

NUMERIKUS MATEMATIKA KURZUS, PROJEKTFELADAT
NUMERIKUS INTEGRÁLÁS, KVADRATÚRAFORMULÁK

Unger Tamás István

B.Sc. szakos matematika hallgató

ungert@maxwell.sze.hu, <http://maxwell.sze.hu/~ungert>

KIVONAT. Választott projektmunkám témája a numerikus integrálás. A dokumentumban az elméleti áttekintés részeként ismertetem a határozott integrál fogalmát, valamint általánosan bemutatom a különböző technikák által használt közelítési módszerek alapelvét. Bemutatom a Lagrange-féle interpolációs polinomot, ismertetem annak alkalmazását az interpolációs numerikus integrálási formulák esetén. Ismertetek néhány alapvető Newton-Cotes formulát. Röviden ismertetem a Gauss-típusú kvadraturák elméletét, végül pedig bemutatok egy kubatúra-módszert speciálisan egy négyzet felett, valamint általános téglalapok felett is.

1. Bevezetés, motivációk

Egy- vagy többváltozós függvények határozott integráljának kiszámításával az analízis alkalmazásának számos területén találkozhatunk. Mérnöki tanulmányaim során a különböző elektromágneses problémák vizsgálata során szinte minden esetben elegendhetetlen eszköznek bizonyult, de széleskörű használatára van szükség a fizika egyéb tudományterületein is.

A gyakorlati alkalmazások során sokszor szembesülhetünk, hogy a szükséges primitív függvény nem, vagy csak rendkívül nehezen adható meg zárt alakban, ezért az előtanulmányainkban már megismert Newton-Leibniz-formula sem alkalmazható. Ilyen esetekben nem is törekszünk a határozott integrál pontos meghatározására, csupán az integrál értékének egy – valamilyen pontosságú – meghatározására van szükségünk. Erre szolgálnak a különböző numerikus integrálási technikák, melynek számos – az adott probléma megoldásához optimális – változata létezik.

A továbbiakban ezekből szeretnék bemutatni a teljesség igénye nélkül néhányat, speciálisan az egy- és kétváltozós függvények numerikus integrálására alkalmazva.

2. A határozott integrál definíciója

Tekintsünk egy $f(x)$ egyváltozós függvényt, mely egy $[a, b]$ zárt intervallum minden egyes pontjában értelmezett. Azt mondjuk, hogy ennek az $f(x)$ függvénynek az a -tól b -ig értelmezett határozott integrálja [1]:

$$I := \int_a^b f(x) \, dx = \lim_{n \rightarrow \infty} \sum_{i=1}^n \Delta x_i f(\bar{x}_i), \quad (1)$$

amely összefüggés tulajdonképpen a Riemann-féle közelítő összegek határértéke. A definícióban Δx_i a teljes $[a, b]$ intervallum felosztásának i -edik részintervallumának hossza ($\Delta x_i = x_i - x_{i-1}$), az $f(\bar{x}_i)$ függvényérték pedig ennek az i -edik részintervallumnak egy tetszőlegesen kiválasztott pontjához tartozó függvényérték, formálisan leírva: $\bar{x}_i \in [x_{i-1}, x_i]$. Világos az is, hogy az x_i pontok az eredeti $[a, b]$ intervallum egy n -től függő úgynevezett felosztását képezik, így

$$a = x_0 < x_1 < x_2 < \dots < x_i < \dots < x_n = b. \quad (2)$$

Amennyiben az (1) által definiált határérték létezik, úgy az $f(x)$ függvény integrálható az $[a, b]$ intervallumon. Klasszikus kalkulus-kurzusokról ismert, hogy amennyiben $f(x)$ F -fel jelölt primitív függvénye (definíció szerint: $F' = f(x)$) ismert, úgy alkalmazható a Newton-Leibniz-formula [2]:

$$\int_a^b f(x) \, dx := [F]_a^b = F(b) - F(a). \quad (3)$$

Világos, hogy $f(x)$ F primitív függvénye a gyakorlati alkalmazások esetén sokszor nem, vagy csak nagyon nehezen adható meg zárt alakban. Az ilyen esetekben a Newton-Leibniz-formula sem alkalmazható, így más módszerekhez kell folyamodnunk. Ezek a módszerek a numerikus integrálási technikák, melyek a határozott integrál közelítő értékének meghatározására szolgálnak, és amelyeknek széleskörű gyakorlati alkalmazása ismert.

3. A közelítés alapelve

Jelöljük az $f(x)$ függvény $[a, b]$ intervallumon vett határozott integráljának értékét $I(f)$ -fel, azaz

$$I(f) := \int_a^b f(x) \, dx. \quad (4)$$

Ekkor $I(f)$ egy közelítését a következőképpen tudjuk meghatározni:

$$I_n(f) = \sum_{i=0}^n c_i f(x_i), \text{ ahol } x_i \in [a, b]. \quad (5)$$

Definíció szerint az $I_n(f) = I_n(f, \{c_0, x_0\}, \{c_1, x_1\}, \dots, \{c_n, x_n\})$ képletet *kvadraturaképletnek* nevezzük, c_i értékeit a kvadraturaképlet *súlyainak*, x_i értékeit pedig a kvadraturaképlet *alappontjainak* hívjuk [1].

Vegyük észre, hogy az alapelv az, hogy az eredeti $I(f)$ határozott integrált $I_n(f)$ összeggel közelítjük. Világos az is, hogy $I_n(f)$ értéke és pontossága jelentősen függ egyrészt a súlyok és alappontok számától, valamint azok megválasztásának módjától is. Az is világos, hogy mivel $I_n(f)$ közelítés, így szükséges és érdemes vizsgálni az eltérést $I(f)$ és $I_n(f)$ között. Mivel a kvadraturaképletek pontosságának kérdése összetett tudományterület, így ezzel projektfeladatunk csak érintőlegesen fog foglalkozni.

4. Interpolációs képletek, Lagrange-féle interpolációs polinom

A határozott integrál közelítő meghatározásához tehát az

$$\int_a^b f(x) dx \approx I_n(f) = \sum_{i=0}^n c_i f(x_i), \quad x_i \in [a, b]. \quad (6)$$

összefüggést alkalmazzuk, ahol c_i súlyok egyelőre még ismeretlen együtthatók, melyeket meg kell határoznunk. Kérdés, hogyan.

Az interpolációs kvadratúráképletek bevezetéséhez és megértéséhez először is be kell vezetnünk a *Lagrange-féle interpolációs polinom* fogalmát. Ehhez tekintsük az $(x_0, x_1, \dots, x_n)$ és $(f(x_0), f(x_1), \dots, f(x_n))$ értékeket, melyek adottak. A Lagrange-féle interpolációs polinom definíciója a következő [3]:

$$L_{n,f}(x) := \sum_{k=0}^n f(x_k) l_k(x), \quad \text{ahol } l_k := \prod_{j=0, j \neq k}^n \frac{x - x_j}{x_k - x_j} \quad (7)$$

A megértéshez tekintsünk egy egyszerű példát, melynek **Matlab**-kódja [4] a dokumentum mellékletében **lagrange_inter_pol_xmpl.m** néven megtalálható.

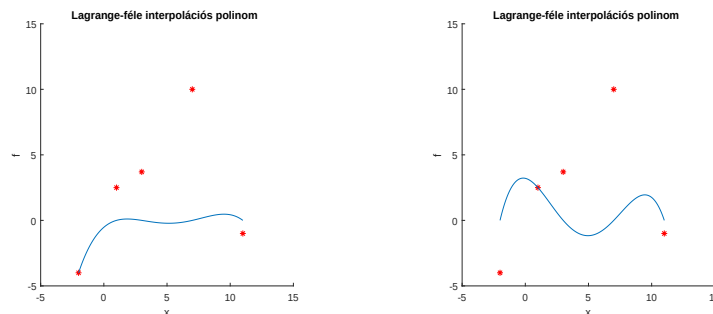
Adott 5 darab x -érték $(x_0, \dots, x_4) = (-2, 1, 3, 7, 11)$, valamint a hozzájuk tartozó 5 darab f függvényérték $(f(x_0), \dots, f(x_4)) = (-4, 2.5, 3.7, 10, -1)$. A Lagrange-féle interpolációs polinom ebben az esetben az

$$L_{4,f}(x) = f(x_0) l_0(x) + f(x_1) l_1(x) + \dots + f(x_4) l_4(x) \quad (8)$$

alakot ölti, ahol l_k értékeket kell még kifejtenünk az eredeti összefüggésnek megfelelően. Vizsgáljuk meg példaként l_0 alakját, a többi pedig teljesen analóg módon felírható:

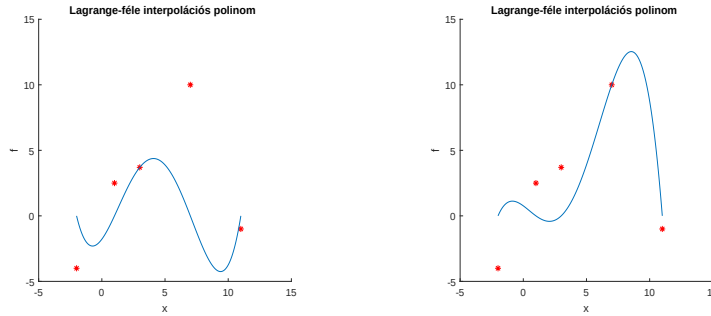
$$l_0(x) = \left(\frac{x-1}{-2-1} \right) \left(\frac{x-3}{-2-3} \right) \left(\frac{x-7}{-2-7} \right) \left(\frac{x-11}{-2-11} \right). \quad (9)$$

Az 1. ábrán az $f(x_0) l_0(x)$ és $f(x_1) l_1(x)$ alappolinomok láthatók.

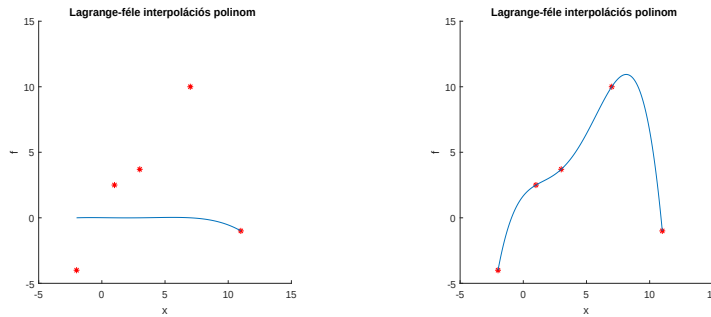


1. ábra. Alappolinomok: $f(x_0) l_0(x)$ és $f(x_1) l_1(x)$

Látható, hogy a polinom tökéletesen illeszkedik az $(x_0, f(x_0))$ és $(x_1, f(x_1))$, koordinátákra, míg a többire nem. A további alappolinomokat és a teljes Lagrange-féle interpolációs polinomot a 2. ábra és a 3. ábra mutatja.



2. ábra. Alappolinomok: $f(x_2)l_2(x)$ és $f(x_3)l_3(x)$



3. ábra. $f(x_4)l_4(x)$ és a teljes Lagrange-féle interpolációs polinom

Visszatérve az eredeti kérdésre: a súlyok meghatározásához helyettesítsük f -et az $x_i \in [a, b]$ alappontokra ($i = 0, 1, \dots, n$) támaszkodó Lagrange-féle interpolációs polinommal, majd integráljuk azt [1]. Így:

$$\int_a^b f(x) dx \approx \int_a^b L_{n,f}(x) dx \quad (10)$$

írható fel, amely kifejtve:

$$\int_a^b L_{n,f}(x) dx = \int_a^b \sum_{k=0}^n f(x_k) l_k(x) dx = \sum_{k=0}^n f(x_k) \int_a^b \prod_{j=0, j \neq k}^n \frac{x - x_j}{x_k - x_j} dx. \quad (11)$$

Összevetve (6) és (11) összefüggéseket látjuk, hogy c_i súlyok a következőképpen adódnak:

$$c_i = \int_a^b l_i(x) dx = \int_a^b \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j} dx. \quad (12)$$

Az olyan formulákat, melyekben a súlyokat (együtthatókat) (12) alapján számoljuk, interpolációs formuláknak nevezzük. Nem bizonyítom, de belátható [1], hogy az ilyen formulák, ha $n + 1$ pontra támaszkodnak, akkor *legalább* az n -edfokú polinomra pontosak.

Amennyiben a vizsgált tartomány felosztása ekvidisztáns, azaz $x_k = x_0 + k \cdot h$, ahol $h = \frac{b-a}{n}$ és $k = 0, 1, \dots, n$, *Newton-Cotes kvadratúráknak* nevezzük. A továbbiakban megismerkedünk néhány Newton-Cotes formulával.

5. Érintőformula

Az érintőformula a legegyszerűbb és legalapvetőbb numerikus integrálási formula. Lényege abban áll, hogy az eredeti $[a, b]$ intervallumot felbontjuk n darab egyenlő hosszúságú részintervallumra úgy, hogy

$$[a, b] = [x_0 = a, x_1 = x_0 + h] \cup [x_1, x_2 = x_1 + h] \cup \dots \cup [x_{n-1}, x_n = b]. \quad (13)$$

Az egyes részintervallumok tehát $[x_k, x_k + h]$ alakúak, ahol $k = 0, 1, \dots, n-1$. Az osztópontokat az intervallumok középpontjában vesszük fel, azaz általános esetben a $[x_k, x_k + h]$ intervallum osztópontja $\xi_k = x_k + \frac{h}{2}$. Az eredeti integrált a Lagrange-féle interpolációs polinommal közelítjük ezen intervallumok felett a következőképpen:

$$\int_{x_k}^{x_k+h} f(x) dx \approx f(\xi_k) \cdot \int_{x_k}^{x_k+h} l_0(x) dx. \quad (14)$$

Mivel intervallumonként egyetlen osztópontot vettünk fel, így az összeg is egytényezős lesz, l_0 értéke pedig 1, mert az üresszorzat értéke definíciószerűen 1. Így:

$$c_0 = \int_{x_k}^{x_k+h} 1 dx = [x]_{x_k}^{x_k+h} = h \quad (15)$$

adódik. Adott minden, most már fel lehet írni egyetlen intervallumra az érintőformula összefüggését:

$$E_k(f) := f(\xi_k) \cdot h, \quad (16)$$

és világos, hogy a függvény közelítő határozott integrálja ezen elemi szakaszok felett értelmezett határozott integrálok szuperpozíciója lesz:

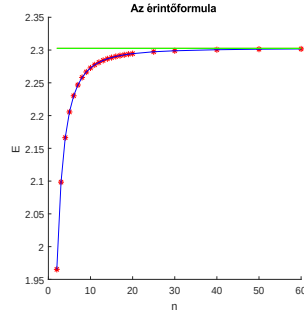
$$E := \sum_{k=0}^{n-1} E_k = h \sum_{k=0}^{n-1} f(\xi_k). \quad (17)$$

A formula `Matlab`-kódja a dokumentum mellékletében `erintoform_xmpl.m` néven megtalálható. A példában a formula működését az $f(x) = \frac{1}{x}$ függvényre vizsgáltam a $[2, 20]$ intervallum felett.

A példa segítségével a megvalósítás helyessége és a pontos eredményhez történő konvergencia is vizsgálható n felosztásfinomság függvényében, hiszen ezt a határozott integrált pontosan is ki tudjuk számolni:

$$\int_2^{20} \frac{1}{x} dx = [\ln |x|]_2^{20} = \ln 20 - \ln 2 = \ln 10 \approx 2.3026. \quad (18)$$

Megvizsgáltam, miként függ az érintőformula által kiszámolt közelítő integrál értéke n nagyságától, azaz a felosztás finomságától. Az eredmények a 4. ábrán láthatók, zöld aszimptotaként az integrál pontos értéke látható.



4. ábra. Az érintőformula által adott eredmények n függvényében

6. Trapézformula

Klasszikus numerikus integrálási formula az úgynevezett trapézformula. A formula lényege, hogy a vizsgált $[a, b]$ intervallumot – az érintőformulához teljesen hasonlóan – ismételten egyenlő, h hosszúságú diszjunkt intervallumokra bontjuk fel, mely szakaszok uniója kiadja az eredeti $[a, b]$ intervallumot, azaz

$$[a, b] = [x_0 = a, x_1 = x_0 + h] \cup [x_1, x_2 = x_1 + h] \cup \dots \cup [x_{n-1}, x_n = b]. \quad (19)$$

Amíg az érintőformula esetén a kvadrátúra pontjai az egyes intervallumok közepén helyezkedtek el, addig ezúttal minden egyes intervallumn két pontot veszünk fel, mégpedig ezek a pontok az egyes részintervallumok kezdő- és végpontjai lesznek. Formálisan: egy általános $[x_k, x_k + h]$ részintervallum kvadrátúrapontjai: $\xi_k = x_k$ és $\xi_{k+1} = x_k + h$.

Ebben az esetben az a feladatunk, hogy Lagrange-féle interpolációs polinomot fektessünk a $(\xi_k, f(\xi_k))$ és $(\xi_{k+1}, f(\xi_{k+1}))$ pontokra:

$$\int_{x_k}^{x_k+h} f(x) \, dx = \int_{\xi_k}^{\xi_{k+1}} f(x) \, dx \approx \sum_{i=0}^1 c_i f(x_i), \quad (20)$$

ahol a kvadrátúra súlyait a következőképpen számoljuk:

$$c_0 = \int_{\xi_k}^{\xi_{k+1}} l_0(x) \, dx, \quad c_1 = \int_{\xi_k}^{\xi_{k+1}} l_1(x) \, dx, \quad (21)$$

ahol

$$l_0 = \frac{x - x_2}{x_1 - x_2} = \frac{x - \xi_{k+1}}{\xi_k - \xi_{k+1}} \quad l_1 = \frac{x - x_1}{x_2 - x_1} = \frac{x - \xi_k}{\xi_{k+1} - \xi_k}. \quad (22)$$

Elvégezve az integrálásokat:

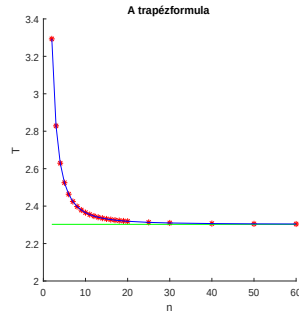
$$\begin{aligned}
c_0 &= \int_{\xi_k}^{\xi_{k+1}} \frac{x - \xi_{k+1}}{\xi_k - \xi_{k+1}} dx = \frac{1}{\xi_k - \xi_{k+1}} \int_{\xi_k}^{\xi_{k+1}} (x - \xi_{k+1}) dx = \\
&= \frac{1}{\xi_k - \xi_{k+1}} \left[\frac{x^2}{2} - \xi_{k+1}x \right]_{\xi_k}^{\xi_{k+1}} = \frac{1}{\xi_k - \xi_{k+1}} \left(\frac{\xi_{k+1}^2}{2} - \xi_{k+1}^2 - \frac{\xi_k^2}{2} + \xi_{k+1}\xi_k \right) = \\
&= \frac{-\frac{1}{2}(-\xi_{k+1}^2 + 2\xi_{k+1}^2 + \xi_k^2 - 2\xi_{k+1}\xi_k)}{\xi_k - \xi_{k+1}} = -\frac{1}{2}(\xi_k - \xi_{k+1}) = \frac{1}{2}(\xi_{k+1} - \xi_k).
\end{aligned}$$

$$\begin{aligned}
c_1 &= \int_{\xi_k}^{\xi_{k+1}} \frac{x - \xi_k}{\xi_{k+1} - \xi_k} dx = \frac{1}{\xi_{k+1} - \xi_k} \int_{\xi_k}^{\xi_{k+1}} (x - \xi_k) dx = \\
&= \frac{1}{\xi_{k+1} - \xi_k} \left[\frac{x^2}{2} - \xi_k x \right]_{\xi_k}^{\xi_{k+1}} = \frac{1}{\xi_{k+1} - \xi_k} \left(\frac{\xi_{k+1}^2}{2} - \xi_k \xi_{k+1} - \frac{\xi_k^2}{2} + \xi_k^2 \right) = \\
&= \frac{\frac{1}{2}(\xi_{k+1}^2 - 2\xi_{k+1}\xi_k - \xi_k^2 + 2\xi_k^2)}{\xi_{k+1} - \xi_k} = \frac{1}{2}(\xi_{k+1} - \xi_k).
\end{aligned}$$

A két súly értéke tehát megegyezik. A trapézformula így a következőképpen alakul:

$$\int_{\xi_k}^{\xi_{k+1}} f(x) dx \approx T(f) = \frac{\xi_{k+1} - \xi_k}{2} (f(\xi_k) + f(\xi_{k+1})). \quad (23)$$

A formula **Matlab**-kódja a dokumentum mellékletében **trapezform_xmpl.m** néven megtalálható. A példában a formula működését ismét az $f(x) = \frac{1}{x}$ függvényre vizsgáltam a $[2, 20]$ intervallum felett, hogy a konvergencia módját össze lehessen hasonlítani az érintőformula esetében tapasztaltakkal. Megvizsgáltam, miként függ a trapézformula segítségével kiszámolt közelítő integrál értéke n nagyságától, azaz a felosztás finomságától. Az eredmények az 5. ábrán láthatók, zöld aszimptotaként az integrál pontos értéke látható. Ezúttal megfigyehe-



5. ábra. A trapézformula által adott eredmények n függvényében

tő, hogy ez a formula speciálisan az $f(x) = \frac{1}{x}$ függvény esetében felső becslést ad az integrál pontos értékére, ellentétben a korábban bemutatott érintőformulával.

7. Simpson-formula

Szintén széleskörben elterjedt formula az úgynevezett Simpson-formula [1] is. A formula alkalmazásának lényege abban áll, hogy az eredeti $[a, b]$ intervallumunkat az eddig megszokott módon diszjunkt, egyenlő h hosszúságú részintervallumokra bontjuk, azaz

$$[a, b] = [x_0 = a, x_1 = x_0 + h] \cup [x_1, x_2 = x_1 + h] \cup \dots \cup [x_{n-1}, x_n = b], \quad (24)$$

de ez a kvadratúratípus már három pontra támaszkodik. Ezen kvadratúrapontok az egyes részintervallumok két szélén, valamint az intervallumok közepén helyezkednek el, azaz

$$\xi_k = x_k, \quad \xi_{k+1} = x_k + \frac{h}{2}, \quad \xi_{k+2} = x_{k+1}. \quad (25)$$

Az eddigiek alapján már teljesen világos, hogy a formulát az

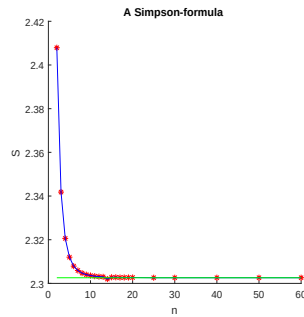
$$S(f) = c_0 f(\xi_k) + c_1 f(\xi_{k+1}) + c_2 f(\xi_{k+2}) \quad (26)$$

alakban keressük, ahol c_0, c_1 és c_2 a kvadratúra meghatározandó súlyai. Világos, hogy ismét a Lagrange-féle interpolációs polinomhoz kell nyúlni, tehát az egyes súlyokat a következőképpen lehet számolni:

$$c_0 = \int_{x_k}^{x_k+h} l_0(x) dx, \quad c_1 = \int_{x_k}^{x_k+h} l_1(x) dx, \quad c_2 = \int_{x_k}^{x_k+h} l_2(x) dx. \quad (27)$$

Mivel $l_0(x)$, $l_1(x)$ és $l_2(x)$ polinomfüggvények, integráljuk könnyedén kiszámolható a trapézformula esetén már ismertetett módon. A levezetést mellőzésével közlöm a kvadratúra súlyait [1]:

$$c_0 = \frac{(x_{k+1} - x_k)}{6}, \quad c_1 = \frac{4(x_{k+1} - x_k)}{6}, \quad c_2 = \frac{(x_{k+1} - x_k)}{6}. \quad (28)$$



6. ábra. A Simpson-formula által adott eredmények n függvényében

A Simpson-formula összefüggése tehát – egyváltozós esetben – a következőképpen alakul:

$$S(f) = \frac{(x_{k+1} - x_k)}{6} (f(\xi_k) + 4f(\xi_{k+1}) + f(\xi_{k+2})). \quad (29)$$

A formula `Matlab`-kódja a dokumentum mellékletében `simpson_xmpl.m` néven megtalálható. A példában a formula működését az eddig megszokott $f(x) = \frac{1}{x}$ függvényre vizsgáltam a $[2, 20]$ intervallum felett, hogy a konvergencia módját össze lehessen hasonlítani az eddigiekkel. Megvizsgáltam, miként függ a trapéz-formula segítségével kiszámolt közelítő integrál értéke n nagyságától, azaz a felosztás finomságától. Az eredmények a 6. ábrán láthatók, zöld aszimptotaként az integrál pontos értéke látható.

8. A Gauss-típusú kvadraturákról általában

Az eddig ismertett formulák során láttuk, hogy alkalmazásuk csak igen magas pontszám alkalmazása esetén ad megfelelően pontos értéket. Közös jellemzőjük továbbá, hogy csak a kvadratura súlyait választhattuk meg szabadon, az alappontjait nem. A Gauss-típusú kvadraturák alkalmazása során ezzel szemben mi magunk választhatjuk meg nemcsak a súlyokat, de az alappontokat is [1].

A Gauss-típusú kvadraturák pontos értéket adnak $2n - 1$ vagy ennél alacsonyabb fokú polinomok esetében az x_i alappontok és c_i súlyok megfelelő megválasztása esetén. ($i = 1, \dots, n$). Ennek a pontosságnak a megvalósításához az alappontoknak és a súlyoknak az alábbi feltételt kell kielégíteniük:

$$\sum_{k=1}^n c_k (x_k)^i = \int_a^b x^i dx, \quad i = 0, 1, \dots, 2n - 1. \quad (30)$$

Legyen

$$Q(f) := \int_a^b w(x) f(x) dx, \quad (31)$$

ahol $w : (a, b) \rightarrow \mathbb{R}$ folytonos, pozitív, valamint $\int_a^b w(x) dx$ létezik, w egy úgynevezett súlyfüggvény. Amennyiben feltesszük, hogy $[a, b] = [-1, 1]$, úgy w alakjától függően beszélhetünk például Gauss-Legendre-, vagy Gauss-Csebisev-kvadraturáról is.

Azt mondjuk, hogy az

$$\int_a^b w(x) f(x) dx \approx \sum_{k=1}^n c_k f(x_k) \quad (32)$$

formula *Gauss-kvadratura*, ha az integrál minden $p \in P_{2n-1}$ polinom esetén pontos.

Példaként tekintsük a Gauss-Legendre-kvadratura súlyainak és pontjainak meghatározását $[-1, 1]$ felett, $n = 2$ esetén. Gauss-Legendre kvadraturáról beszélünk akkor, ha a $w(x) = 1$ választással élünk. Ebben az esetben (30) alapján a következő feltételrendszerünk adódik c_1, c_2 súlyokra és x_1, x_2 pontokra:

$$\begin{cases} c_1 + c_2 = \int_{-1}^1 1 dx = [x]_{-1}^1 = 2, & c_1 x_1 + c_2 x_2 = \int_{-1}^1 x dx = \left[\frac{x^2}{2}\right]_{-1}^1 = 0, \\ c_1 x_1^2 + c_2 x_2^2 = \int_{-1}^1 x^2 dx = \left[\frac{x^3}{3}\right]_{-1}^1 = \frac{2}{3}, & c_1 x_1^3 + c_2 x_2^3 = \int_{-1}^1 x^3 dx = \left[\frac{x^4}{4}\right]_{-1}^1 = 0. \end{cases}$$

Kaptunk tehát egy – nemlineáris – egyenletrendszert négy egyenlettel és négy ismeretlennel. Az egyenletrendszer megoldható, könnyen ellenőrizhető, hogy

$$c_1 = 1, \quad c_2 = 1, \quad x_1 = -\frac{1}{\sqrt{3}}, \quad x_2 = \frac{1}{\sqrt{3}}. \quad (33)$$

megoldás. A kvadratúraformula tehát ebben a konkrét esetben:

$$\int_{-1}^1 f(x) \, dx \approx f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right). \quad (34)$$

Ez a formula alkalmas minden, legfeljebb harmadfokú polinom integráljának pontos meghatározására. Könnyen megmutathatjuk azt is, hogy negyedfokú polinomokra ez már nem igaz. Tekintsük például az $f(x) = x^4$ függvényt:

$$\int_{-1}^1 x^4 \, dx = \left[\frac{x^5}{5} \right]_{-1}^1 = \frac{2}{5}, \text{ de } f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right) = \frac{2}{9}.$$

Fontos felhívni az olvasó figyelmét, hogy a Gauss-típusú kvadratúrák elmélete ennél sokkal mélyebb és összetettebb. Projekt munkámnak nem célja, hogy feltárja a teljes elméleti hátteret. A cél inkább az, hogy megértsük, léteznek olyan módszerek, amelyek bizonyos előre meghatározott súlyok és pontok alkalmazásával képesek igen kis pont- és súlymennyiség mellett is sokkal pontosabb eredményt biztosítani, mint az eddig ismertetett formulák.

9. Példa Gauss-típusú kvadratúrára

A Gauss-típusú kvadratúráképletek középérték-képletek [2]:

$$\int_a^b f(x) \, dx \approx \sum_{\nu=1}^k c_\nu y_\nu, \text{ ahol } y_\nu = f(x_{g,\nu}). \quad (35)$$

A formulában c_ν a Gauss-kvadratúra súlyait, $x_{g,\nu}$ a pontjait, y_ν pedig az integrandus $x_{g,\nu}$ pontban értelmezett helyettesítési értékét jelöli. Miként azt már korábban említettük, ebben az esetben nem csak a c_ν együtthatókat, de az x_ν alappontokat is szabad paraméterként kezelhetjük úgy, hogy az integrálközelítő összefüggésünk minél magasabb fokszámú polinom esetén is pontos lehessen. A gyakorlati tapasztalatok azt mutatják, hogy a Gauss-típusú kvadratúráképletek többnyire rendkívül pontos közelítést adnak, de ehhez az alappontok igen speciális megválasztására van szükség.

Amennyiben az integrációs intervallumot úgy választjuk, hogy az $[a, b] = [-1, 1]$ legyen, az alappontokat pedig az úgynevezett Legendre-polinomok gyökei [2], akkor c_ν együtthatókat úgy lehet meghatározni, hogy a közelítő képletünk legfeljebb a $2n + 1$ -ed fokú polinomok integrálját pontosan adja. A hivatkozott Legendre-polinom gyökei az origóra szimmetrikusan helyezkednek el. A megfelelő súlyok és pontok számos internetesen oldalon elérhetőek (például [7]), de generálásukra találhatók számos programozási nyelvre megírt függvények is.

Ahhoz, hogy az itt definiált Gauss-kvadraúra segítségével tetszőleges $[a, b]$ intervallumra meg tudjuk határozni a függvények határozott integráljának közelítő értékét, egy egyszerű transzformációs eljárásra van szükség. Egyszerűen belátható, hogy a $t = \frac{b-a}{2}x + \frac{a+b}{2}$ transzformáció az $[-1, 1]$ intervallumnak az $[a, b]$ intervallumot felelteti meg, így az integrál közelítő értéke:

$$\int_a^b f(x) \, dx \approx \frac{b-a}{2} \sum_{\nu=1}^k c_\nu f\left(\frac{b-a}{2}x_{g,\nu} + \frac{a+b}{2}\right). \quad (36)$$

A kubatúra `Matlab`-kódja a dokumentum mellékletében `gausskvad_xmpl.m` néven megtalálható. A script bemeneti paramétereként szabadon megválasztható, hogy az hány ponttal és súllyal végezze el az integrál közelítését. A példában a formula működését ismét az $f(x) = \frac{1}{x}$ függvényre vizsgáltam a $[2, 20]$ intervallum felett, hogy az eddigiekkel összehasonlítható eredményt kaphassak. Azt tapasztaltam, hogy $n = 8$ súly és pont alkalmazása esetén a tényleges (2.3026) és a közelítő érték (2.3025) közötti eltérés (0.0038 %) jelentősen kisebbnek adódott, mint az eddig tapasztaltak.

10. Kubatúra egyszerű téglalap felett

A feladat a következő: legyen adott egy $f(x, y)$ kétváltozós függvény, amely értelmezve van egy S téglalap felett, ahol $|x| \leq h$ és $|y| \leq h$, teljesül továbbá minden olyan feltétel, amely lehetővé teszi, hogy az

$$\iint_S f(x, y) \, dx dy \quad (37)$$

integrál létezzen. Ekkor az $f(x, y)$ kétváltozós függvény S feletti integrálja meghatározható az

$$\iint_S f(x, y) \, dx dy = 4h^2 \left(\sum_{j=1}^n w_j f(x_j, y_j) + R \right) \quad (38)$$

formula segítségével, ahol w_j a kvadratúra súlyait, $x_j, y_j \in S$ a kvadratúra pontjait, R pedig a kvadratúra hibatagját jelöli [5].

Vizsgáljuk meg a kubatúra működését és megvalósítási módját. Legyen először

$$f(x, y) = x^2 + y^2, \quad S = [-2, 2]^2, \quad (39)$$

kérdés pedig:

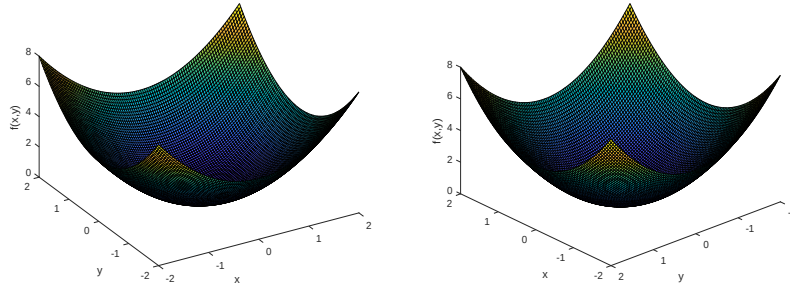
$$\iint_S f(x, y) \, dx dy = \int_{-2}^2 \int_{-2}^2 (x^2 + y^2) \, dx dy. \quad (40)$$

A kétváltozós függvény grafikonja a 7. ábrán látható. Mivel ez a kétváltozós függvény másodfokú, így az eddigiek alapján használhatjuk a négy pontot és négy súlyt alkalmazó kubatúra-formulát.

Az alkalmazott kubatúra súlyai és pontjai a következők [5, 7]:

$$w_1 = w_2 = w_3 = w_4 = \frac{1}{4}, \quad (x_i, y_i) = \left(\pm \frac{1}{3}\sqrt{3}h, \pm \frac{1}{3}\sqrt{3}h \right),$$

ahol h ebben az esetben S definíciója miatt 2. A kubatúra `Matlab`-kódja a dokumentum mellékletében `kubaturanegyp_xmpl.m` néven megtalálható. Mivel a választott kétváltozós függvény integrálja analitikus úton meghatározható (a levezetés mellőzésével az integrál pontos értéke: $\frac{128}{3} \approx 42.6667$), így könnyen tudjuk ellenőrizni az eredményt.



7. ábra. Az $f(x, y)$ függvény S felett

A script lefutattása után kapott eredményünk szintén 42.6667-re adódott, ami azt jelenti, hogy a kubatúra erre a függvény – mindösszesen négy pont és súly alkalmazásával – az alkalmazott számábrázolásunk mellett megfelelő eredményt adott.

11. Kubatúra tetszőleges csúcskoordinátájú téglalapok felett

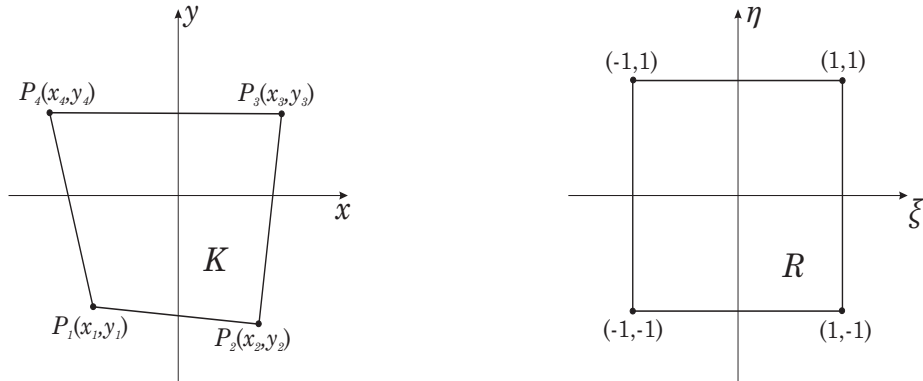
Felmerülhet a kérdés, hogy miként lehet tetszőleges csúcskoordinátákkal definiált téglalapok felett kvadratúra segítségével kiszámítani kétváltozós függvények határozott integrálját.

A következőkben ismertetett formula [6] alkalmas tetszőleges típusú és tetszőleges csúcspont-koordinátájú négyszög területe (K) felett értelmezett $f(x, y)$ kétváltozós függvény közelítésére:

$$I = \iint_K f(x, y) \, dxdy. \quad (41)$$

Ez úgy tehető meg, hogy az általános négyszöget egy szabályos, origó közép-ponttal rendelkező, 2 egység oldalhosszúságú négyzetté (R) szükséges transzformálni, amely felett már alkalmazható a kétdimenziós Gauss-kvadratúra. A transzformáció módját a 8. ábra mutatja.

Lényege, hogy az xy -síkon értelmezett eredeti négyszöget csomóponti formafüggvények segítségével a $\xi\eta$ -síkra kell transzformálni, majd a függvény integrálását ezen a síkon szükséges elvégezni. A csomóponti formafüggvények legfontosabb tulajdonsága, hogy i -edik sorszámú értéke az i -edik csomópontban 1, az összes többiben nulla. A transzformációhoz az alábbi négy formafüggvényt



8. ábra. Transzformáció a kvadrátúrához

alkalmaztam [6]:

$$\begin{aligned}
 N_1 &= \frac{1}{4} (1 - \xi) (1 - \eta), \\
 N_2 &= \frac{1}{4} (1 + \xi) (1 - \eta), \\
 N_3 &= \frac{1}{4} (1 + \xi) (1 + \eta), \\
 N_4 &= \frac{1}{4} (1 - \xi) (1 + \eta).
 \end{aligned} \tag{42}$$

A formafüggvények segítségével végrehajtott transzformációval az eredeti integrál kiszámítható az

$$\iint_K f(x, y) \, dx dy = \iint_R f(P(\xi, \eta), Q(\xi, \eta)) |\det J(\xi, \eta)| \, d\xi d\eta \tag{43}$$

összefüggés felhasználásával, ahol $P(\xi, \eta)$ az eredeti x , $Q(\xi, \eta)$ pedig az eredeti y változó helyettesítési értéke, kifejtésük a formafüggvények segítségével:

$$\begin{aligned}
 P(\xi, \eta) &= \sum_{i=1}^4 x_i N_i(\xi, \eta) = x_1 N_1(\xi, \eta) + x_2 N_2(\xi, \eta) + x_3 N_3(\xi, \eta) + x_4 N_4(\xi, \eta), \\
 Q(\xi, \eta) &= \sum_{i=1}^4 y_i N_i(\xi, \eta) = y_1 N_1(\xi, \eta) + y_2 N_2(\xi, \eta) + y_3 N_3(\xi, \eta) + y_4 N_4(\xi, \eta).
 \end{aligned} \tag{44}$$

A transzformáció Jacobi-mátrixának determinánsához szükség van x és y ξ és η szerinti deriváltjaira is, hiszen

$$\det J(\xi, \eta) = \left| \frac{\partial(x, y)}{\partial(\xi, \eta)} \right| = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{vmatrix} = \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial y}{\partial \xi} \frac{\partial x}{\partial \eta}. \tag{45}$$

A deriváltakat analitikus módon határoztam meg:

$$\begin{aligned}
\frac{\partial x}{\partial \xi} &= \frac{x_1}{4}(-1 + \eta) + \frac{x_2}{4}(1 - \eta) + \frac{x_3}{4}(1 + \eta) + \frac{x_4}{4}(-1 - \eta), \\
\frac{\partial x}{\partial \eta} &= \frac{x_1}{4}(-1 + \xi) + \frac{x_2}{4}(-1 - \xi) + \frac{x_3}{4}(1 + \xi) + \frac{x_4}{4}(1 - \xi), \\
\frac{\partial y}{\partial \xi} &= \frac{y_1}{4}(-1 + \eta) + \frac{y_2}{4}(1 - \eta) + \frac{y_3}{4}(1 + \eta) + \frac{y_4}{4}(-1 - \eta), \\
\frac{\partial y}{\partial \eta} &= \frac{y_1}{4}(-1 + \xi) + \frac{y_2}{4}(-1 - \xi) + \frac{y_3}{4}(1 + \xi) + \frac{y_4}{4}(1 - \xi).
\end{aligned} \tag{46}$$

A kvadratura összefüggése a transzformáció alkalmazásával tehát:

$$\sum_{i=1}^N \sum_{j=1}^N w_i w_j f(P(\xi_i, \eta_j), Q(\xi_i, \eta_j)) |\det J(\xi_i, \eta_j)|, \tag{47}$$

ahol w_i, w_j a kvadratura súlyait, ξ_i, η_j pedig a pontjait jelöli, melyek megtalálhatóak például a hivatkozott forráson [7] is.

Az itt bemutatott transzformációs kvadraturát `Matlab`-környezetben meg is valósítottam, a kódja a dokumentum utolsó mellékleteként megtalálható. Működését a következő kettős integrál kiszámításával teszteltem:

$$\int_{-3}^3 \int_6^{10} \frac{1}{\sqrt{x^2 + y^2}} dx dy. \tag{48}$$

Ezen határozott integrál öt tizedesre megadott, kerekített értéke a – bonyolult, de zárt alakban megadható – primitív függvény felhasználásával (a levezetés mellőzésével, pl. Wolfram Alpha segítségével könnyen ellenőrizhető) 2.99049. Amennyiben $n = 7$ súly és pont felhasználásával futtatjuk le a scriptet, úgy az integrál közelítő értéke öt tizedesjegyen 2.99049, azaz megegyezik a primitív függvény segítségével kiszámított értékkel. A Jacobi-determináns értéke $|J| = 6$ -ra adódott.

Megvizsgáltam azt is, miként változik ez az eredmény, ha az előző fejezetben ismertetett négyponos módszer segítségével próbáljuk meg közelíteni a határozott integrál értékét. Ebben az esetben az eredmény öt tizedesjegyen ábrázolva 2.99048, amely annak ellenére, hogy az adott számaábrázolás mellett $\varepsilon = 1 \cdot 10^{-5}$ hibát eredményez, gyakorlati szempontból szintén megfelelő eredménynek tekinthető.

12. A formulák hibájának vizsgálata

Egyszerű kvadraturaformuláknak nevezzük azokat a kvadraturaformulákat, melyek során az eredetileg vizsgált $[a, b]$ intervallum nem kerül feldarabolásra diszjunkt részintervallumokra, hanem a korábban ismertetett kvadraturaformulák kizárólagosan egyszer, a teljes $[a, b]$ intervallumra kerülnek felírásra, így közelítve a függvény határozott integrálját. Ezen egyszerű kvadraturaformulák hibaképlete a következőképpen alakul [1]:

Egyszerű érintőformula: Legyen $f : [a, b] \rightarrow \mathbb{R}$ kétszer folytonosan differenciálható. Ebben az esetben az egyszerű érintőformula hibaképlete a következő:

$$\int_a^b f(x) \, dx - E_e(f) = \frac{(b-a)^3}{24} \cdot f''(\xi), \quad \xi \in [a, b]. \quad (49)$$

Egyszerű trapézformula: Legyen $f : [a, b] \rightarrow \mathbb{R}$ kétszer folytonosan differenciálható. Ebben az esetben az egyszerű trapézformula hibaképlete a következő:

$$\int_a^b f(x) \, dx - T_e(f) = -\frac{(b-a)^3}{12} \cdot f''(\xi), \quad \xi \in [a, b]. \quad (50)$$

Egyszerű Simpson-formula: Legyen $f : [a, b] \rightarrow \mathbb{R}$ négyszer folytonosan differenciálható. Ebben az esetben az egyszerű Simpson-formula hibaképlete a következő:

$$\int_a^b f(x) \, dx - S_e(f) = -\frac{(b-a)^5}{2880} \cdot f^{(4)}(\xi), \quad \xi \in [a, b]. \quad (51)$$

A projektmunkámban ismertetett interpolációs kvadratúraformulák mindegyike *összetett kvadratúraformula* volt. Ezeket a kvadratúraformulákat az egyszerű formulák pontosítására használjuk, lényege a korábban ismertetett elméleti háttér szerint abban áll, hogy az eredeti intervallumot diszjunkt, egyenlő hosszúságú, n darab részintervallumokra bontjuk fel, melyek felett elvégezve az egyszerű kvadratúraformulák által adott számításokat, majd végül ezek eredményét összeadva pontosabb eredményt kaphatunk. Pontosságuk miatt világos, hogy a gyakorlatban ezen formulák az elterjedtebbek. A hibaképletek a következőképpen alakulnak [8]:

Összetett érintőformula: Legyen $f : [a, b] \rightarrow \mathbb{R}$ kétszer folytonosan differenciálható. Ebben az esetben az összetett érintőformula hibaképlete a következő:

$$\int_a^b f(x) \, dx - E(f) = \frac{(b-a)^3}{24n^2} \cdot f''(\xi), \quad \xi \in [a, b]. \quad (52)$$

Összetett trapézformula: Legyen $f : [a, b] \rightarrow \mathbb{R}$ kétszer folytonosan differenciálható. Ebben az esetben az összetett trapézformula hibaképlete a következő:

$$\int_a^b f(x) \, dx - T(f) = -\frac{(b-a)^3}{12n^2} \cdot f''(\xi), \quad \xi \in [a, b]. \quad (53)$$

Összetett Simpson-formula: Legyen $f : [a, b] \rightarrow \mathbb{R}$ négyszer folytonosan differenciálható. Ebben az esetben az összetett Simpson-formula hibaképlete a következő:

$$\int_a^b f(x) \, dx - S(f) = -\frac{(b-a)^5}{2880n^4} \cdot f^{(4)}(\xi), \quad \xi \in [a, b]. \quad (54)$$

Az egyes formulák által adott integrálértékeket n függvényében az 1. táblázat tartalmazza úgy, hogy n értékét kettő hatványaira választottam. Látjuk, hogy míg az érintőformula az integrál értékét alulról becsli, addig a trapéz- és a Simpson-formula felső becslést ad. Ez teljes mértékben összhangban van a hibaképletekkel, hiszen a tényleges integrál és a formula által adott érték különbsége az utóbbi két esetben negatív, míg az első esetben pozitív előjelű.

n	$E(f)$	$T(f)$	$S(f)$	n	E_h [%]	T_h [%]	S_h [%]
2	1.9653	3.2932	2.4079	2	14.6481	43.0218	4.5738
4	2.1663	2.6292	2.3206	4	5.9187	14.1847	0.7813
8	2.2582	2.3977	2.3047	8	1.9276	4.1307	0.0914
16	2.2902	2.3208	2.3028	16	0.5378	0.7910	0.0077
32	2.2994	2.3091	2.3026	32	0.1383	0.2829	< 0.0006
64	2.3018	2.3042	2.3026	64	0.0340	0.0701	< 0.0006
128	2.3024	2.3030	2.3026	128	0.0080	0.0180	< 0.0006
256	2.3025	2.3027	2.3026	256	0.0036	0.049	< 0.0006
512	2.3026	2.3026	2.3026	512	0.0006	0.0006	< 0.0006
1024	2.3026	2.3026	2.3026	1024	0.0006	0.0006	< 0.0006
2048	2.3026	2.3026	2.3026	2048	0.0006	0.0006	< 0.0006
4096	2.3026	2.3026	2.3026	4096	0.0006	0.0006	< 0.0006
8192	2.3026	2.3026	2.3026	8192	0.0006	0.0006	< 0.0006

1. táblázat. Az egyes kvadrátúraformulák által adott eredmények n függvényében

Látható, hogy a Simpson-formula segítségével a vizsgált példa esetén már 16 részszeresre történő felbontás esetén is megfelelő pontosságú eredmény érhető el, míg az érintő- és trapézformulák esetében hasonló pontosságú eredmény eléréséhez jelentősen finomabb felbontásra van szükség.

Hivatkozások

- [1] N. Pásztor. *Numerikus integrálás*. B.Sc. szakdolgozat, Eötvös Lóránd Tudományegyetem, Természettudományi Kar, Budapest, 2010.
- [2] Bronstein I. N., Szemengyaljev K. A. *Matematikai kézikönyv*. TypoT_EX kiadó, Budapest, 2009.
- [3] D. Deford. *Lagrange Interpolation*. Dartmouth College, 2015.
- [4] <https://www.mathworks.com/products/matlab.html> (utolsó látogatás: 2018. november 6).
- [5] <https://dlmf.nist.gov/3.5#x> (utolsó látogatás: 2018. november 8.)
- [6] Dr. Shaozhong D. *Quadrature Formulas in Two Dimensions*. Lecture Notes, Math 5172 - Finite Element Method, Section 001, Spring, University of North Carolina at Charlotte, 2010.
- [7] <https://pomax.github.io/bezierinfo/legendre-gauss.html> (utolsó látogatás: 2018. november 8.)
- [8] Zs. Varga. *Numerikus integrálás*. B.Sc. szakdolgozat, Eötvös Lóránd Tudományegyetem, Természettudományi Kar, Budapest, 2017.

0. Mellékletek: Matlab-kódok

0.1. lagrange_inter_pol_xmpl.m

```
%Lagrange-fele interpolacios polinom, pelda

clear; clc;

x_pontok = [-2 1 3 7 11]; f_pontok = [-4 2.5 3.7 10 -1]; felbontas = 200;

x = linspace(-2,11,felbontas); L = zeros(1,felbontas);

for k = 1:length(x_pontok)
    x_k = x_pontok(k); f_k = f_pontok(k);
    l = ones(1,felbontas);
    for j = 1:length(x_pontok)
        if j ~= k
            x_j = x_pontok(j);
            for n = 1:length(l)
                l(n) = l(n) * (x(n) - x_j) / (x_k - x_j);
            end
        end
    end
    L = L + (f_k .* l);
end
figure(k); hold on; plot(x_pontok,f_pontok,'red*'); xlim([-5 15]); ylim([-5 15]);
plot(x,L); xlabel('x'); ylabel('f');
axis square; title('Lagrange-féle interpoláció polinom');
```

0.2. erintoform_xmpl.m

```
%Erintoformula alkalmazasa, pelda

clear; clc;

%Felbontas (max.) finomsaga
n = 3;

a = 2; b = 20; h = (b-a)/n; I = log(b) - log(a);
x=linspace(a,b,200); f = 1 ./ x; %Egyszeru integralando: 1/x

figure(1); hold on; plot(x,f); xlim([a,b]); xlabel('x'); ylabel('y');

xi = zeros(1,n-1); x_pontok = zeros(1,n+1);
x_pontok(1) = a;

for i = 2:(n+1)
    x_pontok(i) = x_pontok(i-1) + h;
    xi(i-1) = x_pontok(i-1) + h/2;
end

E = 0;
for i = 1:length(xi)
    E = E + h * (1 / xi(i));
end
```

0.3. trapezform_xmpl.m

```
%Trapezformula alkalmazasa, pelda

clear; clc;

%Felbontas (max.) finomsaga
n = 3;

a = 2; b = 20; h = (b-a)/n; I = log(b) - log(a);
x=linspace(a,b,200); f = 1 ./ x; %Egyszeru integralando: 1/x

figure(1); hold on; plot(x,f); xlim([a,b]); xlabel('x'); ylabel('y');
```

```

xi = zeros(1,n-1); x_pontok = zeros(1,n+1);
x_pontok(1) = a;

for i = 2:(n+1)
    x_pontok(i) = x_pontok(i-1) + h;
end

T = 0;

for i = 1:(length(x_pontok)-1)
    T = T + ((x_pontok(i+1)-x_pontok(i)) / 2) * ...
        ( (1 / x_pontok(i)) + (1 / x_pontok(i+1)) );
end

```

0.4. simpson_xmpl.m

```

%Simpson-formula alkalmazasa, pelda

clear; clc;

%Felbontas (max.) finomsaga
n = 10;

a = 2; b = 20; h = (b-a)/n; I = log(b) - log(a);
x=linspace(a,b,200); f = 1 ./ x; %Egyszeru integralando: 1/x

figure(1); hold on; plot(x,f); xlim([a,b]); xlabel('x'); ylabel('y');

xi = zeros(1,2*n+1);

for i = 1:length(xi)
    if i == 1
        xi(i) = a;
    else
        xi(i) = xi(i-1) + 0.5*h;
    end
end

xi_k = xi(1); xi_kp1 = xi(2); xi_kp2 = xi(3);
S = 0;

S = S + ((xi_kp2 - xi_k) / 6) * ...
    ((1/xi_k)+4*(1/xi_kp1)+(1/xi_kp2));

for i = 2:n
    xi_k = xi(2*i-1);
    xi_kp1 = xi(2*i);
    xi_kp2 = xi(2*i+1);
    S = S + ((xi_kp2 - xi_k) / 6) * ...
        ((1/xi_k)+4*(1/xi_kp1)+(1/xi_kp2));
end

```

0.5. gausskvad_xmpl.m

```

%Gauss-kvadratura, 1D

clear; clc;

Gaussp = load('./Gaussp_1D.txt'); Gaussss = load('./Gaussss_1D.txt');

a = 2; b = 20; I = log(b) - log(a);
x=linspace(a,b,200); f = 1 ./ x; %Egyszeru integralando: 1/x

figure(1); hold on; plot(x,f); xlim([a,b]); xlabel('x'); ylabel('y');

G = 0; bmap2 = ((b-a)/2);

for i = 1:length(Gaussp)
    G = G + bmap2*Gaussss(i)*(1/(bmap2*Gaussp(i)+((a+b)/2)));
end

```

0.6. kubaturanegyp_xmpl.m

```

%Kubatura teglalap felett, negy ponttal pelda

```

```

clear; clc;

h = 2;
x = linspace(-h,h,100); y = linspace(h,-h,100); [X,Y] = meshgrid(x,y);
f = zeros(length(x));

for i = 1:length(x)
    for j = 1:length(y)
        f(i,j) = X(i,j)^2+Y(i,j)^2;
    end
end

figure(1); hold on; surf(X,Y,f); xlabel('x'); ylabel('y'); zlabel('f(x,y)');

I_negyp = 0; sulyok = [1/4 1/4 1/4 1/4];
pontok_x = [1/3*sqrt(3)*h -1/3*sqrt(3)*h 1/3*sqrt(3)*h -1/3*sqrt(3)*h];
pontok_y = [1/3*sqrt(3)*h -1/3*sqrt(3)*h -1/3*sqrt(3)*h 1/3*sqrt(3)*h];

for i = 1:length(pontok_x)
    I_negyp = I_negyp + 4*h^2*sulyok(i) * (pontok_x(i)^2+ ...
        pontok_y(i)^2);
end

```

0.7. kubatura_transzformacioval.m

```

%Kubatura transzformacioval, pelda

clear; clc;

Gaussp_x = load('./Gaussp_x.txt'); Gaussss = load('./Gaussss.txt');
Gaussp_y=load('./Gaussp_y.txt');

%Teglalap megadása a koordinataival
x = [6 10 10 6];
y = [-3 -3 3 3]; I = 0;

for o = 1:length(Gaussp_x)
    for p = 1:length(Gaussp_y)

        N_1 = 0.25*(1-Gaussp_x(p))*(1-Gaussp_y(o));
        N_2 = 0.25*(1+Gaussp_x(p))*(1-Gaussp_y(o));
        N_3 = 0.25*(1+Gaussp_x(p))*(1+Gaussp_y(o));
        N_4 = 0.25*(1-Gaussp_x(p))*(1+Gaussp_y(o));

        P = x(1)*N_1+x(2)*N_2+x(3)*N_3+x(4)*N_4;
        Q = y(1)*N_1+y(2)*N_2+y(3)*N_3+y(4)*N_4;

        dx_dkszi = 0.25*x(1)*(-1+Gaussp_y(o)) + 0.25*x(2)*(1-Gaussp_y(o)) + ...
            0.25*x(3)*(1+Gaussp_y(o)) + 0.25*x(4)*(-1-Gaussp_y(o));

        dx_deta = 0.25*x(1)*(-1+Gaussp_x(p)) + 0.25*x(2)*(-1-Gaussp_x(p)) + ...
            0.25*x(3)*(1+Gaussp_x(p)) + 0.25*x(4)*(1-Gaussp_x(p));

        dy_dkszi = 0.25*y(1)*(-1+Gaussp_y(o)) + 0.25*y(2)*(1-Gaussp_y(o)) + ...
            0.25*y(3)*(1+Gaussp_y(o)) + 0.25*y(4)*(-1-Gaussp_y(o));

        dy_deta = 0.25*y(1)*(-1+Gaussp_x(p)) + 0.25*y(2)*(-1-Gaussp_x(p)) + ...
            0.25*y(3)*(1+Gaussp_x(p)) + 0.25*y(4)*(1-Gaussp_x(p));

        det_jacobi = dx_dkszi*dy_deta - dy_dkszi*dx_deta;

        I = I + Gaussss(o)*Gaussss(p)* 1/sqrt(P^2+Q^2) * abs(det_jacobi);

    end
end

```