}{N} \right),$$

($n = 0 \dots N-1$), $0 < \alpha < 1$ módon definiáljuk, akkor az utolsó $[x_{N-1}, x_N]$ intervallum végtelen, de az $x_{N-\frac{1}{2}}$ pont véges és lehetőségünk van a végtelen intervallumot véges sok intervallum segítségével leírni. Az egyszerűbb programozási jelölés érdekében ebben a fejezetben alkalmazzuk az $\eta = x$, $f(\eta) = u(x)$ jelöléseket.

Legyen $u_n = u(x_n)$, $u_{n+1} = u(x_{n+1})$, stb. A rész-intervallumok felezőpontjában:

$$u_{n+\frac{1}{2}} \approx \frac{x_{n+1} - x_{n+\frac{1}{2}}}{x_{n+1} - x_n} u_n + \frac{x_{n+\frac{1}{2}} - x_n}{x_{n+1} - x_n} u_{n+1}. \quad (4.2)$$

A differenciálhányados közelítésére a [12] cikkben megadott formulát alkalmazzuk a nem egész számú csomópontokban:

$$\left. \frac{du}{dx} \right|_{n+\frac{1}{2}} \approx \frac{u_{n+1} - u_n}{2 \left(x_{n+\frac{3}{4}} - x_{n+\frac{1}{4}} \right)}, \quad (4.3)$$

ugyanis

$$\left. \frac{du}{dx} \right|_{n+\frac{1}{2}} = \left. \frac{du}{d\xi} \right|_{n+\frac{1}{2}} \left. \frac{d\xi}{dx} \right|_{n+\frac{1}{2}} \approx \frac{u_{n+1} - u_n}{\xi_{n+1} - \xi_n} \frac{2 \left(\xi_{n+\frac{3}{4}} - \xi_{n+\frac{1}{4}} \right)}{2 \left(x_{n+\frac{3}{4}} - x_{n+\frac{1}{4}} \right)}$$

és $2 \left(\xi_{n+\frac{3}{4}} - \xi_{n+\frac{1}{4}} \right)$ mert ξ -ben egyenközű a rács. Mind a (4.2), mind a (4.3) formulák $\mathcal{O}(h^2)$ rendű közelítések. Először egy tetszőleges d -edrendű differenciálegyenlet peremérték-feladatának

diszkretizált alakját írjuk fel a fenti közelítések alkalmazásával:

$$\begin{aligned}\frac{d\mathbf{u}}{dx} &= \mathbf{F}(x, \mathbf{u}), \quad x \in [0, \infty), \\ \mathbf{G}(\mathbf{u}(0), \mathbf{u}(\infty)) &= \mathbf{0},\end{aligned}\tag{4.4}$$

ahol $\mathbf{u}(x) \in \mathbb{R}^d$ vektor-skalár függvény, $\mathbf{F} : [0, \infty) \times \mathbb{R}^d \rightarrow \mathbb{R}^d$ és $\mathbf{G} : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^d$ függvények. Az $\mathbf{U}_n = (u_1(x_n), \dots, u_d(x_n))$, $n = 0 \dots N$ jelölés bevezetése után a (4.4) diszkretizált alakja a (4.2) és (4.3) felhasználásával az alábbi differenciaegyenletre vezet:

$$\begin{aligned}\mathbf{U}_{n+1} - \mathbf{U}_n - a_{n+\frac{1}{2}}\mathbf{F}(x_{n+\frac{1}{2}}, b_{n+\frac{1}{2}}\mathbf{U}_{n+1} + c_{n+\frac{1}{2}}\mathbf{U}_n) &= \mathbf{0} \quad (n = 0, \dots, N-1), \\ \mathbf{G}(\mathbf{U}_0, \mathbf{U}_N) &= \mathbf{0},\end{aligned}\tag{4.5}$$

ahol

$$a_{n+\frac{1}{2}} = 2 \left(x_{n+\frac{3}{4}} - x_{n+\frac{1}{4}} \right), \tag{4.6}$$

$$b_{n+\frac{1}{2}} = \frac{x_{n+\frac{1}{2}} - x_n}{x_{n+1} - x_n}, \tag{4.7}$$

$$c_{n+\frac{1}{2}} = \frac{x_{n+1} - x_{n+\frac{1}{2}}}{x_{n+1} - x_n}, \tag{4.8}$$

$n = 0, \dots, N-1$. A (4.5) egyenletrendszer $d(N+1)$ ismeretlent és ugyanennyi skalár egyenletet tartalmaz, ahol d a (4.4) differenciálegyenlet rendje. Fazio a $b_{N-\frac{1}{2}} = 0$ és a $c_{N-\frac{1}{2}} = 1$ helyett a $b_{N-\frac{1}{2}} = b_{N-\frac{3}{2}}$ és $c_{N-\frac{1}{2}} = c_{N-\frac{3}{2}}$ összefüggéseket használta, azért hogy az együtthatókban ne legyen ugrásszerű növekedés. A leképezéshez a QuasiUniform (lásd 7.2.2) elnevezésű MATLAB függvényt írtuk, amely a többi programmal együtt megtalálható a függelékben.

Alkalmazzuk a leírtakat a (2.14)-(2.15) feladatra. Mivel $f''(\eta) < 0$, $\forall \eta \in [0, \infty)$, ezért (2.14) egyenértékű az

$$f''' = \frac{1}{n(n+1)} f |f''|^{2-n} \tag{4.9}$$

egyenlettel ($f = f(x)$). A (4.9) a hozzá tartozó (2.15) peremfeltételekkel felírható három elsőrendű differenciálegyenletből álló egyenletrendszerrel

$$\begin{aligned}f' &= g, \\ g' &= h, \\ h' &= \frac{1}{n(n+1)} f |h|^{2-n}\end{aligned}\tag{4.10}$$

és feltételekkel:

$$\begin{aligned}f(0) &= 0, \\ g(0) &= 1, \\ g(\infty) &= 0.\end{aligned}$$

Az előzőek alapján az ismeretlen függvények és a (4.10) differenciálegyenlet-rendszer jobb

oldala, továbbá a peremfeltételek vektorikus alakja:

$$\begin{aligned}\mathbf{u} &= (f, g, h), \\ \mathbf{F}(x, \mathbf{u}) &= \left(g, h, \frac{1}{n(n+1)} f |h|^{2-n} \right), \\ \mathbf{G}(\mathbf{u}(0), \mathbf{u}(\infty)) &= (f(0), g(0) - 1, g(\infty)),\end{aligned}$$

ahol $\mathbf{u}(x) \in \mathbb{R}^3$, $\mathbf{F} : [0, \infty) \times \mathbb{R}^3 \rightarrow \mathbb{R}^3$ és $\mathbf{G} : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}^3$. A nemlineáris egyenletrendszer programkóddá alakítása érdekében, a már korábban ismertett $\mathbf{U}_n = \mathbf{u}(x_n)$ -et úgy írjuk fel, hogy az $\mathbf{u}$ komponenseit indexeljük: $\mathbf{U}_n = [U_n(1) \ U_n(2) \ U_n(3)]^T$, ($n = 0, \dots, N$). Hasonlóan, legyen $\mathbf{F} = [\mathbf{F}_1 \ \mathbf{F}_2 \ \mathbf{F}_3]^T$, $\mathbf{G} = [\mathbf{G}_1 \ \mathbf{G}_2 \ \mathbf{G}_3]^T$. Ha az egyenletrendszer összes egyenletét azonos betűkkel akarjuk jelölni, akkor az indexeknek fel kell vennie az $1, 2, \dots, 3N + 3$ értékeket.² Mindezeket figyelembe véve az $\mathbf{f} = \mathbf{0}$ egyenletrendszert kell megoldanunk, részletesen kiírva:

$$\begin{aligned}f_1 &= U_2 - U_1 - a_1(b_1 U_{N+3} + c_1 U_{N+2}) \\ &\vdots \\ f_N &= U_{N+1} - U_N - a_N(b_N U_{2N+2} + c_N U_{2N+3}) \\ f_{N+1} &= U_1 \\ f_{N+2} &= U_{N+3} - U_{N+2} - a_1(b_1 U_{2N+4} + c_1 U_{2N+3}) \\ &\vdots \\ f_{2N+1} &= U_{2N+2} - U_{2N+1} - a_N(b_N U_{3N+3} + c_N U_{3N+2}) \\ f_{2N+2} &= U_{N+2} - 1 \\ f_{2N+3} &= U_{2N+4} - U_{2N+3} - a_1 \frac{1}{n(n+1)} (b_1 U_2 + c_1 U_1) |b_1 U_{2N+4} + c_1 U_{2N+3}|^{2-n} \\ &\vdots \\ f_{3N+2} &= U_{3N+3} - U_{3N+2} - a_N \frac{1}{n(n+1)} (b_N U_{N+1} + c_N U_N) |b_N U_{3N+3} + c_N U_{3N+2}|^{2-n} \\ f_{3N+3} &= U_{2N+2}\end{aligned} \tag{4.11}$$

és az $f_{N+1} = 0$, $f_{2N+2} = 0$ és $f_{3N+3} = 0$ egyenletek tartalmazzák a peremfeltételeket. Megoldandó a (4.11) nemlineáris algebrai egyenletrendszer az U_i ($i = 1, \dots, 3N + 3$) változókra. A megoldás során a MATLAB *Levenberg-Marquardt* és *trust-region dogleg* algoritmusait használtuk felváltva, melyekhez nem szükséges a Jacobi-mátrix megadása, mivel minden iterációban az eljárás becslő centrális differenciák segítségével. Mi azonban megadtuk a Jacobi-mátrixot, mert egyrészt így jóval gyorsabbá vált a számítás, másrészt nem minden esetben számítja ki a deriváltakat megfelelő pontossággal. A Jacobi-mátrix a

$$J = \left[\frac{\partial f_i}{\partial U_j} \right]_{i,j=1}^{3N+3}$$

értelmezése alapján számítható. A Jacobi-mátrix blokkos szerkezetű, egyes blokkok nulla

²Az indexeket eggyel eltoltuk, mert a MATLAB-ben 1-től kezdődik a tömbök indexelése.

mátrixokat, mások sávmátrixokat tartalmaznak. Az egyenletrendszert és a Jacobi-mátrixot a `fun4sakiadis` (lásd a 7.2.1 függelékben) függvény tartalmazza. A fentebb említett algoritmusokat az $U^{(0)} = [0 \ 1 \dots 3N + 3]^T$ kezdővektorból indítottuk. Általános tapasztalat az volt, hogy a megoldók legtöbb esetben "elakadtak". Ilyenkor újraindítottuk az iterációt kezdővektornak az előző futás során kapott eredményt beállítva. Nem volt minden esetben jó vagy az egyik, vagy a másik megoldó; amikor az egyik már nem hozott eredményt, akkor a másikkal folytattuk. A megoldás helyes jellege – a program által mért norma megfelelően kicsiny volta mellett – onnan látszott, hogy f' egy állandóhoz, f'' 0-hoz tartott. Nem-newtoni közeg esetében úgy jártunk el, hogy az $n = 1$ -hez tartozó megoldást állítottuk be kezdőfüggvénynek, és n fokozatos csökkentésével vagy növelésével jutottunk el a kívánt n -hez tartozó megoldáshoz. Egy adott anyagjellemzőnél két paramétert lehet állítani: a c szabályozó paramétert és az N intervallumszámot. A c értékét legalább akkorára kell megválasztani, hogy az utolsó előtti néhány csomópontban felvett függvényérték közel azonos legyen. Azt tapasztaltuk, hogy $N = 80$ felosztás már jó eredményt adott, c -re pedig akkora értéket kellett megadnunk, ami nem sokkal nagyobb, mint a minimálisan szükséges c érték. Ez utóbbira onnan következtettünk, hogy $n = 1$ -re rendelkezésre áll az $f''(0)$ közelítő értéke, ami [3] szerint $f''(0) \approx -0,44374733$. Az U megoldás tartalmazza az f , f' és f'' diszkrét értékeit. Az program futását az `Iteration` (Függelék 7.2.3), a megoldás ábrázolását a `VisualizingResults` (Függelék 7.2.4) script végzi. Megemlítjük, hogy a *Levenberg-Marquardt* és a *trust-region dogleg* megoldók az `Optimization Toolbox`³ részei és a 2011a verzióban maximum nyolc szálon képesek dolgozni. A Jacobi-mátrix megadásának fontosságát bizonyítja, hogy ugyanazon a négymagos processzoron³ $N = 300$, $c = 5$, $n = 1$ paraméterértékek mellett ha mi biztosítottuk a Jacobi-mátrixot a program számára, akkor a számítási idő 323,95s volt, míg ha annak kellett számolnia, a futásidő 2640,53s lett, a függvénykiértékelések száma 3873 helyett 1539108 volt.

A kvázi-egyenletes rácson is lehet alkalmazni a Richardson-extrapolációt hibabecslésre. Elvégeztük a becslést néhány n paraméterre. A durvább rácsot $N = 100$ -nak, a finomabbat $N = 200$ -nak választottuk. A pontbeli értékek eltérésének végtelen normáját vettük a `SakiadisRichardson` (Függelék 7.2.5) segítségével, az így számított hibákat és a keresett $f''(0)$ értékeket a 3. táblázatban összegeztük.

	$n = 0,7$ ($c = 10$)	$n = 0,8$ ($c = 7$)	$n = 1$ ($c = 5$)	$n = 1,1$ ($c = 4$)	$n = 1,3$ ($c = 3$)	$n = 1,5$ ($c = 2$)
$f''(0)$	-0,442721	-0,440682	-0,443749	-0,447238	-0,456220	-0,466553
$N = 100, 200$	5,4424e-4	5,7903e-5	4,9797e-5	3,8209e-4	1,1577e-4	9,4670e-3

3. táblázat. Az $f''(0)$ értékei és a becsült hibák különböző n esetén

Ezt a módszert alkalmaztuk a [8] TDK dolgozatban vizsgált általánosított Blasius feladatra mozgó síklap esetén is. Jelenlegi differencia módszer és az ott ismertetett iteratív transzformációs módszerrel számított megoldás négy tizedesjegyre megegyező eredményt adott.

³Intel® Core™ i5-2500K @4500MHz

5. Chebyshev pszeudospektrál módszer

A spektrál módszereket egyes feladatokra a 70-es éveken kezdték alkalmazni, matematikai megalapozása Gottlieb és Orszag nevéhez fűződik 1977-ben, de csak a 90-es évektől került a számítástudományban gyakran alkalmazott módszerek közé [6]. Először a folyadékmechanikával és az időjárás modellezéssel foglalkozó szakemberek fordultak a spektrál módszerekhez. Jellemzője, hogy globális módszer, azaz a tartomány minden pontjában előállítja a közelítő megoldást, szemben például a véges differencia módszerrel. Korábban hátránya volt, hogy több dimenziós feladatoknál csak egyszerű tartományokon (téglalap, körlap, körgyűrű) lehetett alkalmazni, a spektrálem módszer kialakulásával viszont ez a hátrány kiküszöbölhetővé vált. Annak ellenére, hogy napjainkban leggyakrabban áramlástani problémák megoldására használják, egyre inkább alkalmazzák a biomechanikában, az asztrofizikában és az elektromágneses hullámok vizsgálatában. Egyik legnagyobb előnye, hogy nagy a konvergenciasebessége és pontossága.

Közelítsünk egy ismeretlen $u(x)$ függvényt polinomokkal. Jelöljük a legfeljebb N -edfokú polinomok halmazát $\mathbb{P}_N$ -nel és legyen ennek egy eleme $p_n(x)$, $n < N$. Gyakran célszerű a polinomokat úgy megválasztani, hogy ortogonálisak legyenek egy bizonyos súlyfüggvényre nézve, azaz

$$\int_a^b p_m(x)p_n(x)w(x)dx = c\delta_{mn}, \quad c \in \mathbb{R} \quad (5.1)$$

teljesüljön. A spektrál kollokációs vagy más néven pszeudospektrál módszerek elve a következő. Vegyünk fel az értelmezési tartományban $N + 1$ számú pontot: $x_0, \dots, x_N$ – ezek az úgynevezett kollokációs pontok. Írjuk fel az interpolációs függvényt a $\mathbb{P}_N$ -beli függvényekkel (bázisfüggvények):

$$P_N u = \sum_{n=0}^N a_n p_n(x), \quad (5.2)$$

ahol a_n ($n = 0, \dots, N$) a meghatározandó együtthatók, melyeket a kollokációs pontokra vonatkozó egyenletekből határozunk meg:

$$\sum_{n=0}^N a_n p_n(x_i) = u(x_i), \quad i = 0, \dots, N. \quad (5.3)$$

Az interpolációs függvény deriváltja, $D_N u$, mely $N - 1$ -edfokú és $D_N u = (P_N u)'$. Közelítő értékeinek a kollokációs pontokban történő kiszámítására alapvetően két módszert szoktak alkalmazni. Az egyik a mátrix szorzás, a másik módszer a gyors Fourier-transzformáció *FFT*. Dolgozatunkban mi az előbbit használjuk, ugyanis az *FFT* használata – feladattól függően – $N > 64$ felett ajánlatos. Tehát

$$D_N u(x_i) = \sum_{n=0}^N [D_N]_{in} u(x_n), \quad i = 0, \dots, N, \quad (5.4)$$

ahol $[D_N]_{in}$ a differenciáló mátrix. Az approximáció során elsőfajú Chebyshev polinomokat alkalmazunk. Az alábbiakban összefoglaljuk a legfontosabb tulajdonságait. Az

$$(1 - x^2)y'' - xy' + n^2y = 0, \quad n \in \mathbb{N}$$

Chebyshev-féle differenciálegyenlet megoldásai a $T_n(x)$ ($n = 0, \dots, n$) elsőfajú n -edfokú Chebyshev polinomok, melyeket az alábbi módon definiálnak:

$$T_n(\cos \theta) = \cos n\theta, \quad \theta \in [0, \pi], \quad n \in \mathbb{N}.$$

Trigonometrikus átalakítások után rekurzióval is előállíthatók (lásd `chebpoly`, Függelék 7.3.4):

$$\begin{aligned} T_0(x) &= 1, \\ T_1(x) &= x, \\ T_n(x) &= 2xT_{n-1}(x) - T_{n-2}(x), \quad n \geq 2. \end{aligned}$$

Ha n páros, akkor $T_n(x)$ is páros, ha n páratlan, akkor $T_n(x)$ is páratlan. A $\{T_n(x)\}_{n=0}^N$ polinomok a $w(x) = 1/\sqrt{1-x^2}$ súlyfüggvényre nézve ortogonálisak. Szélsőérték helyük egybeesik a Gauss-Lobatto integrálási pontokkal, ezért az $x_j = \cos(j\pi/N)$, $j = 0, \dots, N$ pontokat Chebyshev-Gauss-Lobatto pontoknak is nevezik [23].

Induljunk ki ismét a differenciálegyenlet (4.9) alakjából. Az előzőekből láttuk, hogy a Chebyshev polinomok a $[-1, 1]$ intervallumon vannak értelmezve, ezért csonkítsuk a $[0, \infty)$ intervallumot a $[0, L]$ -re, majd pedig a $[-1, 1]$ -re a következő módon.

$$\eta \in [0, \infty) \longrightarrow \xi \in [0, L] \xrightarrow{\zeta = \frac{\xi}{L}} \zeta \in [0, 1] \xrightarrow{x = 2\zeta - 1} x \in [-1, 1] \quad (5.5)$$

Az intervallumcsonkítással két új hibaforrásunk lesz. Hiba keletkezik egyrészt azért, mert nem a félig végtelen intervallumon oldjuk meg a peremértékfeladatot. További hiba lép fel azért is, mert az egyik peremfeltételt nem a végtelenben írjuk elő zérusnak. Ebből következik, hogy az számítási intervallum hosszát (L) és a bázisfüggvények számát (N) egyidejűleg kell növelnünk [5].

Írjuk fel a (4.9) differenciálegyenletet és a peremfeltételeket az x változó függvényeként. A deriváltakat kiszámítva $\bar{f}(x)$ -re

$$\begin{aligned} \bar{f}'_x &= f'_\xi \xi'_x = \frac{L}{2} f'_\xi \Rightarrow f'_\xi = \frac{2}{L} \bar{f}'_x \\ \bar{f}''_{xx} &= \frac{L^2}{4} f''_{\xi\xi} \Rightarrow f''_{\xi\xi} = \frac{4}{L^2} \bar{f}''_{xx} \\ f'''_{\xi\xi\xi} &= \frac{8}{L^3} \bar{f}'''_{xxx} \end{aligned}$$

és behelyettesítve a (4.9) differenciálegyenletbe és a (2.15) peremfeltételekbe, az új peremértékfeladat:

$$\frac{8}{L^3} \bar{f}''' - \frac{1}{n(n+1)} \left(\frac{4}{L^2} \right)^{2-n} \bar{f} |\bar{f}''|^{2-n}, \quad (5.6)$$

$$\bar{f}(-1) = 0, \quad \bar{f}'(-1) = L/2, \quad \bar{f}'(1) = 0. \quad (5.7)$$

Mivel $N + 1$ számú ismeretlenünk van és $N + 2$ egyenletünk (a differenciálegyenlet diszkrét alakja az $N - 1$ belső pontban és a három peremfeltétel), ezért az egyenletrendszer túlhatározott lenne. Az egyik módszer az, hogy olyan polinomot alkalmazunk, amely már magába foglal egy peremfeltételt, vagy pedig egy fiktív csomópontot vezetünk be, ezáltal nő az ismeretlenek száma. Mi az első módszert használtuk. Keressük az új g approximációs polinomot az alábbi alakban:

$$\bar{f}(x) = P(x)g(x),$$

ahol

$$P(x) = ax^2 + bx + c. \quad ^4$$

Ahhoz, hogy a $P(-1) = 0$, $P'(-1) = L/2$ és $P'(1) = 0$ teljesüljön, az a , b , c együtthatókra

$$a = -L/8, \quad b = L/4, \quad c = 3L/8$$

adódik. A (5.7) peremfeltételek az új g függvényvel:

$$\begin{aligned} \bar{f}(-1) &= \underbrace{P(-1)}_0 g(-1) = 0 \Rightarrow g(-1) \text{ tetszőleges értéket felvehet} \\ \bar{f}'(-1) &= L/2 g(-1) = L/2 \Rightarrow g(-1) = 1 \\ \bar{f}'(1) &= \underbrace{P(1)}_{L/2} g'(1) = 0 \Rightarrow g'(1) = 0. \end{aligned}$$

Az új peremértékfeladat:

$$\begin{aligned} \frac{8}{L^3} [6ag' + (6ax + 3b)g'' + (ax^2 + bx + c)g'''] - \frac{1}{n(n+1)} \left(\frac{4}{L^2} \right)^{2-n} (ax^2 + bx + c)g \\ |2ag + (4ax + 2b)g' + (ax^2 + bx + c)g''|^{2-n} = 0, \end{aligned} \quad (5.8)$$

$$g(-1) = 1, \quad g'(-1) = 0. \quad (5.9)$$

Rövidítsük a (5.4)-beli differenciáló mátrixot D -vel. Általában nem igaz az, hogy $D^{(2)} = D^2$, azaz a második deriváltakat tartalmazó mátrix számítható a differenciáló mátrix négyzeteként, azonban a Chebyshev pszeudospektrál módszerre ez fennáll [23]. A D és $D^{(2)}$ kiszámítására

⁴Az itt szereplő g függvény független a skálázási csoportnál bevezetett g függvénytől.

egzakt alak létezik:

$$\begin{cases} D_{kj} = \frac{c_k}{c_j} \frac{(-1)^{j+k}}{x_k - x_j}, & k \neq j \\ D_{kk} = -\frac{1}{2} \frac{x_k}{1-x_k^2}, & k \neq 0, N \\ D_{00} = \frac{2N^2+1}{6} \\ D_{NN} = -\frac{2N^2+1}{6} \end{cases} \quad (5.10)$$

Ezt a képletet azonban nem célszerű alkalmazni, mert nagy lesz D kondíciószáma az egymáshoz közel álló számok kivonása miatt. A Chebyshev pontokat és a differenciáló mátrixot előállító program a `cheb` (Függelék 7.3.1), amelyet a [23]-beli könyvből vettünk át. Jellegzetessége a pszeudospektrál módszernek, hogy kis méretű telt mátrixú egyenletrendszert kapunk, szemben például a véges differenciák módszerével, vagy a végeelem módszerrel, ezért direkt módszerrel oldjuk meg az egyenletrendszert. Az (5.9) differenciálegyenletbe a diszkrét értékeket beírjuk, azaz $g(x) \rightarrow g(x_i)$, $g'(x) \rightarrow Dg(x_i)$, $g''(x) \rightarrow D^2g(x_i)$, $g'''(x) \rightarrow D^3g(x_i)$. Az (5.9)-beli peremfeltételek számítására: $g(x_N) = 1$ és $D(0, \mathbf{j})g(\mathbf{x}) = \mathbf{0}$, ahol $\mathbf{j} = [0 \ 1 \dots N]$, $\mathbf{x} = [x_0 \ x_1 \dots x_N]^T$. Az így előállított nemlineáris egyenletrendszert a `sakcheb` (Függelék 7.3.2) függvénybe vittük be. Az egyenletrendszer megoldása során ugyanazt a két megoldót használtuk, mint az előző fejezetben; a beállított paraméterek ugyanazok, kivéve, hogy most nem adtuk meg a Jacobi-mátrixot, mert a megoldás így is nagyon gyors. Ha megkaptuk g -re a közelítő megoldást, akkor az f és deriváltjai is számolhatók. A gyakorlatban is használható általános hibabecslő formulát nem találtunk, de léteznek a gyakorlatban széles körben alkalmazott szabályok, amelyek segítségével valószínűsíthető, hogy a megoldás helyes-e vagy sem. Két ilyen módszert vizsgáltunk. Egyrészt fennáll, hogy a hiba nagyságrendje az a_N Chebyshev együttható abszolút értéke, másrészt ha $\{a_i\}_{i=0}^N$ monoton csökkenő módon tart nullához, akkor az eredmény valószínűleg jó [5]. Ehhez először képeztük a Chebyshev polinomokat a `chebpoly` függvénnyel (Függelék 7.3.4), majd az (5.3)-beli interpolációs feladatot oldottuk meg a keresett együtthatókra a `chebvand` (Függelék 7.3.3) futtatásával.

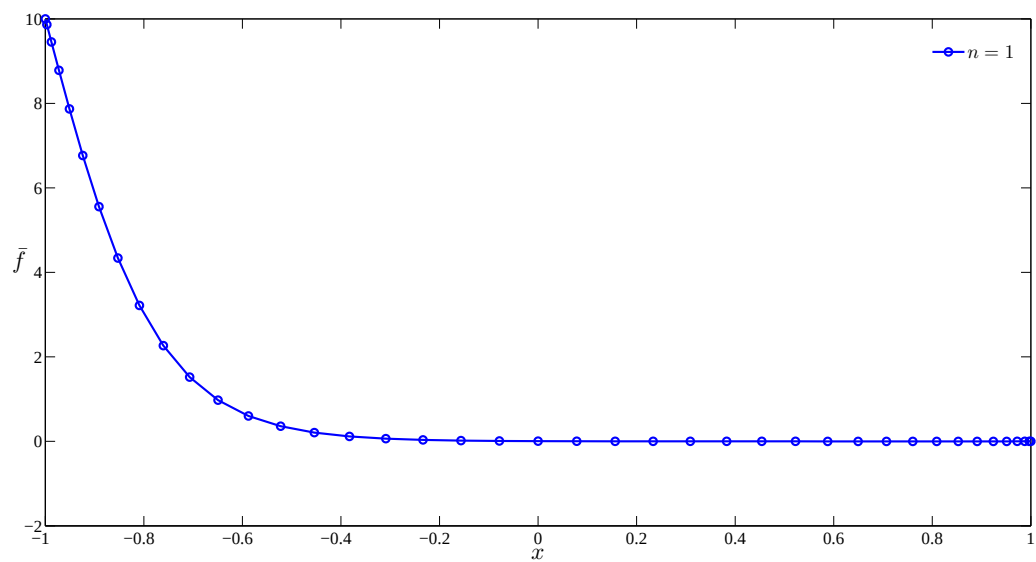
$$\begin{bmatrix} T_0(x_0) & T_1(x_0) & \cdots & T_N(x_0) \\ T_0(x_1) & T_1(x_1) & \cdots & T_N(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ T_0(x_N) & T_1(x_N) & \cdots & T_N(x_N) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix}$$

Az egyenletrendszer együttható mátrixa az úgynevezett Chebyshev-Vandermonde mátrix. A lehető legnagyobb L és N paraméterekhez tartozó megoldás szerepel a 4. táblázatban.

	$n = 0, 5$ ($L = 30, N = 23$)	$n = 0, 7$ ($L = 25, N = 25$)	$n = 0, 8$ ($L = 20, N = 20$)	$n = 1$ ($L = 20, N = 40$)
$f''(0)$	-0,468917	-0,443623	-0,441151	-0,443763
	$n = 1, 1$ ($L = 10, N = 20$)	$n = 1, 3$ ($L = 6, N = 20$)	$n = 1, 5$ ($L = 6, N = 10$)	
$f''(0)$	-0,447283	-0,45624	-0,466919	

4. táblázat. Az $f''(0)$ értékei különböző n esetén

A 3. ábra az $\bar{f}'(x)$ megoldásfüggvényt mutatja.



3. ábra. Az $\bar{f}'(x)$ függvény $n = 1$ esetén

6. Összefoglalás, további célok

Newtoni folyadékra a Sakiadis-feladat numerikus megoldását többféle módszerrel számos dolgozatban vizsgálják [3], [11], [14]. A dolgozatban a Sakiadis-feladat általánosítását vizsgáltuk, amikor nyugvó nem-newtoni hatványtörvény szerinti közegben mozgó síklap feletti sebességprofil meghatározását tűztük ki célul. Három közelítő módszert alkalmaztunk, bemutattuk az egyes módszerek előnyeit és hátrányait, továbbá konkrét hibabecslő eljárást alkalmaztunk a véges differencia módszerhez. A pszeudospektrál megoldásnál az 5. fejezetben leírt hibaellenőrző szabályokat betartva a megoldás helyességére kaptunk rálátást. A differencia módszerrel és a pszeudospektrál módszerrel kapott eredmények jó egyezést mutatnak és eltérnek az iteratív transzformációs módszerrel nyert eredményektől.

További célunk, hogy olyan spektrál módszert alkalmazzunk, amely a félig végtelen intervallumot közvetlenül kezeli – ilyenek például a Laguerre, illetve racionális Chebyshev függvények. Ez várhatóan kiküszöböli azt a jelenséget, hogy $n < 0.8$ esetén az L számítási hossz nem mindig elég. Az N növelése egy bizonyos határon túl nem válik be, mivel a differenciálmátrixok kondíciószáma fokozatosan nő; célunk tehát a mátrixok prekondicionálása.

7. Függelék

7.1. Jelölések

Jelölés	Megnevezés	Mértékegység
<u>latin betűk</u>		
u	x irányú sebességkomponens	m/s
	függvény (4. fejezetben definiálva)	-
v	y irányú sebességkomponens	m/s
K	konzisztencia index	$Pa s^n$
x	derékszögű koordináta	m
	független változó (4. fejezetben definiálva)	-
n	hatványtörvény kitevő	-
U_w	síklap sebessége	m/s
f	hasonlósági függvény	-
a, b	paraméterek	-
Re_x	helyi Reynolds-szám	-
C_D	ellenállás-tényező	-
g	hasonlósági függvény (3. fejezetben definiálva)	-
	függvény (5. fejezetben definiálva)	-
p	szabályozó paraméter	-
h^*	numerikus paraméter	-
L	csonkított intervallum hossza	m
c	szabályozó paraméter	-
N	intervallum felosztás száma	-
$f_i, U_i (i = 1, \dots, 3N + 3)$	változók	-
J	Jacobi-mátrix	-
T_n	elsőfajú n -edfokú Chebyshev polinom	-
D	Chebyshev differenciáló mátrix	-
$\bar{f}$	függvény	-
P	másodfokú polinom	-
<u>görög betűk</u>		
τ_{yx}	x normálisú síkon y irányban ébredő nyírófeszültség	N/m^2
ρ	sűrűség	kg/m^3
γ	paraméter	$m^2 s^{n-2}$
ψ	áramfüggvény	m^2/s
η	hasonlósági változó	-
α, β, γ	paraméterek	-
τ_w	fal menti nyírófeszültség	N/m^2
η^*	hasonlósági változó	-
λ	csoport paraméter	-
σ	paraméter	-
ξ	változó	-

7.2. Programok a véges differencia módszerhez

7.2.1. fun4sakiadis.m

```
1 function [F J] = fun4sakiadis(u)
2 % Gives the system of nonlinear equations and the corresponding Jacobian
3 % for the Blasius-problem using quasi-uniform finite difference grid
4 N = 200; % number of intervals
5 d = 5; % a parameter which controls the distribution of the mapped points
6 n = 1;
7 [x,a,b,c] = QuasiUniform(N,d); % performs quasi-uniform mapping
8 a = a'; b = b'; c = c'; % make a column vector
9 F = zeros(3*N+3,1);
10 i = 1:N;
11 j = N+2:2*N+1;
12 k = 2*N+3:3*N+2;
13 F(i) = u(i+1)-u(i)-a.*(b.*u(j+1)+c.*u(j));
14 F(N+1) = u(1);
15 F(j) = u(j+1)-u(j)-a.*(b.*u(k+1)+c.*u(k));
16 F(2*N+2) = u(N+2)-1;
17 F(k) = u(k+1)-u(k)-1/(n*(n+1))*a.*(b.*u(i+1)+c.*u(i)).*(abs((b.*u(k+1)+c.*u(k))))^(2-n);
18 F(3*N+3) = u(2*N+2);
19
20 % Calculate the analytical value of the Jacobian. By passing this argument
21 % to the fsolve solver, the solver needs less computation time.
22 J = zeros(3*N+3);
23 J(1:N,1:N) = diag(-ones(N,1)) + diag(ones(N-1,1),1);
24 J(N,N+1) = 1;
25 J(1:N,N+2:2*N+1) = diag(-a.*c) + diag(-a(1:N-1).*b(1:N-1),1);
26 J(N,2*N+2) = -a(N)*b(N);
27 J(N+1,1) = 1;
28 J(N+2:2*N+1,N+2:3*N+3) = J(1:N,1:2*N+2);
29 J(2*N+2,N+2) = 1;
30 J(2*N+3:3*N+2,1:N) = ...
    diag(-1/(n*(n+1))*a.*c.*(abs((b.*u(2*N+4:3*N+3)+c.*u(2*N+3:3*N+2))))...
31    .^(2-n))+diag(-1/(n*(n+1))*a(1:N-1).*b(1:N-1).*(abs((b(1:N-1).*u(2*N+4:3*N+2)+...
32    c(1:N-1).*u(2*N+3:3*N+1))))^(2-n),1);
33 J(3*N+2,N+1) = -1/(n*(n+1))*a(N)*b(N).*(abs((b(N).*u(3*N+3)+c(N).*u(3*N+2))))^(2-n);
34 J(2*N+3:3*N+2,2*N+3:3*N+2) = diag(-1-1/(n*(n+1))*a.*c.*(b.*u(2:N+1)+c.*u(1:N)).*(2-n)...
35    .*(abs((b.*u(2*N+4:3*N+3)+c.*u(2*N+3:3*N+2))))^(1-n).*sign((b.*u(2*N+4:3*N+3)+...
36    c.*u(2*N+3:3*N+2))))+diag(1-1/(n*(n+1))*a(1:N-1).*b(1:N-1).*(b(1:N-1).*u(2:N)+...
37    c(1:N-1).*u(1:N-1)).*(2-n).*(abs((b(1:N-1).*u(2*N+4:3*N+2)+c(1:N-1).*u(2*N+3:3*N+1))))...
38    .^(1-n).*sign((b(1:N-1).*u(2*N+4:3*N+2)+c(1:N-1).*u(2*N+3:3*N+1))),1);
39 J(3*N+2,3*N+3) = ...
    1-1/(n*(n+1))*a(N)*b(N).*(b(N).*u(N+1)+c(N).*u(N)).*(2-n).*(abs(b(N).*u(3*N+3)+...
40    c(N).*u(3*N+2))))^(1-n).*sign((b(N).*u(3*N+3)+c(N).*u(3*N+2)));
41 J(3*N+3,2*N+2) = 1;
42 end
```

7.2.2. QuasiUniform.m

```
1 function [x,a,b,c] = QuasiUniform(N,d)
2 % Set up a quasi-uniform grid for a semi-infinite interval.
3 % The grid produced is applicable for solving differential equations with
4 % the finite difference method (FDM) on a semi-infinite interval.
5 % Code is written according to the article:
6 %     R. Fazio: Finite difference schemes on quasi-uniform grids for BVPs
7 %     on infinite intervals, 2011.
8 %
9
10 for alpha = 0:1/4:3/4
11     n = 0:N-1;
12     xi = (n+alpha)/N;
13     if alpha==0
14         x = -d*log(1-xi); x = [x Inf];
15     elseif alpha==1/4
16         xx = -d*log(1-xi); xx = [xx Inf];
17     elseif alpha==1/2
18         xxx = -d*log(1-xi); xxx = [xxx Inf];
19     elseif alpha==3/4
20         xxxx = -d*log(1-xi); xxxx = [xxxx Inf];
21     end
22 end
23 i = 1:N;
24 a = 2*(xxxx(i)-xx(i));
25 b = (xxx(i)-x(i))./(x(i+1)-x(i)); b(N) = b(N-1);
26 c = (x(i+1)-xxx(i))./(x(i+1)-x(i)); c(N) = c(N-1);
```

7.2.3. Iteration.m

```
1 N = 200;
2 u0 = (1:(N+1)*3)';
3 matlabpool('open');
4 options=optimset('Display','iter','MaxIter',Inf,'MaxFunEvals',Inf, ...
5     'Algorithm','levenberg-marquardt','UseParallel','always','Jacobian','on');
6 tic, [u fval] = fsolve(@fun4sakiadis,u0,options); toc
7 str = sprintf('f(0)= %.6f',min(u)); disp(str);
```

7.2.4. VisualizingResults.m

```
1 N = 200;
2 d = 5;
3 [x,a,b,c] = QuasiUniform(N,d);
4 U = [u(1:length(u)/3) u(length(u)/3+1:length(u)/3*2) u(length(u)*2/3+1:end)];
5 figure; plot(x,U,'-o');
```

7.2.5. SakiadisRichardson.m

```
1 N = 100;
2 u100 = [u100(2:N+1); u100(N+3:2*N+1); u100(2*N+3:end)];
3 N = 200;
4 u200 = [u200(2:N+1); u200(N+3:2*N+1); u200(2*N+3:end)];
5 u200_red = u200(2:2:end);
6 R = (u200_red-u100)/3;
7 R = norm(R, Inf);
```

7.3. Programok a pszeudospektrál módszerhez

7.3.1. cheb.m

```
1 function [D,x] = cheb(N)
2 x = cos(pi*(0:N)/N)';
3 c = [2; ones(N-1,1); 2].*(-1).^(0:N)';
4 X = repmat(x,1,N+1);
5 dX = X-X';
6 D = (c*(1./c)') ./ (dX+(eye(N+1)));
7 D = D - diag(sum(D'));
```

7.3.2. sakcheb.m

```
1 function DD = sakcheb(g)
2 global n L N;
3 a = -L/8; b = L/4; c = 3/8*L;
4 [D,x] = cheb(N);
5 DD = 8/L^3*(6*a*D*g+diag(6*a*x+3*b)*D^2*g+diag(a*x.^2+b*x+c)*D^3*g) - ...
6     1/(n*(n+1))*(4/L^2)^(2-n)*diag(a*x.^2+b*x+c)*g .* ...
7     (abs(2*a*g+diag(4*a*x+2*b)*D*g+diag(a*x.^2+b*x+c)*D^2*g)).^(2-n);
8 DD(1) = D(1,:)*g;
9 DD(end) = g(end)-1;
10 end
```

7.3.3. chebvand.m

```
1 function TV = chebvand(n)
2 [T,x] = cheb(n);
3 T = chebpoly(n);
4 TV(length(x),n+1) = 0;
5 for k = 1:n+1
6     TV(:,k) = polyval(T{k},x);
7 end
```

7.3.4. chebpoly.m

```
1 function C = chebpoly(n)
2 % C = CHEBPOLY(N) returns Chebyshev polynomials of the first kind from
3 % degree 0 to n
4
5 C = cell(n+1,1); % using cell array because of the different matrix sizes
6 C{1} = 1; % T_0 = 1
7 C{2} = [1 0]; % T_1 = x
8 for k = 3:n+1 % T_n = 2xT_{n-1} - T_{n-2}
9     C{k} = [2*C{k-1} 0] - [0 0 C{k-2}];
10 end
```

7.3.5. SakiadisCheb.m

```
1 global n L N;
2 n = 1;
3 L = 10;
4 N = 40;
5 [D,x] = cheb(N); a = -L/8; b = L/4; c = 3/8*L;
6 options=optimset('Display','iter','MaxIter',Inf,'MaxFunEvals',Inf, ...
7     'Algorithm','levenberg-marquardt','UseParallel','always','Jacobian','off');
8 tic, [g funval] = fsolve(@sakcheb,x,options); toc;
9 gprime = D*g;
10 g2prime = D^2*g;
11 f = (a*x.^2+b*x+c).*g;
12 fprime = 2/L*((2*a*x+b).*g + (a*x.^2+b*x+c).*gprime);
13 f2prime = 4/L^2*(2*a*g+(4*a*x+2*b).*gprime+(a*x.^2+b*x+c).*g2prime);
14 xi = L/2*(x+1);
15 plot(xi,f,'-o',xi,fprime,'-o',xi,f2prime,'-o');
16 str = sprintf('f'(0)= %.6f',f2prime(end)); disp(str);
```

8. Hivatkozások

- [1] Altan, T., Oh, S., and Gegel, H. *Metal Forming Fundamentals and Applications*. Asm. Intl., 1979.
- [2] Andersson, H. I. and Aarseth, J. B. Sakiadis flow with variable fluid properties revisited. *International Journal of En*, 45:554–561, 2007.
- [3] Bataller, R. C. Numerical comparisons of Blasius and Sakiadis flows. *MATEMATIKA*, 26 (2):187–196, 2010.
- [4] Bognár, G. and Csáti, Z. Iterative transformation method for the generalized sakiadis problem. In *XXVII. microCAD International Scientific Conference*, 2013.
- [5] Boyd, J. P. *Chebyshev and Fourier Spectral Methods*. Dover Publications, 2000.
- [6] Canuto, C., Hussaini, M. Y., Quarteroni, A., and Zang, T. A. *Spectral Methods: Fundamentals in Single Domains*. Springer, first edition, 2006.
- [7] Csáti, Z. Hatványközeg hasonlósági megoldásai konvektív felületi peremfeltétellel, 2011. TDK dolgozat.
- [8] Csáti, Z. Hatványközeg határréteg áramlását leíró egyenletek numerikus és elméleti megoldása, 2012. TDK dolgozat.
- [9] Fazio, R. A novel approach to the numerical solution of boundary value problems on infinite intervals. *SIAM J. Numer. Anal.*, 33:1473–1483, 1996.
- [10] Fazio, R. A Numerical Test for the Existence and Uniqueness of Solution of Free Boundary Problems. *Applicable Analysis*, 66:89–100, 1997.
- [11] Fazio, R. The Iterative Transformation Method for the Sakiadis Problem. <http://mat521.unime.it/fazio/preprints/Sakiadis.pdf>, April 2012.
- [12] Fazio, R. and Jannelli, A. Finite difference schemes on quasi-uniform grids for Bvps on infinite intervals. <http://mat521.unime.it/fazio/preprints/Quniform.pdf>, December 2011.
- [13] Fisher, E. G. *Extrusion of Plastics*. Wiley, New York, 1976.
- [14] Ishak, A., Nazar, R., and Pop, I. Boundary layer on a moving wall with suction and injection. *Chin. Phys. Lett.*, 24:2274–2276, 2007.
- [15] Lajos, T. *Az áramlástan alapjai*. Budapest, 4. kiadás edition, 2008.
- [16] Pantokratoras, A. Further results on the variable viscosity on flow and heat transfer to a continuous moving flat plate. *International Journal of Engineering Science*, 42:1891–1094, 2004.

- [17] Pop, I., Gorla, R. S. R., and Rashidi, M. The effect of variable viscosity on flow and heat transfer to a continuous moving flat plate. *International Journal of Engineering Science*, 30:1–6, 1992.
- [18] Prandtl, L. Über Flüssigkeitsbewegung bei sehr kleiner Reibung. *Intern. Math. Kongr.*, 3: 484–491, 1904.
- [19] Sakiadis, B. C. Boundary-layer behavior on continuous solid surfaces: I. boundary-layer equations for two-dimensional and axisymmetric flow. *AIChE*, 7:26–28, 1961.
- [20] Sakiadis, B. C. Boundary layer behavior on continuous solid surfaces. ii: The boundary layer on a continuous flat surface. *AIChE*, 7:221–225, 1961.
- [21] Tadmor, Z. and Klein, I. *Engineering Principles of Plasticating Extrusion*. Polymer Science and Engineering Series. Van Nostrand Reinhold, New York, 1970.
- [22] Takhar, H. S., Nitu, S., and Pop, I. Boundary layer flow due to a moving plate: variable fluid properties. *Acta Mechanica*, 90:37–42, 1991.
- [23] Trefethen, L. N. *Spectral Methods in MATLAB*. SIAM, England, 2000.
- [24] Tsou, F. K., Sparrow, E. M., and Goldstein, R. J. Flow and heat transfer in the boundary layer on a continuous moving surface. *Int. J. Heat Mass Transfer*, 10:219–235, 1967.