

**O‘ZBEKISTON RESPUBLIKASI
OLIY VA O‘RTA MAXSUS TA‘LIM
VAZIRLIGI**

M.Olimov T.Juraev S.Abdusalilov

**SONLI USULLAR VA
ALGORITMLAR**

*O‘zbekiston Respublikasi Oliy va o‘rta maxsus ta‘lim
vazirligi tomonidan oliy o‘quv yurtlari uchun Darslik
sifatida tavsiya etilgan*

**“Namangan” nashriyoti
2022**

UO‘K: 518.12

KBK: 22.19

I – 74

**M. Olimov, A. Juraev S.Abdujalilov Sonli usullar va algoritmlar. – N.:
“Namangan” nashriyoti, 2022, 188 b.**

Ushbu darslik bakalavriyatning Axborot tizimlari va texnologiyalari yo‘nalishlari talabalari uchun mo‘ljallangan bo‘lib, unda «Sonli usullar va algoritmlar» fanidan nazariy va amaliy mashg‘ulotlariga oid o‘quv materiallari hamda o‘qitishning innovasion texnologiyalari asosida darslarni tashkil etish bo‘yicha ishlanmalardan na‘munalar keltirilgan.

Darslikdan magistrantlar va ilmiy tadqiqotchilar ham foydalanishlari mumkin.

Taqrizchilar:

ISBN 978-9943-3285-8-2

© M.Olimov A.Juraev S.Abdujalilov
© “Mashrab” nashriyoti, 2022.

SO‘Z BOSHI

Kompyuterning qo‘llanilish sohalalaridan biri mexanik jarayonlarni va ob‘yektlarning matematik modellarini hisoblash usullari va kompyuterlarning dasturiy vositalari yordamida tadqiq etish bo‘lib qolmoqda. Hisoblash usullari va kompyuterlarning zamonaviy imkoniyatlari birgalikda mexanik jarayonlar va ob‘yektlarning shu paytgacha noma‘lum xususiyatlarini ochishga va, shu asnoda, axborot tizimlarini takomillashtirishga xizmat qilmoqda.

Shu bilan bir qatorda, yuqori texnologiyalar, fan va texnikaning rivojlangan bugungi kunida zamon talablariga javob beradigan, ta‘lim yo‘nalishlarining DTS va fan dasturlari asosida yozilgan mazkur darsliklarga talab juda kattadir.

Sonli usullar va kompyuterning ilmiy tadqiqot ishlarda qo‘llanilishiga bog‘liq bo‘lib, ilmiy va amaliy jihatdan dolzarbdir.

Ayniqsa, raqobatbardosh kadrlar tayyorlashda hisoblash matematikasi yutuqlaridan, zamonaviy axborot va kommunikasiya texnologiyalari, dasturlash tillari va vositalarini qo‘llab xalq ho‘jaligining turli-tuman masalalarini kompyuter yordamida yechishni o‘rgatishga bag‘ishlangan Mazkur Darslik larni yaratilishi katta ahamiyat kasb etadi.

Hozirgi kunda fan-texnika rivojlanib borgan sari matematika va kompyuterning o‘rni ortib bormoqda. Shu jumladan matematikadan fizika, mexanika, biologiya, kimyo va astronomiya hamda iqtisodiy masalalarni yechishda, mexanik jarayonlarni tahlil etishda va boshqa ko‘p sohalarda foydalaniladi. Bu sohalardagi jarayonlarning matematik modeli oddiy yoki xususiy hosilali differensial tenglamalar nomi bilan yuritiladi. Ushbu ishda issiqlik tarqalishi va taqsimoti hamda tebranish masalalari yuqori aniqlikdagi chekli ayirmalar usuli bilan sonli yechishga e‘tibor qaratilgan. Quyida ana shunga erishish uchun avvalo xususiy hosilali differensial tenglama, chegaraviy masala, ularning umumiy va xususiy yechimlari, ularni analitik usulda topish, qay hollarda matematik paketlardan va dasturlardan qanday foydalanish mumkinligi haqida so‘z

yuritildi. Xususiyl hosilali differensial tenglamalardan iborat bo'lgan bir qator mexanik jarayonlar modellari differensial tenglamalarini taqribiy yechish masalasi qaralib, taqribiy hisob usuli (chekli ayirmalar, to'rlar usuli) bo'yicha aniq amaliy masalalar matematik paketlar va dasturlar yordamida yechildi.

Mazkur Darslik o'zaro bir-birini to'ldiruvchi yettita bobdan iborat.

Darslikning birinchi bobida amaliy masalalarni Kompyuterda yechish bosqichlari to'liq tahlil qilib berilgan va masalalarni yechish jarayonida paydo bo'ladigan xatoliklar haqida batafsil ma'lumotlar keltirilgan.

Ikkinchi bobda amaliy hisob ishlarida juda ko'p uchrovchi chiziqli tenglamalar va ularning sistemalari, ularni taqribiy yechish usullari, usullarning geometrik ma'nolari, yechimlarning yaqinlashish shartlari va usullarning o'ziga xos tomonlari ochib berilgan.

Uchinchi bobda chiziqli algebraning masalalarini yechish bo'yicha to'g'ri va iteratsion usullar haqida ma'lumotlar berilgan. Chiziqli algebraik tenglamalar sistemasini yechishning Gauss usuli (bosh elementlarni tanlashga asoslangan), yuqori tartibli determinantlar hisobi va teskari matrisalarni topish yo'llari ko'rsatilgan.

To'rtinchi bobda interpolatsiyalash masalasiga to'xtalib o'tilgan. Interpolatsiya masalasi nima, uni amalda qo'llash qay tarzda kechadi degan savollarga javob izlangan. Lagranj va Nyuton interpolatsiya (ko'phadi) formulalari keltirilgan.

Beshinchi bobda aniq integrallar, ularning geometrik ma'nosi, ularni taqribiy hisoblash usullari bo'yicha batafsil ma'lumotlar berilgan.

Oltinchi bobda oddiy differensial tenglamalar, ularni umumiy va xususiyl yechimlari, boshlang'ich va chegaraviyl shartlar haqida tushunchalar berilgan. Koshi va chegaraviyl masalalar, ularni yechishning sonli-taqribiy va taqribiy-analitik usullari haqida ma'lumotlar keltirilgan.

Yettinchi bobda xususiy hosilali differensial tenglamalar (matematik-fizika tenglamalari)ni tiplarga ajratish, ularni yechishda to‘r usulini qo‘llash bo‘yicha ma’lumotlar berilgan.

Mazkur darslikda berilgan hisoblashning sonli usullari uchun ishlab chiqilgan algoritmlar blok-sxemalar tarzida dasturlashga qulay tarzda ifoda qilingan. Har bir usul uchun algoritm asosida Python dasturlash tili misolida standart dastur ta’minotlari yaratilgan. Ishlab chiqilgan algoritm va dasturning ishonchliligini baholash uchun ular yordamida «test misollari» yechib ko‘rsatilgan. Olingan natijalar batafsil tahlil qilib berilgan.

Mazkur darslikdan foydalanish jarayonida talabanning mustaqil fikrlash qobiliyatini rivojlantirish maqsadida muammoli vaziyat(MV)lar hosil qilingan. Shu bilan birgalikda, muammoli savol (MS) va muammoli topshiriq(MT)lar ham berilganki, ular yordamida talabani faollashtirish ko‘zda tutiladi.

Har bir bob yakuni talabanning mustaqil ishlari uchun nazariy va amaliy savol hamda topshiriqlar bilan jihozlangan. Mazkur darslikni tayyorlashda mashhur olimlar va taniqli professor-o‘qituvchilar tomonidan yaratilgan adabiyotlardan, shu jumladan: A.A. Samarskiy, N.S. Baxvalov, G.I. Marchuk, N.N. Kalitkin, B.N. Demidovich, V.Q. Qobulov, M.I. Isroilov, F.B. Badalov, A.A. Abduqodirov, G.Shodmonov, T.X. Xolmatov, A.Axmedov, N.Toyloqov, B.X. Xo‘jayorov, A. Boyzoqov, Sh.Qayumov, A.Siddiqov va boshqalarning Darsliklari va risolalaridan foydalanilgan.

1-BOB. HISOBLASH TAJRIBASI VA XATOLIKLAR

Hozirgi kunda tayyorlanayotgan bakalavrlarning matematik ma'lumoti oliy matematika fanida o'qitilayotgan an'anaviy bo'limlar bilan chegaralanib qolmasligi zarur. Ayniqsa, "Axborot tizimlari va texnologiyalari uchun" yo'nalishi bo'yicha ta'lim olayotgan talabalardan zamonaviy matematikaning zarur bo'limlarini bilishni, birinchi galda esa, hisoblash matematikasining zamonaviy usullarini mustahkam egallashni va ulardan amaliy masalalarni yechishda foydalanishni hamda yechilayotgan masalani dasturini yaratib, zarur sonli yechimni olishga erisha olishlari talab etiladi.

Yangi texnika va texnologiyaning keskin o'sib borishi matematika fanining zamonaviy bo'limlarini xalq ho'jaligi masalalarini yechishga yanada ko'proq qo'llanila boshlagani amaliy masalalarni yechishga ixtisoslashtirilgan bakalavrlar va magistrnlarni tayyorlashga bo'lgan talabni borgan sari orttirib bormoqda.

"Axborot tizimlari va texnologiyalari" yo'nalishi bo'yicha ta'lim olayotgan bakalavrlar amaliy masalalarni kompyuterda yechishlari uchun ikkita asosiy yo'nalish bo'yicha yetarlicha chuqur bilimga ega bo'lishlari kerak.

Birinchiidan, ular biror zamonaviy algoritmik tilda ma'lum algoritm asosida dastur tuzishni yoki matematik dasturlar asosida hisoblash jarayonini tashkil etishni bilishlari, ikkinchiidan amaliy masalalarni yechishning sonli-taqribiy usullari haqida ham yetarlicha bilimga ega bo'lishlari kerak. Mazkur Mazkur Darslik ham ana shu yo'nalishlar bo'yicha nazariy va amaliy bilimlar berishga mo'ljallab yozilgan.

1-§. Masalani kompyuterda yechish bosqichlari



Tayanch soʻz va atamalar

Matematik model, hisoblash tajribasi bosqichlari, amaliy masala, algoritm, dastur, sonli usullar, modul-dastur, amaliy dasturlar bogʻlami.

Zamonaviy hisoblash texnikasini unumli ishlatish taqribiy va sonli analiz usullaridan oqilona foydalanishsiz mumkin emas. Shuning uchun, rivojlangan chet el mamlakatlarida va davlatimizda hisoblash matematikasiga boʻlgan qiziqish keskin ortib bormoqda. Kompyuterlarning oxirgi paytlarda rivojlanib borishi esa sonli-taqribiy usullarning amalga tadbqiqiga keng istiqbol yaratdi.

Maʼlumki, amaliy jarayonlarni oʻrganish ularni ifodalovchi matematik modelni tuzishdan boshlanadi, yaʼni uning asosiy oʻziga xos xususiyatlari ajratiladi va ular oʻrtasida matematik munosabat oʻrnatiladi. Hayotda uchraydigan barcha jarayonlarning matematik modellarini tuzish mumkin. Bu modellar oʻrganilayotgan jarayonning asosiy xususiyatlarini oʻzida iloji boricha toʻlaroq, toʻkisroq mujassam qilishi kerak. Bu esa matematik modellarning ilojisiz murakkablashuviga sabab boʻladi. Bunday matematik modellarni ishlatish, ular asosida qaralayotgan jarayon koʻrsatkichlarining xususiyatlarini tasvirlovchi yechim olish ham oʻz navbatida murakkablashadi. Demak, izlanuvchi oldida bir-biriga zid ikki masala koʻndalang boʻladi: matematik modellar yetarli darajada mukammal va murakkab boʻlishi kerak, lekin bunday modellarni ishlatish qator qiyinchiliklarni ham keltirib chiqaradi.

Matematik model tuzilgach, yaʼni masala matematik koʻrinishda ifodalangach, uni maʼlum matematik usullar bilan tahlil qilish mumkin. Matematik modellarni tashkil qiluvchi algebraik, chiziqsiz, differensial, integral, integro-differensial va boshqa tenglamalarni yechish uchun matematika kurslarida keltirilayotgan aniq, analitik usullar faqat hususiy koʻrinishdagi, sodda tenglamalarning yechimini topish imkonini beradi, xolos. Shuning uchun, umumiyroq, murakkab tenglamalarning yechimlarini aniqlashga imkoniyat

yaratuvchi sonli-taqribiy usullarni o'rganish va ulardan amaliy masalalarni yechishda unumli foydalanish dolzarb masalalardan hisoblanadi.

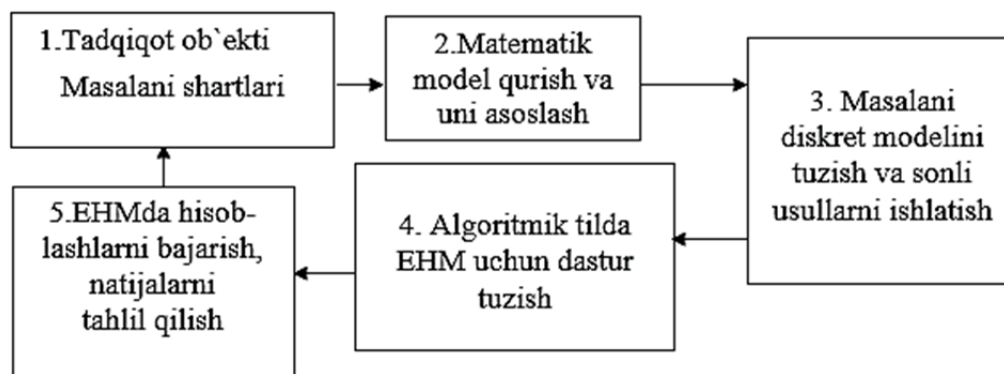
Matematik model hech qachon qaralayotgan ob'ektning xusu-siyatlarini aynan, to'la o'zida mujassam qilmaydi. U har xil faraz va cheklanishlar asosida tuzilgani uchun taqribiy xarakterga ega, demak uning asosida olinadigan natijalar ham taqribiy bo'ladi.

Modelning aniqligi, natijalarning ishonchlilik darajasini baholash masalasi matematik modellashirishning asosiy masalalaridan biridir.

Amaliy masalalarni xal qilish jarayoni turli xil vositalar yordamida ifodalanishi mumkin. Bu vositalar funksional analiz elementlarini ishlatib, differensial va integral tenglamalar tuzishdan to hisoblash algoritmi va kompyuter dasturlarini yozishgacha bo'lgan bosqichlarni o'z ichiga oladi. Har bir bosqich yakuniy natijaga o'ziga xos ta'sir ko'rsatadi va ulardagi yo'l qo'yiladigan xatoliklar oldingi bosqichlardagi xatoliklar bilan ham belgilanadi.

Ob'ektning matematik modelini tuzish, uni kompyuterda bajariladigan hisoblashlar asosida tahlil qilish "hisoblash tajribasi" deb ataladi.

"Hisoblash tajribasi"ning umumiy sxemasi 1-rasmda ko'rsatilgan.



1-rasm

Birinchi bosqichda masalaning aniq qo'yilishi, berilgan va izlanuvchi miqdorlar, ob'ektning matematik model tuzish uchun ishlatish lozim bo'lgan boshqa xususiyatlari tasvirlanadi.

Ikkinchi bosqichda fizik, mexanik, ximiyaviy va boshqa qonuniyatlar asosida matematik model tuziladi. U asosan algebraik, chiziqsiz, differensial, integral va

boshqa turdagi tenglamalardan iborat bo'ladi. Tizimda o'rganilayotgan jarayonga ta'sir ko'rsatuvchi omillarning barchasini bir vaqtning o'zida hisobga olib bo'lmaydi, chunki matematik model juda murakkablashib ketadi. Shuning uchun, model tuzishda eng kuchli ta'sir etuvchi asosiy omillargina hisobga olinadi.

Uchinchi bosqichda masalaga mos diskret modellar tuzish va tenglamalar yechish uchun usullar aniqlanishi lozim. Masalan, matematik model differensial tenglama bilan tasvirlangan bo'lsa, sonli usullar yordamida u chekli sondagi nuqtalarda aniqlangan chekli-ayirmali tenglamalar bilan almashtiriladi.

To'rtinchi bosqichda sonli usullar yordamida aniqlangan algoritm asosida biror - bir algoritmik tilda EHMda ishlatish uchun dastur tuziladi. Dasturni tuzish ham o'ta murakkab va ma'suliyatli jarayon bo'lib, bunda qator talablar hisobga olinishi lozim. Masalan, u umumiylik xususiyatiga ega bo'lishi kerak, ya'ni matematik modelda ifodalangan masala parametrlarining yetarlicha katta sohada o'zgaruvchi qiymatlarida dastur ishonchli natija berishi kerak.

Keyingi paytlarda dastur tuzmasdan masalalarni analitik yoki taqribiy yechish imkoniyatini beradigan matematik dasturlar (Mathcad, Maple, Matematika, Matlab kabi) ishlab chiqildi. Ularda qo'yilgan masalaning yechimini olish uchun buyruqlar (Maple, Matlab) yoki hisoblash jarayonini tashkil etuvchi formulalar (Mathcad) ketma-ketligidan foydalaniladi.

Oxirgi bosqichda mazkur dasturiy vositalar yordamida olingan natijalar bilan taqribiy usulda olingan sonli natijalar chuqur tahlil qilinib, baholanadi.

Natijalarga qarab, mutaxassis tahlil qilinayotgan jarayon to'g'risida xulosalar chiqaradi, uning amalga oshishiga ma'lum maqsad asosida ta'sir ko'rsatadi, jarayonni boshqarish vositalarini ishlab chiqadi, tavsiyalar beradi. Ko'plab variantlar asosida bajariluvchi hisoblash tajribalari yordamida loyihachi u yoki bu belgiga ko'ra barcha variantlar ichidan eng ma'qulini tanlashi mumkin. Shunday qilib, masalani yechuvchi o'z algoritmi va dasturini sinchiklab tekshirib chiqishi kerak. Aks xolda, Piter aytganidek: «Kompyuter hisoblovchining nochorligini ko'p martaga oshiradi», degan xodisa ro'y berishi mumkin.

1990 yillardan boshlab zamonaviy EHMLarning ishlab chiqilishi, ularni ilmiy va o'quv jarayonlariga kirib kelishi, ma'lum bir yutuqlardan tashqari ba'zi noqulayliklarni ham yuzaga keltirdi. Bu noqulaylik shaxsiy kompyuterlardan ilmiy, texnik va ijodiy masalalarni yechishda foydalanuvchilar uchun ancha sezilarli bo'ldi. Bunga asosiy sabab, shaxsiy kompyuterlarda katta kompyuterlar uchun yaratilgan tadbqiqiy masalalarni yechish uchun mo'ljallangan dasturlar kutubxonasini mavjud emasligidir. Shuning uchun, hozirda ana shu kamchilikni bartaraf qilish yo'lida turli xil izlanishlar olib borilmoqda. Shulardan biri sifatida ma'lum bir sinf masalalarini yechishga mo'ljallangan amaliy dasturlar bog'lamlari(ADB)ni yaratishni ko'rsatish mumkin.

Ma'lumki, biror jarayonni hisob ishlarini bajarib beruvchi standart dastur o'z ichiga bir necha modul-dasturni olishi mumkin. Bu modul - dasturlar aniq bir masalani yechishga mo'ljallangan bo'ladi. Unga beriladigan va undan olinadigan ma'lumotlar tiplari, ko'rinishlari oldindan aniqlanib qo'yiladi. Modul-dastur prosedura yoki prosedura-funksiya ko'rinishida aniqlanilib, kompilyasiya qilinadi va foydalanuvchi yaratayotgan umumiy dasturning bosh qismida unit fayllari ro'yhatiga kiritib qo'yiladi (biz dasturlash tili sifatida Python tilidan foydalanganligimiz uchun barcha ko'rsatmalar shu tilga nisbatan aytiladi). Shunday qilib, dasturchi o'zining dasturlar kutubxonasiga ega bo'ladi va bu dasturlardan istalgan masalani yechish dasturida foydalanishi mumkin.

ADBni menyu prinsipida ishlashini tashkil etish dasturdan foydalanish unumdorligini keskin orttiradi. Bu holda asosiy menyuda yechiladigan masalalar sinfi ko'rsatilsa (2-rasm), menyu osti menyusida esa mos ravishda sanab o'tilgan masalalarga mos yechish usullari tanlanadi (3-rasm).



2-rasm

Yechish usullari menyusida zarur boʻlgan usul tanlangach, shu usulga mos kompilyasiya qilingan fayl oʻz ishini davom ettiradi. Bu fayl standart holatda modulli prinsipda tuzilgan ishchi dasturni oʻz ichiga oladi. Faylni ishlashi uchun zarur maʼlumotlar berilgach, masalaning natijalari kompyuter ekraniga, printeriga yoki koʻrsatilgan yoʻl boʻyicha diskka yozilishi mumkin.

1. <i>Oraliqni teng ikkiga bo'lish usuli</i> 2. <i>Vatarlar usuli</i> 3. <i>Urinmalar usuli</i> 4. <i>Oddiy ketma-ketlik usuli</i> 5. <i>Yordam</i> 6. <i>Chiqish</i>	1. <i>Gauss usuli</i> 2. <i>Determinat hisobi</i> 3. <i>Teskari matrisa topish</i> 4. <i>Yordam</i> 5. <i>Chiqish</i>	1. <i>Oddiy iterasiya usuli</i> 2. <i>Nyuton usuli</i> 3. <i>Yordam</i> 4. <i>Chiqish</i>	1. <i>Nyuton interpolyasiion formulasi</i> 2. <i>Lagranj interpolyasiion formulasi</i> 3. <i>Eng kichik kvadratlar usuli</i> 4. <i>Yordam</i> 5. <i>Chiqish</i>
1. <i>Trapeziya usuli</i> 2. <i>To'g'ri to'rtbur-chaklar usuli</i> 3. <i>Parabolalar (Simpson) usuli</i> 4. <i>Yordam</i> 5. <i>Chiqish</i>	1. <i>Eyler usuli</i> 2. <i>Runge-Kutta usuli</i> 3. <i>Yordam</i> 4. <i>Chiqish</i>	1. <i>Chekli-ayirmala usuli</i> 2. <i>Galerkin usuli</i> 3. <i>Yordam</i> 4. <i>Chiqish</i>	1. <i>Elliptik tipdagi tenglamani yechish uchun to'r usuli</i> 2. <i>Parabolik tipdagi tenglamani yechish uchun to'r usuli</i> 3. <i>Giperbolik tipdagi tenglamani yechish uchun to'r usuli</i> 4. <i>Yordam</i> 5. <i>Chiqish</i>

3-rasm

Nazorat savollari

1. Matematik model nima?
2. Matematik model qanday koʻrinishlarda boʻladi?
3. Qanday usullarni sonli usullar deb ataymiz?
4. Hisoblash tajribasi qanday bosqichlardan iborat?
5. Amaliy dasturlar bogʻlami nima?

2-§. Xatoliklar va ularni baholash



Tayanch soʻz va atamalar

Xatolik, absolyut xatolik, nisbiy xatolik, xatoliklar ustida amallar, xatolik manbaalari, usulning xatoligi, boshlangʻich xatolik, bartaraf qilish mumkin boʻlmagan xatoliklar, hisoblash xatoliklari.

Taqribiy sonlar. Kompyuterda sonlar qoʻzgʻalmas va suzuvchi vergul shaklida tasvirlanadi. Haqiqiy sonlar cheksiz oʻnli kasrlardan iborat, lekin kompyuterning xotirasida chekli xonalarga ega boʻlgan sonlarga yozilishi mumkin. Shuning uchun, haqiqiy sonlar monitorda taqribiy tarzda tasvirlanadi.

Taqribiy sonlar ustida amallar bajarilganda xatolikni baholash katta ahamiyatga ega. Xatolik ikki xil boʻladi: absolyut va nisbiy xato. Absolyut xato sonning aniq va taqribiy qiymatlari orasidagi farqdan iboratdir, yaʼni agar $\bar{X}$ -biror sonning aniq, X esa uning taqribiy qiymati boʻlsa, absolyut xato $\Delta X = |X - \bar{X}|$ boʻladi. Nisbiy xato sonning absolyut xatosini uning taqribiy qiymatiga nisbatiga teng, yaʼni $\delta_X = \Delta X / |\bar{X}| \approx \Delta X / |X|$. Sonlarning aniq qiymati koʻp masalalarni yechishda nomaʼlum boʻladi. Shuning uchun, pirovard (chegara, limit) absolyut xato tushunchasi kiritiladi: u absolyut xatolarning yuqori chegarasidir, yaʼni $\Delta_X \geq \Delta X$. Sonning aniq qiymati quyidagi oraliqda boʻladi:

$$X - \Delta_X \leq \bar{X} \leq X + \Delta_X$$

Arifmetik amallar bajarishda absolyut va nisbiy xatolarning oʻzgarishini koʻrib chiqaylik.

Yigʻindi (ayirma) xatoligi. Ikkita $\bar{X} = X + \Delta_X, \bar{Y} = Y + \Delta_Y$ son berilgan boʻlsa, ularning yigʻindisi

$$\bar{X} + \bar{Y} = X + Y + \Delta_X + \Delta_Y$$

boʻladi. Yigʻindining absolyut xatoligi:

$$\Delta_{X+Y} = \Delta_X + \Delta_Y$$

Xuddi shunday, ayirmaning absolyut xatoligi quyidagicha bo'ladi:

$$\Delta_{X-Y} = \Delta_X + \Delta_Y$$

Ko'paytma xatoligi. Ko'paytmaning xatoligi quyidagicha topiladi:

$$\bar{X} \cdot \bar{Y} = (X + \Delta_X)(Y + \Delta_Y) = X \cdot Y + X \cdot \Delta_Y + Y \cdot \Delta_X + \Delta_X \cdot \Delta_Y$$

$\Delta_X \cdot \Delta_Y$ miqdor ikkinchi darajali kichik miqdordir, uni e'tiborga olmaymiz.

Demak, ko'paytmaning absolyut xatoligi quyidagicha bo'ladi:

$$\Delta_{XY} = X \cdot \Delta_Y + Y \cdot \Delta_X$$

Bo'linma xatoligi. Nisbatning absolyut xatosini topish uchun ushbu:

$$\frac{\bar{X}}{\bar{Y}} = \frac{X + \Delta_X}{Y + \Delta_Y} = \frac{X + \Delta_X}{|Y|(1 + \Delta_Y/|Y|)} = \frac{X + \Delta_X}{|Y|} \left[1 - \frac{\Delta_Y}{|Y|} + \left(\frac{\Delta_Y}{|Y|} \right)^2 - \dots \right]$$

almashtirishlarni bajaramiz. Bu yerda $\Delta Y/|Y| \ll 1$ ekanligidan foydalanib $(1 + \Delta Y/|Y|)^{-1}$ ifodani qatorga yoydik. Ikkinchi va undan yuqori darajali kichik miqdorlarni hisobga olmagan holda ushbu formulalarga ega bo'lamiz:

$$\frac{\bar{X}}{\bar{Y}} \approx \frac{X}{Y} + \frac{\Delta_X}{|Y|} - \frac{X}{|Y|^2} \Delta_Y \rightarrow \Delta_{X/Y} = \frac{\Delta_X}{|Y|} + \frac{X}{|Y|^2} \Delta_Y$$

Arifmetik amallarning nisbiy xatolar (δ) quyidagicha topiladi:

$$\delta_{X+Y} = \frac{\Delta_{X+Y}}{|X+Y|} = \frac{X}{|X+Y|} \cdot \frac{\Delta_X}{X} + \frac{Y}{|X+Y|} \cdot \frac{\Delta_Y}{Y} = \frac{X}{|X+Y|} \cdot \delta_X + \frac{Y}{|X+Y|} \cdot \delta_Y;$$

$$\delta_{X-Y} = \frac{\Delta_{X-Y}}{|X-Y|} = \frac{X}{|X-Y|} \cdot \delta_X + \frac{Y}{|X-Y|} \cdot \delta_Y;$$

$$\delta_{XY} = \frac{\Delta_{XY}}{|XY|} = \frac{X \cdot \Delta_Y}{|XY|} + \frac{Y \cdot \Delta_X}{|XY|} = \delta_X + \delta_Y;$$

$$\delta_{X/Y} = \frac{\Delta_{X/Y}}{|X/Y|} = \left(\frac{\Delta_X}{|Y|} + \frac{\Delta_Y}{|Y|^2} X \right) \frac{|Y|}{|X|} = \delta_X + \delta_Y$$

Yuqorida keltirilgan formulalar arifmetik amallar bajarishda yo'l qo'yiladigan absolyut va nisbiy xatolarni baholash imkoniyatini beradi.

Biror $u = u(x) = u(x_1, \dots, x_n)$ funksiyada $x = (x_1, \dots, x_n)$ argumentlar $\Delta_x = (\Delta_{x_1}, \dots, \Delta_{x_n})$ xatoliklar bilan berilgan bo'lsa, funksiyaning absolyut va nisbiy xatoliklari quyidagi ko'rinishda aniqlanadi:

$$\Delta_u = \sum_{i=1}^n \left| \frac{\partial u(x_1, \dots, x_n)}{\partial x_i} \right| \Delta_{x_i}, \quad (1)$$

$$\delta_u = \sum_{i=1}^n \left| \frac{\partial \ln u(x_1, \dots, x_n)}{\partial x_i} \right| \Delta_{x_i}. \quad (2)$$

Yuqoridagi barcha formulalar shu umumiy formulalardan keltirib chiqarilishi mumkin.

Misol 1. $u = x_1 + \dots + x_n$, $\partial u / \partial x_i = 1$, $\Delta_u = \Delta_{x_1} + \dots + \Delta_{x_n}$.

Misol 2. $u = x_1 \cdot \dots \cdot x_n$, $\partial \ln u / \partial x_i = 1/x_i$, $\delta_u = \delta_{x_1} + \dots + \delta_{x_n}$.

Izoh. Xatoliklarni hisoblashda, avval algebraik yig'indining absolyut xatosini, so'ng ko'paytmaning nisbiy xatosini hisoblash qulay.

Hisoblash xatoliklari. Masalani kompyuterda yechish jarayonida muayyan xatoliklarga yo'l qo'yish mumkin. Quyida ulardan ayrimlarini keltirib o'tamiz.

1. Bartaraf qilish mumkin bo'lmagan xatoliklar.

Bu xildagi xatoliklar masalani yechishda tuzilgan matematik modelda yo'l qo'yilgan taxminlar, farazlar va shuning oqibatida modelda paydo bo'lgan ayrim kamchiliklar va qusurlar bilan aniqlanadi. Masalan, matematik model unga kiruvchi o'zgaruvchilar va parametrlarning o'zgarish sohasining ma'lum bir qismida yaxshi natijalar berib, boshqa bir qismida esa yaroqsiz yechim berishi mumkin. Shuning uchun, matematik modelning "ishlash" sohasini topish masalani yechish bosqichlaridagi hal qilinishi lozim bo'lgan asosiy vazifalardan biridir.

Bartaraf qilish mumkin bo'lmagan xatoliklarga matematik modellarda ishlatiluvchi parametrlarning dastlabki berilgan qiymatlarining xatoliklari ham kiradi. Parametrlarning bu qiymatlarini har xil fizik, texnik, kimyoviy tajribalar, muhandislik izlanishlari asosida topiladi. Ayrim parametrlar esa dastlabki hisob-kitoblar orqali asoslanadiki, shu bosqichning o'zidayoq ularga hisoblash

xatoliklari qo'shiladi. Tajribalar aniqligini oshirib bu xatoliklarni kamaytirish mumkin, lekin ularni batamom bartaraf etib bo'lmaydi. Hisoblashlarda matematik modelda qatnashuvchi parametrlarning dastlabki qiymatlari bir-biriga yaqin tartibdagi xatoliklarga ega bo'lishiga erishish zarur. Chunki ma'lum parametrlarning juda yuqori tartibdagi aniqlik bilan olinishi yakuniy natijalarni ham shunday aniqlikda olishga hamma vaqt imkoniyat yaratmaydi.

2. Matematik usullarning xatoliklari.

Matematik modeldagi tenglamalarni hamma vaqt ham aniq usullar bilan yechib bo'lmaydi. Faqat, ayrim hususiy hollardagina buning imkoniyati mavjud. Lekin, olingan yechim ko'pincha juda murakkab ko'rinishda bo'ladi, ular asosida topilgan ko'rsatkichlarning son qiymatlarini kompyuterda hisoblash o'z navbatida oson masala emas. Bunday hollarda masala taqribiy usullar yordamida yechiladi. Tabiiyki, bunda aniq yechim emas, balki taqribiy yechim hosil qilinadi. Taqribiy usullarning asosini sonli usullar tashkil qiladi. Sonli usullarning aniqligini ma'lum darajada oshirish mumkin, lekin, bu usulning EHMda ishlashiga ketadigan vaqt miqdorini keskin ko'paytirib yuboradi. Sonli usul aniqligini o'ta oshirish hamma vaqt ham natijalarning aniqligini oshiravermaydi. Shuning uchun, sonli usullarning aniqligini matematik modelga kiruvchi parametrlar aniqligidan bir-ikki tartib yuqoriroq olish bilan cheklanish mumkin.

Sonli usullarga qo'yiladigan talablar. Matematik modeldagi tenglamalarni har xil sonli usullar bilan yechish mumkin. Lekin, hamma usullar ham kerakli aniqlikdagi yechimni beravermaydi. Ayniqsa, masala hozirgi zamon kompyuterlarida yechilganda hisoblash algoritmi turli, o'ziga xos shartlarni bajarishi kerak. Sonli usullarga qo'yiladigan talablar ikki guruhga bo'linadi. Birinchi guruhga sonli usullar qo'llanishi natijasida xosil qilingan diskret (uzuq-uzuq) masalaning matematik modeldagi dastlabki masalaga mos kelish shartlari kiradi.

Sonli usullarning yaqinlashishi, diskret masalalarda saqlanish qonunlarining bajarilishi, turg'unlik, korrektlik kabi talablar birinchi guruhga kiradi. Shulardan ayrimlarini qarab o'tamiz.

Matematik modeldagi parametrlarning dastlabki qiymatlaridagi xatolikni bartaraf etish mumkin bo'lmagan xatolik ekanligini yuqorida ko'rsatgan edik. Bu xatolikni masala yechimiga ko'rsatadigan ta'sir darajasini bilish katta ahamiyatga ega. Sonli usullarning bunday sezuvchanligini (ta'sirchanligini) turg'unlik degan tushuncha yordamida tekshirish mumkin.

Agar quyidagi shartlar bajarilsa, masala korrekt qo'yilgan deyiladi: 1)yechim mavjud; 2)yagona; 3)turg'un. Ko'rsatilgan shartlardan birortasi bajarilmasa, masala korrekt qo'yilmagan deyiladi. Bunday masalalarga sonli usullarni qo'llash foydasizdir, chunki bunda yetarli darajadagi shartlarni qanoatlantiruvchi sifatli yechimni olish imkoniyati yo'qdir. Shuni ham aytish kerakki, ayrim korrekt qo'yilmagan masalalarni yechish usullari ham yaratilgan. Bu usullar dastlabki qo'yilgan masalani emas, unga korrekt qilib qo'yilgan yordamchi masalani yechishga asoslangandir.

Yordamchi masalada qo'shimcha α parametr qatnashadi. Shunday yo'l bilan dastlabki masala regulyarlashtiriladi. Agar $\alpha \rightarrow 0$ bo'lsa, yordamchi masalaning yechimi dastlabki masalaning yechimiga intilishi kerak.

Yuqoridagiga o'xshash sonli usullarning korrektilik tushunchasi kiritilgan. Agar masaladagi parametrlarning barcha qiymatlarida sonli yechim mavjud, yagona va turg'un bo'lsa, u korrekt deyiladi.

Sonli usullar bilan topilgan yechim masalaning haqiqiy yechimiga yaqin bo'lishi kerak. Buni sonli usullarning yaqinlashishi tushunchasi yordamida tahlil qilishimiz mumkin. Diskretlashgan masalalar misolida yaqinlashish tushunchasini quyidagicha berishimiz mumkin. Agar diskretlashtirilgan masalaning yechimi diskretlashtirish parametri nolga intilganda dastlabki uzluksiz masalaning yechimiga intilsa, sonli usul yaqinlashadi deyiladi.

Sonli usullar ichida eng ko'p ishlatiladiganlari ayirmali usullardir. Bu usullar yordamida uzluksiz matematik modellardan diskret modellar xosil qilinadi. Buning uchun, masala qaralayotgan soha diskret nuqtalar majmuasi-to'r bilan almashtiriladi, tenglamadagi, chegaraviy va boshlang'ich shartlardagi xossalardan chekli ayirmalarga o'tiladi. Natijada, to'rning tugun nuqtalarida aniqlangan

funksiyalarga nisbatan algebraik tenglamalar sistemasi xosil qilinadi. Ma'lumki, matematik modellar asosida yotuvchi tenglamalar aksariyat hollarda fizika, mexanikadagi saqlanish qonunlari asosida tuziladi. Bu qonunlar matematik modeldagi tenglamalar diskret tenglamalar-cheqli ayirmali sxemalar bilan almashtirilganda ham bajarilishi kerak.

Bunday chekli ayirmali sxemalarga konservativ sxemalar deyiladi. Konservativ sxemalar tenglamalar yechimini fizik nuqtai-nazardan to'g'ri olish imkoniyatini beradi. Shuning uchun, chekli ayirmali sxemalarning konservativlik sharti masalalar yechishda boshqa shartlar qatori tekshirilishi kerak.

Sonli usullarga qo'yiladigan talablarning ikkinchi guruhini diskret modelni kompyuterda o'tkazish imkoniyatlari tashkil qiladi. Sonli usullar shunday algoritmlarga olib kelishi kerakki, kompyuterning xotira qurilmasi ular uchun yetarli bo'lishi va hisob-kitob vaqti iloji boricha kam bo'lishi lozim. Hisoblash algoritmlari yetarli samaradorlikka ega bo'lishi uchun algoritmdagi arifmetik va mantiqiy amallar soni iloji boricha kam bo'lib, xotira qurilmasida kam hajmni egallashi kerak.



Nazorat savollari

1. Xatolikning qanday turlari mavjud?
2. Absolyut xatolik nima?
3. Nisbiy xatolik nima?
4. Xatolikning manbalari qaerda?
5. Sonli usullarga qanday talablar qo'yiladi?

XULOSA

Mazkur bobda matematik model tushunchasi, uni amaliy masalalarni yechishdagi o'rni, modelni tuzishdagi asosiy omillar, modelning aniqligi, uning ishonchlilik darajasini baholash haqida mulohazalar keltirildi.

Hisoblash tajribasi va uning bosqichlari strukturasi tavsiya qilinib, har bir bosqichning vazifasi tahlil etildi.

Modul- dastur tushunchasi, dasturlar kutubxonasi, amaliy dastur ta'minoti, asosiy menyular haqida umumiy ma'lumotlar berilib, amaliy masalalarni yechishda qo'llanadigan sonli usullarning turli xil guruhlari strukturasi menyu osti bo'limlari sifatida shakllantirildi.

Xatoliklar, ularning turlari, yig'indi xatoligi, ko'paytma va bo'linma xatoliklarini baholashda qo'llanadigan formulalar keltirildi.

Bartaraf qilish mumkin bo'lmagan xatoliklar va matematik usullarning xatoliklari haqida mulohazalar yuritildi.

Sonli usullarda topilgan yechimning haqiqiy yechimga yaqinlashish masalasi, masalani korrektligi, diskret modelni kompyuterda o'tkazish imkoniyatlari tahlil etildi.



Bobga doir muammoli vaziyatlar!

-  Matematik modelni tuzishda izlanuvchi o'rganilayotgan jarayonning barcha xususiyatlarini xisobga olish imkoniga egami? Bu matematik modelga qanday ta'sir qiladi?
-  Matematik modelni yechishda olingan natijalarning taqribiy bo'lishiga sabab bo'luvchi boshlang'ich omilni ko'rsata olasizmi?
-  Matematik modellashtirishda modelning aniqligi va natijalarning ishonchlilik darajasini baholash masalasi qanday hal etiladi?
-  Masalani yechish bosqichlarini ifodalovchi ketma-ketlikni o'zgartirish uchun takliflaringiz bormi? Aksincha bo'lsa nima uchun?
-  Algoritmalar asosida dastur tuzish masalasida qaysi talablarga rioya etish zarur deb hisoblaysiz? Dastur ishonchli natija berishi uchun qanday tavsiyalar berasiz?
-  Amaliy masalalarning matematik modellarini sonli usullar bilan hisoblash zururiyati qaerdan kelib chiqadi?

- ✚ Haqiqiy sonlarni, xususan “juda katta” va “juda kichik” sonlarni monitorda aks ettirish imkoniyatini tushuntira olasizmi?
- ✚ “Hayotda barcha o‘lchov natijalari nisbiydi” degan fikrga Siz qanday qaraysiz? Fikringizni izohlang.
- ✚ Bartaraf qilish mumkin bo‘lmagan xatoliklar haqida nima deya olasiz. Ularning asosiy manbaalari qaerda deb o‘ylaysiz?
- ✚ Qo‘yilgan masalani yechishda sonli usullarning aniqligini juda oshirish imkoni tug‘ildi deylik, undan to‘liq foydalanasizmi? Bu ijobiy samara beradi deb o‘ylaysizmi?
- ✚ Sonli usullarni qo‘llashda kerakli aniqlikdagi yechimni olish uchun qanday talablarga rioya etishni tavsiya qilasiz? Bu talablarni bajarmaslik qanday oqibatlarga olib keliishi mumkin?
- ✚ Masala korrekt qo‘yilmaganligini qachon aniqlash mumkin, uni “yaxshilash” bo‘yicha yo‘l-yo‘riqlar bera olasizmi?

2-BOB. CHIZIQSIZ TENGLAMA VA TENGLAMALAR SISTEMASINI TAQRIBIY YECHISH

Fizika, mexanika, texnika va tabiatshunoslikning xilma-xil masalalari chiziqli bo‘lmagan tenglamalarni yechishga olib keladi. Masalan, noma‘lumlarni yo‘qotish yo‘li bilan murakkab algebraik va geometrik munosabatlar ikkinchi yoki yuqori darajali algebraik tenglamalarga keltiriladi. Shu bois, chiziqli bo‘lmagan tenglamalarni va tenglamalar sistemasini yechish matematik analizning muhim masalalaridan hisoblanadi. Ushbu bobda chiziqli bo‘lmagan har qanday tenglama va tenglamalar sistemasini taqribiy yechish imkoniyatlari tahlil etiladi, bir nechta taqribiy usullar va ularning geometrik ma‘nolari, mohiyati tushuntirilib, usulga mos ishchi algoritmlar va dastur ta‘minotlari tavsiya qilinadi.

1-§. Umumiy tushunchalar

Tayanch so‘z va atamalar



Tenglama, chiziqsiz tenglama, transsendent tenglama, tenglamaning yechimi, ildizning mavjudlik sharti, oraliqni ajratish usullari, grafik usul, analitik usul, algoritmik usul .

Chiziqli bo‘lmagan tenglamalarni umumiy holda quyidagi shaklda ifodalash mumkin:

$$f(x) = 0 \quad (2.1)$$

Chiziqli bo‘lmagan tenglamalarni ikki xilga bo‘lish mumkin: algebraik va transsendent. Algebraik tenglamalar deb algebraik (butun, rasional, irrasional) funksiyalardan tashkil topgan tenglamalarga aytiladi. Agar tenglamada boshqa funksiyalar (trigonometrik, ko‘rsatkichli, logarifmlik va h.k) qatnashsa, bunday tenglamaga transsendent tenglama deyiladi.

Tenglamaning yechimi deb x noma‘lumning shunday qiymat-lariga aytiladiki, ularni (2.1) tenglamaga qo‘yganda, tenglama qanoatlantiriladi. Lekin,

amalda bunday tenglamalarning aniq yechimlarini topish juda qiyin yoki umuman mumkin emas.

Bunday hollarda, yechimni taqribiy qiymatini topishga imkon beruvchi taqribiy hisoblash usullari qo'llaniladi. Chiziqsiz tenglamalarni yechish usullari ikkita guruhga bo'linadi: aniq (to'g'ri) va iteratsion (taqribiy) usullar. Aniq usul yordamida tenglamaning yechimi formulalar orqali aniqlanadi. Masalan, kvadrat tenglamaning yechimini topishni shu usulga misol sifatida ko'rsatish mumkin:

$ax^2 + bx + c = 0$ -chiziqsiz tenglamani yechimlari Viet formulalari orqali beriladi (Kordano, Ferrari formulalari):

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

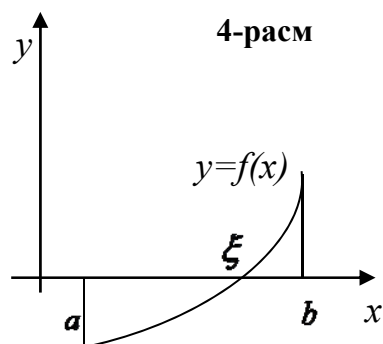
Lekin, bunday formulalar 3-, 4-darajali algebraik tenglamalar uchun mavjud xolos.

Taqribiy yechish uchun qo'llaniladigan ko'pgina usullarda tenglamaning ildizlari ajratilgan, ya'ni shunday kichik atrofchalar topilganki, bu atrofchalarda tenglamaning bittagina ildizi joylashadi, deb faraz qilinadi. Bu atrofning biror nuqtasini dastlabki yaqinlashish sifatida qabul qilib, taqribiy usullardan birortasini qo'llab, izlanayotgan yechimni berilgan aniqlik bilan hisoblash mumkin. Demak, chiziqsiz tenglamani taqribiy yechish ikki bosqichda olib boriladi:

Ildizni ajratish, ya'ni iloji boricha shunday kichik oraliq olinadiki, natijada shu oraliqda tenglamani bitta va faqat bitta haqiqiy ildizi mavjud bo'lsin.

Dastlabki yaqinlashish ma'lum bo'lsa, ildizni berilgan aniqlik bilan hisoblash. Masalaning birinchi qismi ikkinchisiga qaraganda ancha murakkabdir. Chunki umumiy holda ildizni ajratishning samarali usuli mavjud emas.

Quyidagi teoremlar ildiz yotgan oraliqlarni ajratishga yordam beradi:



1-teorema: Agar uzluksiz $f(x)$ funksiya biror (a, b) oraliqning chetki nuqtalarida har xil ishorali qiymatlarni qabul qilsa, u vaqtda bu oraliqda (2.1) tenglamaning hech bo'lmaganda bitta haqiqiy ildizi mavjuddir. Ya'ni, shunday ξ son $\xi \in (a, b)$

topiladiki, $f(\xi) = 0$ bo'ladi (4- rasm).

Agar shu bilan birga, birinchi tartibli hosila $f'(x)$ mavjud bo'lib, u o'zining ishorasini shu oraliqda saqlasa, u vaqtda bu oraliqda olingan ildiz yagonadir.

2-teorema: $f(x)$ funksiya (a, b) oraliqning chetki nuqtalarida har xil ishorali qiymatlarni qabul qilsa, u vaqtda tenglamani a va b nuqtalar orasida yotadigan ildizlar soni toqdir.

Agar $f(x)$ funksiya oraliqning chetki nuqtalarida bir xil ishorali qiymatlarni qabul qilsa, u vaqtda tenglama ildizi oraliqda mavjud emas yoki ularning soni juftdir.

Ildizlarni ajratishning turli usullari mavjud. Amalda analitik, grafik va algoritmik usullardan keng foydalaniladi. Ularni qisqacha tavsiflaymiz:

Analitik usul- bunda $f(x)$ funksiyaning ishorasi o'zgaradigan oraliqlari topiladi. Albatta, $f'(x) = 0$ tenglama yordamida. Bu oraliqlarda tenglamaning yagona ildizlari yotadi.

Algoritmik usul- bunda ildiz aniqlanadigan kesma uzunligi $[a, b]$ iloji boricha kattaroq qilib tanlab olinadi. Oraliqqa tegishli har bir kichik $[x_i, x_{i+1}]$ kesmalarda funksiya ishoralari o'zgaradigan oraliqlar va ularning soni aniqlanadi. Har safar $f(x_i) \cdot f(x_{i+1}) < 0$ sharti tekshiriladi. Agar shart bajarilmasa, navbatdagi kesma tekshirib borilaveradi. Bu jarayon kes-malar $[a, b]$ oraliqni to'liq qoplab olmagunicha davom ettiriladi. Bunda topilgan oraliqlarda ildizning yagonaligiga ham, ba'zi bir ildizlarni aniqlanmay qolishligiga ham asos bor. Chunki, $[a, b]$ yetarlicha katta bo'lganda funksiya ishoralari har xil bo'lgan oraliqda u absissa o'qini bir necha marta kesib o'tgan ham, aslida ishora o'zgarganu, lekin oraliq chetlarida bir xil ishorali bo'lib qolgan va ildizi yo'qotilgan bo'lishi mumkin.

Shuning uchun, olingan natijalarni tekshirish maqsadida ularni $[a, b]$ ning har xil qiymatlarida olib ko'rish maqsadga muvofiqdir. Agar natijalar barcha holda takrorlansa ularni haqiqatga yaqin deb hisoblash mumkin.

Grafik usul-bu usul haqiqiy ildizni ajratishda katta yordam beradi. Buning uchun, $y = f(x)$ funksiyaning grafigini taqribiy ravishda chizib olamiz. Grafikning OX o'qi bilan kesishgan nuqtalarining absissalari ildizning taqribiy qiymatlari deb olinadi. Agar $f(x)$ ning ko'rinishi murakkab bo'lib, uning grafigini chizish qiyin bo'lsa, u vaqtda grafik usulni boshqacha tarzda qo'llash kerak. Buning uchun, $f(x) = 0$ tenglamani unga teng kuchli bo'lgan $f_1(x) = f_2(x)$ ko'rinishda tasvirlanadi. Keyin $f_1(x)$ va $f_2(x)$ funksiyalarning grafiklari alohida-alohida chizilib, ikkala grafikning kesishish nuqtalari topiladi. Bu nuqtalarning absissalari ildizlarning taqribiy qiymatlari deb qabul qilinadi. Shunday qilib, taqribiy yagona ildiz yotgan $[a, b]$ kesmani haqiqatda to'g'ri olinganligini analitik yo'l bilan tekshirib ko'rish mumkin. Buning uchun, yana ildizning mavjudlik sharti $f(a) \cdot f(b) < 0$ dan foydalanamiz. Agar shart bajarilsa oraliq to'g'ri tanlangan bo'ladi.



Oraliqni tekshirish jarayonida $f(a)f(b) = 0$ sharti bajarilsa, siz bu muammo vaziyatda qanday xulosalarga kelasiz?

Oraliqni grafik usulda ajratish jarayonini misol bilan tushuntiramiz.

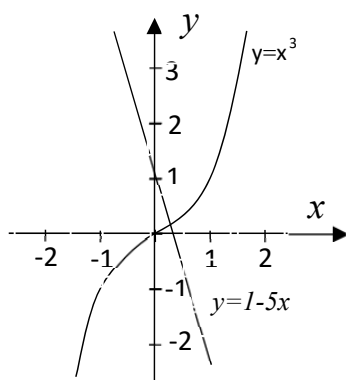
Misol. Ushbu

$$f(x) = x^3 + 5x - 1$$

tenglamaning taqribiy ildizi yotgan oraliqni ajrating.

Yechish. Buning uchun $f_1(x) = x^3$ va $f_2(x) = 1 - 5x$ funksiyalarning grafigini chizib olamiz (5-rasm).

5-rasm



Grafikdan ko‘rinib turibdiki, chiziqsiz tenglama faqat bitta ildizga ega va u $[0,1]$ oraliqda bo‘lishi mumkin. Chunki $x=0$ va $x=1$ nuqtalarda $f(x)$ funksiya har xil ishorali qiymatlarga ega: $f(0)=-1<0$, $f(1)=5>0$. Demak, ildiz $[0,1]$ kesmada yotadi.

Oraliq aniqlangach, turli usullardan birini ishlatib, kerakli aniqlikdagi yechimni olish mumkin.

Algebraik va transsendent tenglamalarni taqribiy yechishda yo‘l qo‘yiladigan xatoni umumiy holda baholashda quyidagi teoremdan foydalanamiz:

3-teorema: Agar (a,b) kesmada ξ soni $f(x)=0$ tenglamaning aniq, x esa taqribiy yechimi va ularning ikkalasi ham $a \leq x \leq b$ kesmada joylashgan bo‘lib,

$$|f'(x)| \geq m_1 > 0 \text{ bo'lsa, u holda quyidagi baho o'rinlidir. } |x - \xi| \leq \frac{f(x)}{m_1}.$$

Keyingi mavzularda chiziqsiz tenglamalarni yechishning iteratsion usullarining ayrimlari batafsilroq keltiriladi.



Nazorat savollari

1. Qanday tenglamani chiziqsiz tenglama deb ataladi?
2. Chiziqsiz tenglamaning nechta yechimi mavjud?
3. Chiziqsiz tenglamani yechishda oraliqni qanday ajratiladi?
4. Oraliqni ajratishning grafik usulini tushuntirib bering.
5. Oraliqni ajratishning analitik usulida qaysi formula qo‘llaniladi ?
6. Algebraik va transsendent tenglamalarni taqribiy yechishda yo‘l qo‘yiladigan xatolikni umumiy holda baholashda qaysi teoremdan foydalaniladi?

2-§. Oraliqni teng ikkiga bo'lish (biseksiya) usuli

Tayanch so'z va atamalar



Kesmaning o'rtasi, ildizning mavjudlik sharti, ildizga yaqinlashish formulasi, usulning geometrik ma'nosi, ishchi algoritm, dastur matni, usulning xatoligi.

Bu usul iteratsion usullar ichida eng soddasidir. Uni ishlatish uchun maxsus shartlarning bajarilishi talab qilinmaydi. Faqat $f(x)=0$ chiziqsiz tenglamaning izlanayotgan ildizi ajratilgan bo'lishi kerak, ya'ni xqs ildiz $[a, b]$ kesmada yotgan bo'lsin. Kesmaning o'rtasi $c_0 = \frac{a+b}{2}$ da $f(c_0)$ ni hisoblaymiz. Berilgan $[a, b]$ kesmani ikkita teng $[a, c_0]$, $[c_0, b]$ kesmalarga bo'lib, shu kesmalarning chetlarida $f(x)$ funksiyaning ishoralarini tekshiramiz. Qaysi kesmaning chetki nuqtalarida $f(x)$ har xil ishorali qiymatlarni qabul qilsa, $x=c$ ildiz o'sha kesmada bo'ladi. U yoki bu kesmada shunday bo'lishi aniq, chunki ildiz $[a, b]$ kesmada yotadi. Ildiz yotmagan $[a, c_0]$, yoki $[c_0, b]$ kesmani tashlab yuborib, qolgan kesmani yana ikkiga bo'lamiz. Masalan $f(a) \cdot f(c_0) < 0$ bo'lsa, $c_1 = \frac{a+c_0}{2}$ deb olib, $f(c_1)$ ni hisoblaymiz. Yana $[a, c_1]$, yoki $[c_1, c_0]$ kesmalarda $f(x)$ ning ishoralari tekshiriladi va hokazo. Shunday qilib, har bir iteratsiyadan so'ng yechim yotgan kesma uzunligi ikki baravar qisqarib boradi.

Bu jarayonni to kesma uzunligi ε dan kichik bo'lguncha davom ettiriladi. Bunda ε - yechim aniqligini ifodalovchi musbat, o'ta kichik son. Oxirgi kesmaning ixtiyoriy nuqtasi taqribiy yechim sifatida qabul qilinadi. Demak, biz usulni qo'llash natijasida bir-birini ichida joylashgan cheksiz $(a_1, b_1) \dots (a_n, b_n)$ kesmalar ketma-ketligini hosil qilamiz va oxirgi toraygan kesma

$$b_n - a_n = \frac{1}{2^n}(b-a)$$

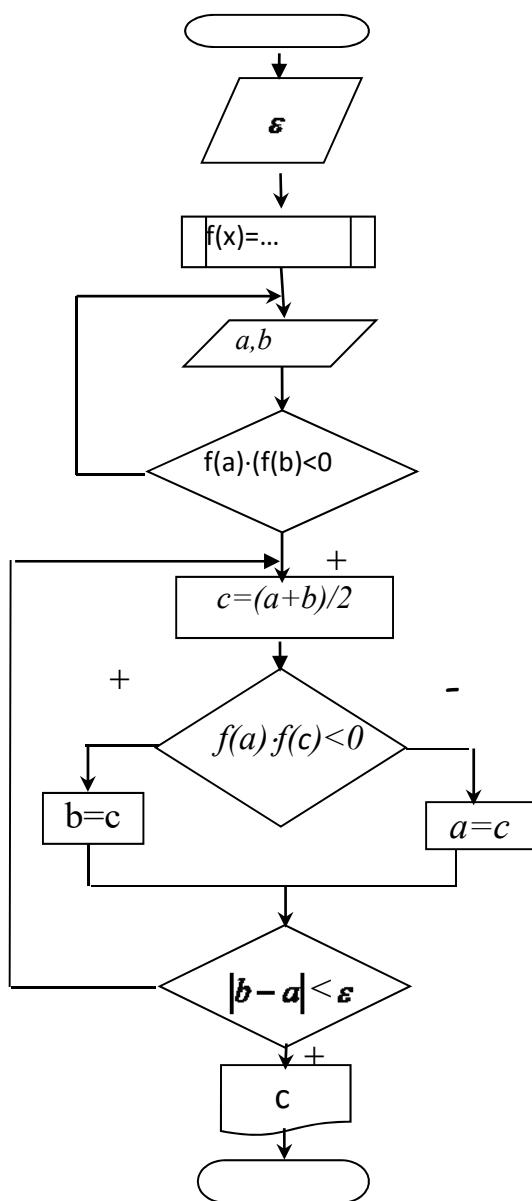
ga teng bo'ladi. Bunda yo'l qo'yilgan xatolik $\Delta x_n = \frac{b-a}{2^n}$ talab qilingan ε aniqlik

bilan solishtirib chiqiladi. Agar $\Delta x_n < \varepsilon$ shart bajarilsa, masala yechilgan bo'ladi.

Yuqorida qayd qilingan ijobiy hislatlari bilan birga biseksiya, ya'ni kesmani ikkiga bo'lish usulining asosiy kamchiligi, uning o'ta sekin yaqinlashishini ham aytib o'tish lozim. Shuning uchun, bu usul ketma-ket yaqinlashishlarning yuqori tezligi talab qilinmagan hollarda ishlatiladi.

Endi usul algoritmini blok-sxemalarda ifoda etib, u asosida dastur ta'minotini yarataylik va algoritmi hamda dasturning ishga yaroqlilik holatini «test» misol orqali baholaylik.

Algoritm blok-sxemasi



Dastur matni

```

# Teng bo'lish
import math
k, a, b, c, epc, i = 0, 0, 0, 0, 0, 0
def f(x):
    f = math.sqrt(x+2)+0.7*x
    return f
k = 0
a, b = map(float, input("a, b = ").split())
while f(a)*f(b) > 0:
    a, b = map(float, input("a, b = ").split())
eps = float(input("eps = "))
while abs(b-a) > eps:
    if k > 14:
        break
    c = (a+b)/2
    k += 1
    print(f'k: {k}, {c:6.5f}, {f(c):8.5f}')
    if f(a) * f(c) < 0:
        b = c
    else:
        a = c
  
```

Misol. Oraliqni teng ikkiga bo‘lish usuli yordamida

$$\sqrt{x+2} + 0.7x = 0$$

tenglamani biror ildizini 0,001 aniqlikda toping.

Yechish. Dastlab, tenglamaning ildizi yotgan oraliqni yuqorida berilgan usullar yordamida ajratamiz.

Grafik usulni qo‘llab tenglamaning yagona ildizi $(-2,-1)$ oraliqda ekanligini ko‘rishimiz mumkin:

$$f(-2) = -1.4 < 0, \quad f(-1) = 0.3 > 0$$

$a = -2$, $b = -1$, $\varepsilon = 0.001$ boshlang‘ich qiymatlarni kiritamiz va oraliqni teng ikkiga bo‘lishlar yordamida yechimga yaqinlashuvchi algoritmni dastur bo‘yicha qo‘llab, quyidagi natijalarga erishamiz.

n	C_n	$f(C_n)$
1	-1.50000	0.34289
2	-1.25000	0.00897
3	-1.12500	0.14791
4	-1.18750	0.07014
5	-1.21875	0.03076
6	-1.23438	0.01094
7	-1.24219	0.00099
8	-1.24609	0.00399
9	-1.24414	0.00150
10	-1.24316	0.00025
11	-1.24268	0.00037
12	-1.24292	0.00006
13	-1.24304	0.00010
14	-1.24298	0.00002

Olingan natijalardan ko‘rinib turibdiki, $\varepsilon = 0,001$ aniqlikdagi

$x = -1,24414$ taqribiy yechimga $n=9$ da erishdik. n ning ortib borishi bilan yechim aniqligi ham ortib borayotganligini jadvaldan kuzatishimiz mumkin.



Nazorat savollari

1. Oraliqni teng ikkiga bo‘lish usulining geometrik ma‘nosi qanday ifodalanadi?
2. Usulga mos asosiy ishchi formula qanday xosil qilinadi?
3. Oraliqni teng ikkiga bo‘lish usulining afzalligi nimada?
4. Bu usulda dastlabki yaqinlashishni aniqlash kerakmi?
5. Oraliqni teng ikkiga bo‘lish usulining xatoligi qanday baholanadi?

3-§. Urinmalar (Nyuton) usuli

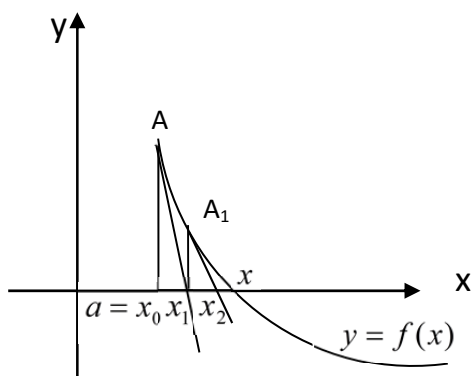
Tayanch soʻz va atamalar



Dastlabki yaqinlashish, dastlabki yaqinlashishni aniqlovchi shart, usulning geometrik maʼnosi, asosiy ishchi formula, Nʼyuton usulining xatoligi, usulning ishchi algoritmi, dastur matni.

Oraliqni teng ikkiga boʻlish usulidagi amallar sonining koʻpligi urinmalar usulida deyarli uchramaydi. Agar dastlabki yaqinlashish toʻgʻri tanlansa, bu usulda taqribiy yechim juda tez topiladi. Usulning mohiyati quyidagicha:

$f(x) = 0$ tenglama $[a, b]$ oraliqda bitta taqribiy ildizga ega deb faraz qilaylik.



6-pacm

Dastlabki yaqinlashish sifatida a yoki b nuqtalardan birini olishimiz mumkin va shu tanlangan nuqtadan urinma oʻtkazamiz. Aytaylik urinma $A(a, f(a))$ nuqtadan oʻtsin (6-rasm). Urinmaning OX oʻqi bilan kesishgan nuqtasi x_1 ga mos nuqtani A_1 deb olib, endi $A_1(x_1, f(x_1))$ nuqtadan urinma oʻtkazamiz, va h. Urinmaning OX oʻqi bilan kesishgan nuqtalari ildizga yetarli

aniqlikkacha yaqinlashguncha jarayon davom etadi.

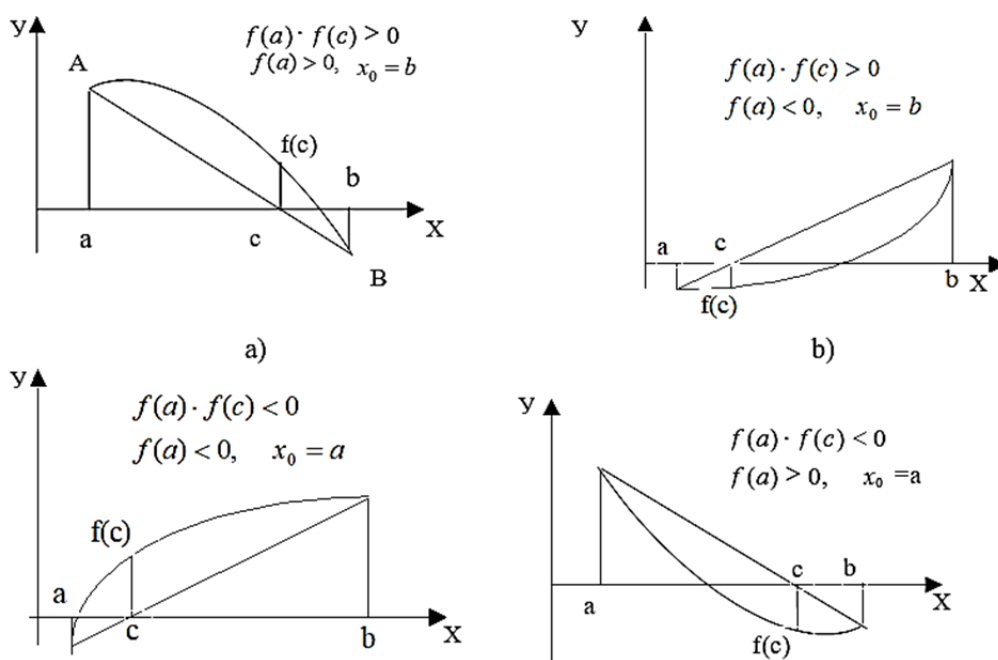
Bu usulda x_0 ni toʻgʻri tanlash juda muhimdir. Shuning uchun, dastlabki yaqinlashish x_0 ni tanlash masalasiga alohida eʼtibor beramiz. Buning uchun $(a, f(a))$ va $(b, f(b))$ nuqtalardan oʻtuvchi vatarni OX oʻqi bilan kesishish nuqtasi c ning qiymatini shu ikki nuqta orqali oʻtuvchi toʻgʻri chiziq tenglamasidan aniqlaymiz.

$$\frac{y - f(a)}{f(b) - f(a)} = \frac{x - a}{b - a}$$

Vatarning OX o'qi bilan kesishish nuqtasi c_0 da $x = c_0$, $y = 0$ bo'ladi, u holda yuqoridagi ifodadan quyidagi ko'rinishga ega bo'lgan formulani xosil qilamiz:

$$c = a - \frac{b-a}{f(b)-f(a)} f(a)$$

c ma'lum bo'lgach, $f(c)$ ning qiymatini hisoblash mumkin. $C(c, f(c))$ nuqtani yechimga nisbatan joylashishi mumkin bo'lgan barcha hollarni ko'rib chiqaylik (7-rasm).



7-rasm

$$y - f(x_0) = f'(x_0)(x - x_0)$$

dan urinma OX o'qi bilan kesishgani uchun $y(x_1) = 0$ deb olib

$x_1 = x_0 - f(x_0) / f'(x_0)$ tenglikni va bu formulani umumlashtirib usulga mos ishchi formulani hosil qilamiz:

Ushbu ko'rinishlarga mos ravishda usul uchun dastlabki yaqinlashish tanlanadi:

- 1) $f(a) > 0$ va $f(a)f(c) > 0$ bo'lsa $x_0 = b$;
- 2) $f(a) < 0$ va $f(a)f(c) > 0$ bo'lsa $x_0 = b$;
- 3) $f(a) < 0$ va $f(a)f(c) < 0$ bo'lsa $x_0 = a$;

4) $f(a) > 0$ va $f(a)f(c) < 0$ bo'lsa $x_0 = a$;

shartlarni umumlashtirib olib, $f(a)f(c)$ ko'paytmaning ishorasi musbat-manfiyligiga qarab, a yoki b qiymatlardan birini urinmalar usulida dastlabki yaqinlashish sifatida olish mumkin degan xulosalarga kelamiz.

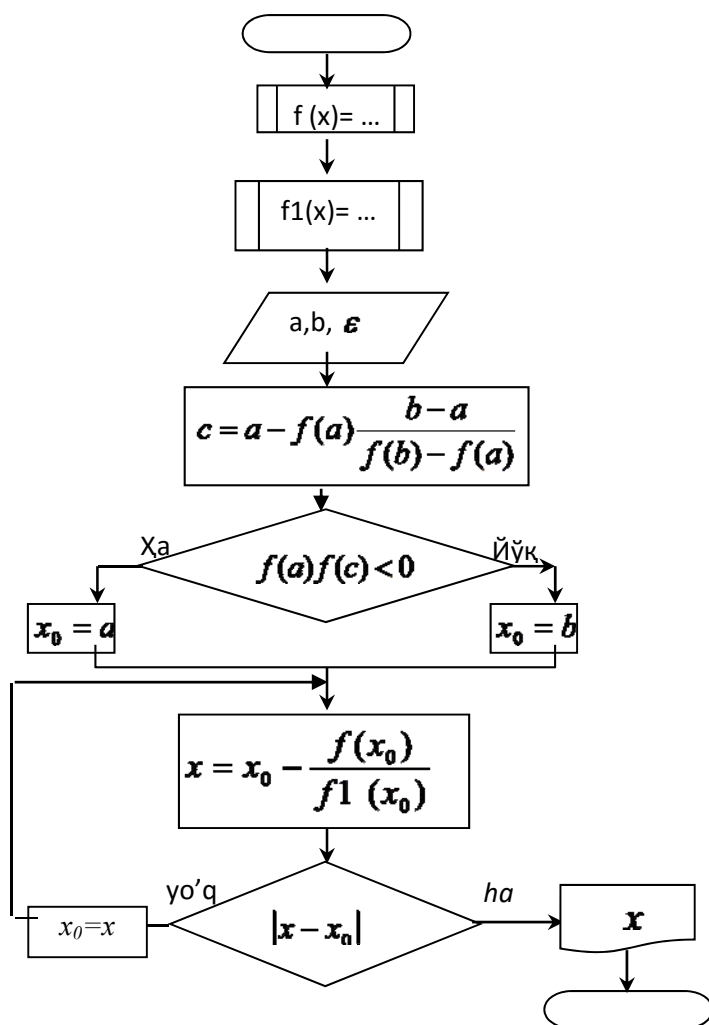
Endi $(x_0, f(x_0))$ nuqtaga o'tkazilgan urinma tenglamasi

$$x_n = x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})}$$

Hosil bo'lgan ishchi formula urinmalar usulining asosiy formu-lasi bo'lib, hisoblashlar $|x_{n+1} - x_n| < \epsilon$ sharti bajarilguncha davom ettiriladi.

Endi usul algoritmining blok-sxemasi va dasturini keltiramiz va ular asosida chiziqsiz tenglamaning taqribiy ildizlarini aniqlaymiz.

Algoritm blok-sxemasi



Dastur matni

```

# Urinma
a,b,x,x0,eps,c,k = 0,0,0,0,0,0,0
def f(x):
    f = 2*x*x*x-0.4
    return f
def f1(x):
    f = 6*x*x
    return f
a, b = map(float, input("a, b = ").split())
eps = float(input("Aniqlikni kiriting: "))
c = a-f(a)*(b-a)/(f(a)-f(b))
if f(a)*f(c)<0:
    x0 = a
else:
    x0 = b
x = x0-f(x0)/f1(x0)
while abs(x-x0) > eps:
    print(f' {k}, {x:5.5f}, {f(x):5.6f} ')
    x0 = x
    k += 1
    if k > 15:
        break
  
```

Demak, yechim aniq ildizga monoton yaqinlashuvchi ketma-ketlik limitidan iborat bo‘ladi: $\lim_{n \rightarrow \infty} x_n = \xi$

n -yaqinlashishda x_n taqribiy ildiz uchun Nyuton usulining xatoligi quyidagicha baholanadi:

$$|\xi - x_n| \leq \frac{|f(x_n)|}{m}$$

Bu yerda m soni (a, b) kesmadagi $|f'(x)|$ ning eng kichik qiymati.

Yaqinlashish jarayoni bu usulda boshqa usullarga qaraganda juda tez bo‘lgani uchun undan amalda ko‘p qo‘llaniladi. Endi Nyuton usulini hamma vaqt ham ishlatish mumkinmi degan savolga javob beramiz:

4-teorema. Agar (a, b) kesmada $f'(x)$, $f''(x)$ lar noldan farqli bo‘lib, ishorasini o‘zgartirmasa, $f(c_0) \cdot f''(c_0) > 0$ shartni qanoatlantiruvchi $c_0 \in [a, b]$ nuqta boshlang‘ich yaqinlashish sifatida qabul qilinishi mumkin va shunda $c_0, c_1, \dots$ ketma-ketlik yagona $x=c$ yechimga intiladi.

Misol. Urinmalar usuli yordamida $2x^3 - 0.4 = 0$ tenglamaning biror haqiqiy ildizini 0,0001 aniqlikda toping.

Yechish. Tenglamaning ildizi $(0, 1)$ oraliqda va u yagona ekanligini grafik usulni qo‘llab aniqlaymiz. Haqiqatan ham

$$f(0) = -0.4 < 0, \quad f(1) = 1.6 > 0$$

Ildizning mavjudlik sharti $(0, 1)$ oraliqda bajarilganligi uchun yechimni shu oraliqdan aniqlaymiz. Bundan tashqari, $f'(x) = 6x^2$, $f''(x) = 12x$ hosilalar mazkur oraliqda ishoralarini o‘zgartirishmaydi. Demak, $c_0, c_1, c_2, c_3, \dots$ ketma-ket yaqinlashishlar tenglamaning yagona ildiziga intiladi.

$a = 0$, $b = 1$, $\varepsilon = 0,0001$ qiymatlarni kiritamiz va urinmalar usuli yordamida yechimga yaqinlashuvchi algoritmi qo‘llab, dastur bo‘yicha quyidagi natijalarga erishamiz. Bu yerda n -yaqinlashishlar soni.

n	C_n	$f(C_n)$
1	0.73333	0.388741
2	0.61286	0.060368
3	0.58607	0.002600
4	0.58481	0.000006

Natijalardan ko‘rinib turibdiki, ishlab chiqilgan algoritm asosida taqribiy yechim juda tez, aniqrog‘i to‘rtta yaqinlashish bilan topildi. Zero, avvalgi oraliqni teng ikkiga bo‘lish usulida 0,001 aniqlik uchun ham taqribiy yechimga yaqinlashishlar soni 10 tadan ortiq edi. Bunday farq urinmalar usulining yechimni tez topish imkoniyati jihatidan ustun ekanligini ko‘rsatadi.



Nazorat savollari

1. Urinmalar usulining asosiy mohiyati nimada?
2. Urinmalar usulining asosiy afzalligi nimada?
3. Urinmalar usulida dastlabki yaqinlashish qanday aniqlanadi?
4. Usulning ishchi formulasi qanday xosil qilinadi?
5. Urinmalar usulida xatolik qanday baholanadi?

4-§. Vatarlar usuli



Tayanch so‘z va atamalar

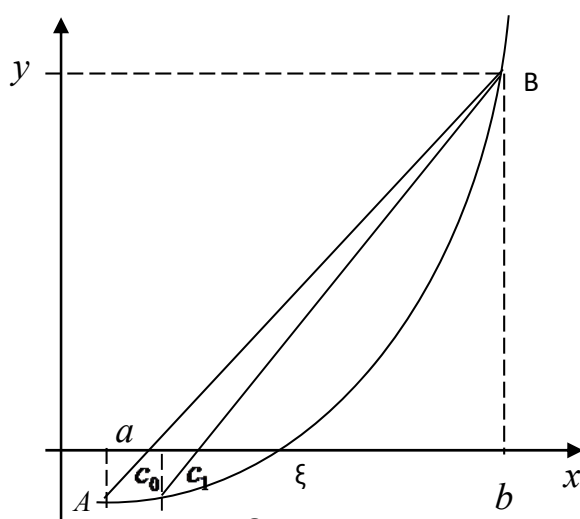
Dastlabki yaqinlashish, dastlabki yaqinlashishni aniqlovchi shart, usulning geometrik ma‘nosi, asosiy ishchi formula, vatarlar usulining xatoligi, usulning ishchi algoritmi, dastur matni.

Bu usul ham $f(x) = 0$ tenglamaning ildizini berilgan $[a, b]$ kesmada tez va aniqroq topish imkonini beradi. Berilgan tenglamalarning $f(x)$ funksiyasi $[a, b]$ kesmada uzluksiz va uning chegaralarida har xil ishorali qiymatlariga ega bo‘lib, $f(a) \cdot f(b) < 0$ sharti bajarilsin. Urinmalar usulidan farqli ravishda bu usulda

haqiqiy yechimga vatarlar yordamida yaqinlashib boramiz. Avval $A(a, f(a))$ va $B(b, f(b))$ nuqtalardan vatar o'tkazaylik (8-rasm). U OX o'qini c_0 nuqtada kesib o'tadi. Ma'lumki, vatarni OX o'qi bilan kesishishidan hosil bo'lgan nuqtaning abssissasi:

$$c_0 = a - \frac{b - a}{f(b) - f(a)} \cdot f(a)$$

dan iborat bo'ladi. Bu nuqtani dastlabki yaqinlashish sifatida olishimiz mumkin. Lekin, keyingi vatarlarni qaerdan o'tkazamiz degan savol tug'iladi. Buning uchun, oraliqni qo'zg'almas nuqtasini



8-rasm

$f(a) \cdot f(c_0) < 0$ sharti yordamida aniqlab olishimiz kerak.

Chizmadan ko'rinib turibdiki, agar $f(a) \cdot f(c_0) < 0$ sharti bajarilsa, $b = c$ bo'lib, a nuqta qo'zg'almas bo'ladi, aks holda $a = c$ bo'lib, b nuqta qo'zg'almas bo'ladi.

Ildizga yaqinlashuvchi $c_0, c_1, \dots, c_n, \dots$ ketma-ketlik $f(x)$ funksiyaning vatarlarini OX o'qi bilan kesishish nuqtalarini tashkil qiladi.

$$c = a - \frac{b - a}{f(b) - f(a)} f(a)$$

formuladan esa ishchi formula sifatida foydalanamiz. Urinmalar usulidagi singari bu usulda ham $f(a) \cdot f''(x) > 0$ sharti bajarilsa ishchi formula yordamida topilgan qiymatlar yechimga yaqinlashuvchan bo'ladi. Jarayon kerakli aniqlikdagi yechim olinmaguncha davom etaveradi.

Endi vatarlar usulining xatosini baholaymiz. $f'(x)$ xosila (a, b) kesmada uzluksiz va o'zining ishorasini saqlaydi, deb faraz qilamiz. ξ va x_n $f(x) = 0$ tenglamaning aniq va taqribiy yechimlari bo'lsin. U holda

$$|\xi - x_n| \leq \frac{M_1 - m_1}{m_1} |x_n - x_{n-1}|$$

tengsizlik o'rinli bo'ladi. Bu yerda m_1 va M_1 lar (a, b) kesmada $f'(x)$ ning moduli bo'yicha eng katta va eng kichik qiymatlari. Ko'pincha amaliyotda, agar talab qilingan aniqlik $\varepsilon > 0$ musbat sonidan iborat bo'lsa, xatoni aniqlash uchun oxirgi x_n yaqinlashish x_{n-1} yaqinlashishdan ε ga nisbatan kamroq farq qilsa, ya'ni $|x_n - x_{n-1}| < \varepsilon$ bo'lsa, u vaqtda limit absolyut xato sifatida olinadi: $|\xi - x_n| < \varepsilon$.

Quyida vatarlar usuliga mos algoritm-blok-sxemasi va dasturi keltirilmoqda.

Ishlab chiqilgan algoritm asosida quyidagi chiziqsiz tenglamani yechishni tashkil etaylik.

Misol. Vatarlar usuli yordamida $x^3 + 2x + 0.5 = 0$ tenglama ildizini 0.0001 aniqlikda hisoblang.

Yechish. Ushbu chiziqsiz tenglama uchun ildiz yotgan oraliq $(-2, -1)$ ekanligini grafik usul bilan aniqlaymiz va oraliqni to'g'riligini analitik usulda tekshiramiz:

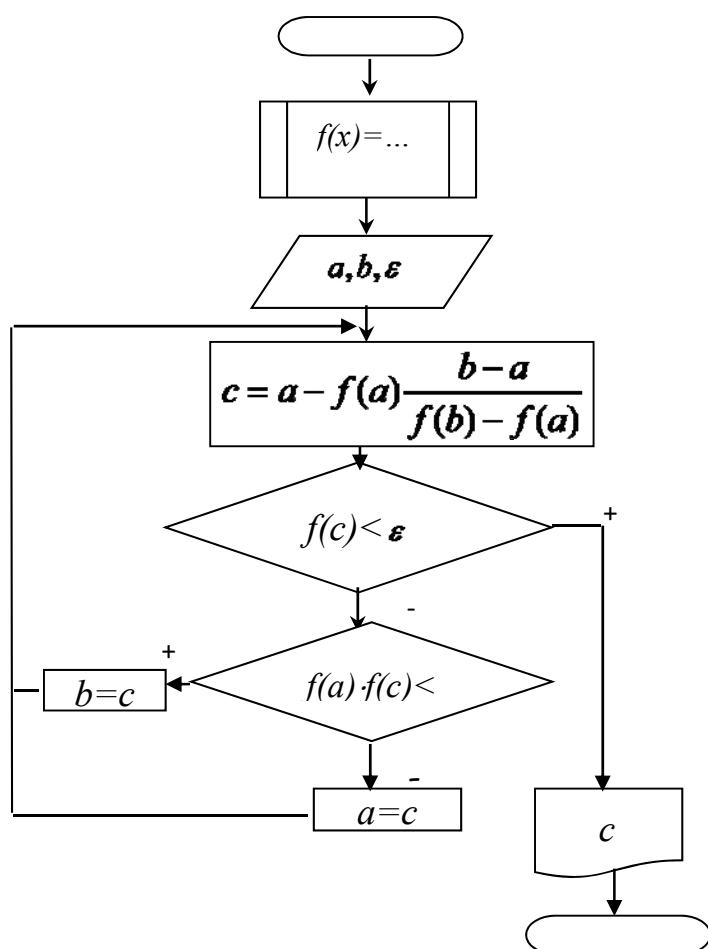
$$f(-1) = -2.5 < 0, \quad f(0) = 0.5 > 0$$

Xuddi urinmalar usulidagi singari birinchi va ikkinchi tartibli xosilalar

$$f'(x) = 3x^2 + 2, \quad f''(x) = 6x$$

$(-2, -1)$ oraliqda o'z ishorasini saqlagani uchun usul yordamida hosil qilingan yaqinlashishlar tenglamaning ildiziga intiladi, deya olamiz.

Algoritm blok-sxemasi



Dastur matni

```

# Vatarlar
a,b,x,x0,eps,c,k = 0,0,0,0,0,0,0
def f(x):
    f = x*x*x+2*x+0.5
    return f
a, b = map(float, input("a,b = ").split())
eps = float(input("Aniqlikni kiriting: "))
while abs(f(c))>eps:
    k += 1
    c = a-f(a)*((b-a)/(f(b)-f(a)))
    print(f"{k}, {c:6.5f}, {f(c):8.6f}")
    if f(a)*f(c)<0:
        b = c
    else:
        a = c
    if k>15:
        break
  
```

Vatarlar yordamida yechimga yaqinlashuvchi algoritmnii qo‘llab, dastur bo‘yicha quyidagi ijobiy natijalarga erishamiz. Bu yerda n-yaqinlashishlar soni.

N	C_n	$f(C_n)$
1	-0.16667	0.162037
2	-0.21739	0.054944
3	-0.23422	0.018708
4	-0.23991	0.006373
5	-0.24184	0.002171
6	-0.24250	0.000740
7	-0.24272	0.000252
8	-0.24280	0.000086



Nazorat savollari

1. Vatarlar usulining geometrik ma`nosi qanday ifodalanadi?
2. Vatarlar usulining ishchi formulasi qaysi formula asosida hosil qilinadi?
3. Vatarlar usulida dastlabki yaqinlashishni qanday qilib aniqlanadi?
4. Usulning xatoligini baholash mumkinmi?
5. Vatarlar usulining kamchiligi mavjudmi?

5-§. Iteratsiya usuli

Oddiy iteratsiya usuli



Tayanch soʻz va atamalar

Nolinchi yaqinlashish, usulning geometrik ma`nosi, yaqinlashishni aniqlovchi shart, yaqinlashuvchi jarayon, uzoqlashuvchi jarayon, iteratsiya usulining xatoligi, usulning ishchi algoritmi, dastur ta`minoti.

Algebraik va transsendent tenglamalarni yechishning eng muhim usullaridan biri iteratsiya usuli hisoblanadi. Iteratsiya usulini qoʻllash uchun (2.1) tenglamani unga teng kuchli boʻlgan quyidagi

$$x = \phi(x) \quad (2.2)$$

kanonik koʻrinishga keltirilgan va ildizlari ajratilgan boʻlishi kerak. (2.2) tenglamaning ildizi yotgan atrofning biror x_0 nuqtasini izlanayotgan ildizning nolinchi yaqinlashishi deb olamiz. Navbatdagi yaqinlashishlarni topish uchun (2.2) formuladan foydalanamiz, yaʼni

$$x_n = \phi(x_{n-1}) \quad (2.3)$$

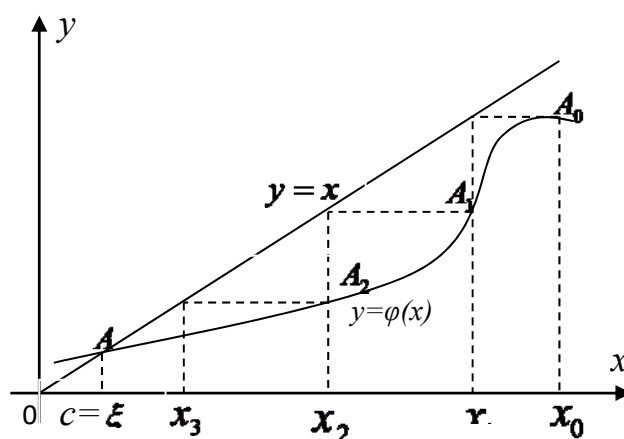
Hosil qilingan sonlar ketma-ketligining limiti

$$\lim_{n \rightarrow \infty} x_n = \xi \quad (2.4)$$

mavjud va $\phi(x)$ funksiya uzluksiz boʻlsa, ξ berilgan tenglamaning ildizi boʻladi. Demak, bu ildizni (2.3) formula yordamida istalgan aniqlik bilan hisoblash mumkin. (2.4) limit mavjud boʻlgan holda iteratsiya jarayoni yaqinlashuvchi

deyiladi. Lekin, mazkur limit har doim ham mavjud bo‘lavermaydi, bunday holda oddiy iteratsiya usulidan foydalanish maqsadga muvofiq bo‘lmay-di.

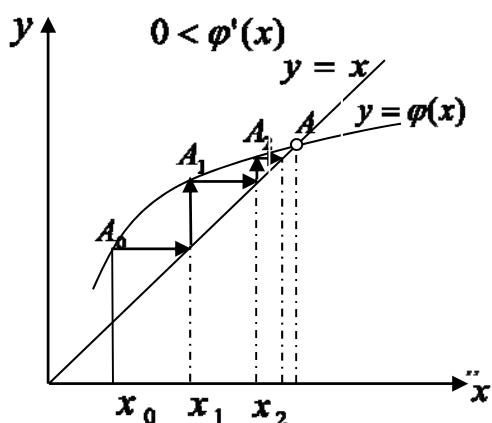
Iteratsiya usuli sodda geometrik ma’noga ega. Buni tushunish uchun $y = x$ va $y = \phi(x)$ funksiyalarning grafiklarini chizamiz.



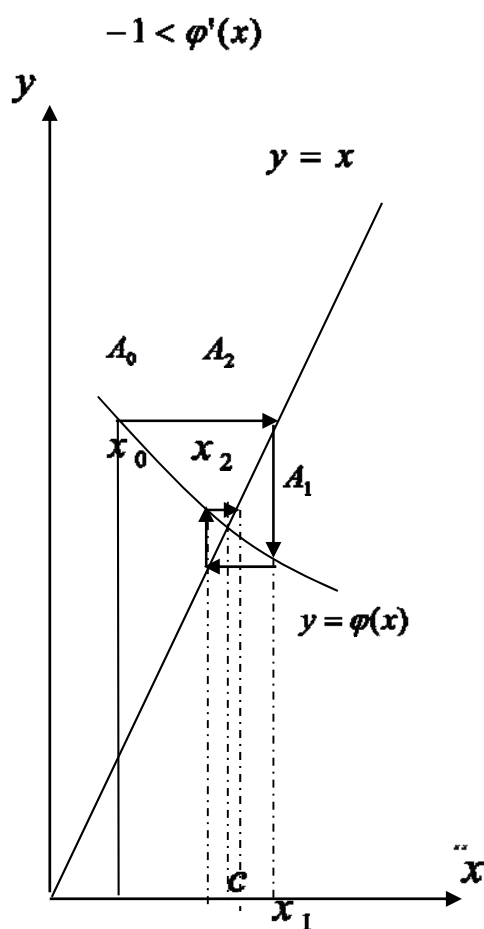
9-rasm

Bu grafiklarning OX o‘qi bilan kesishgan nuqtasining absissasi tenglamaning ildizidan iborat bo‘ladi. Iteratsiya usulining umumiy algoritmiga binoan dastlabki yaqinlashishni tanlab olamiz. Birinchi yaqinlashish bo‘ladi (9-rasm). Bu geometrik nuqtai-nazardan x_0 nuqtaga mos keluvchi $A_0(x_0, \phi(x_0))$ nuqtadan OX o‘qiga parallel to‘g‘ri chiziq o‘tkazib, uning $y = x$ to‘g‘ri chiziq bilan kesishish nuqtasining absissasini topish demakdir. Bu nuqtada $\phi(x_1)$ ni hisoblaymiz.

Natijada $A_1(x_1, \phi(x_1))$ nuqta topiladi. Bu nuqtadan yana OX o‘qiga parallel to‘g‘ri chiziq bilan kesishgan nuqtasining absissasi, ya’ni $x_2 = \phi(x_1)$ ni topamiz va h.k. 10-rasmdan ko‘rinib turibdiki, $0 < \phi'(x) < 1$ sharti bajarilganda iteratsiya jarayoni yaqinlashar ekan, ya’ni $A_0, A_1, \dots$ nuqtalar $A(c, \phi(c))$ nuqtaga yaqinlashib boradi va o‘z navbatida $x_0, x_1, \dots$ ketma-ketlik $x=c$ limitga intiladi.



10-rasm



11-rasm

Endi $-1 < \phi'(x) < 0$ bo'lgan holni qaraymiz (11-rasm). Ketma-ket yaqinlashishlar rasmda strelkalar yordamida yaqqol ko'rsatilgan. Bunda, faqat, oldingi holdan farqli ravishda $x_0, x_1, \dots$ yaqinlashishlar $x = c$ yechimning har xil tarafida yotadi. Bu holda ham yaqinlashuvchi iteratsiya jarayoniga ega bo'lamiz.

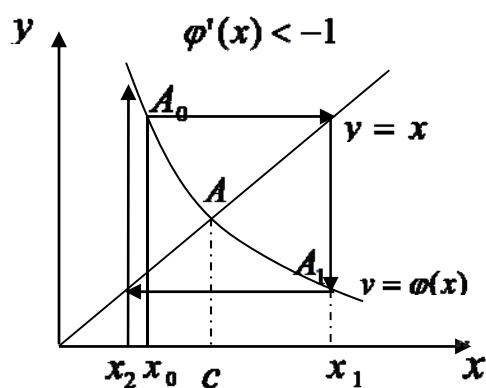
Qolgan $\phi'(x) < -1$, $\phi'(x) > 1$ hollarda (12-13-rasmlar) iteratsiya jarayoni uzoqlashuvchi bo'ladi, $\phi'(x) < -1$ bo'lganda yaqinlashishlar $x = c$ yechimning ikkala tarafida uzoqlashib borsa, $\phi'(x) > 1$ bo'lganda esa ular yechimning bir tarafida uzoqlashadi.

Bu mulohazalarni yakunlab quyidagi umumiy xulosaga kelimiz: iteratsiya usuli qarayotgan sohada $|\phi'(x)| < 1$ bo'lganda yaqinlashadi va $|\phi'(x)| \geq 1$ bo'lganda uzoqlashadi. Iteratsiya usulining hatosini baholash uchun quyidagi formuladan foydalaniladi.

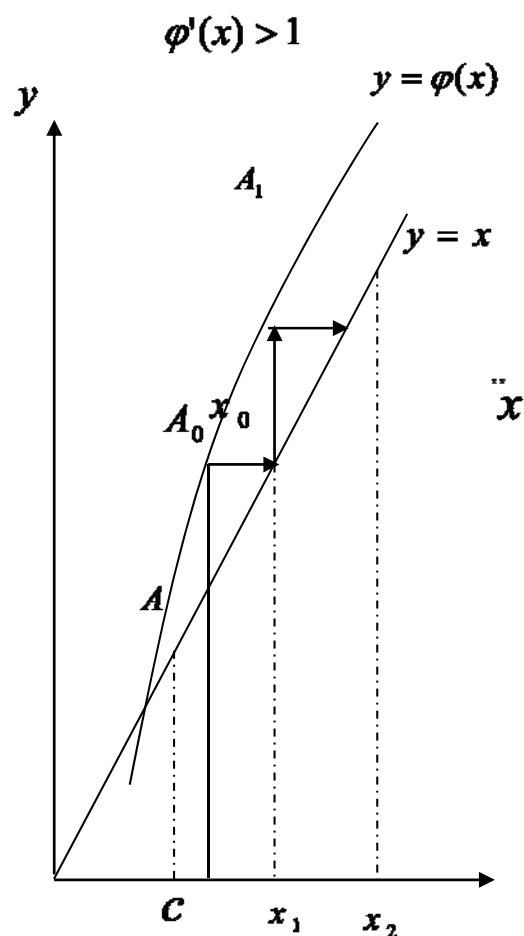
$$|\xi - x_n| < \frac{q^n}{1-q} (x_1 - x_0)$$

Agar q qanchalik kichik bo'lsa, iteratsiya jarayoni shunchalik tez yaqinlashadi. Iteratsiya usulining boshqa usullarga nisbatan ustunligi shundaki, operatsiyalarning bajarilishi har bir qadamda bir xil bo'lib, bu dastur tuzish ishini sezilarli darajada yengillashtiradi.

Quyida usulning ishchi algoritmiga mos blok-sxema va dasturi keltiriladi. Hamda algoritm va dastur ishi aniq misol uchun tahlildan o'tkaziladi.



12-rasm



13-rasm

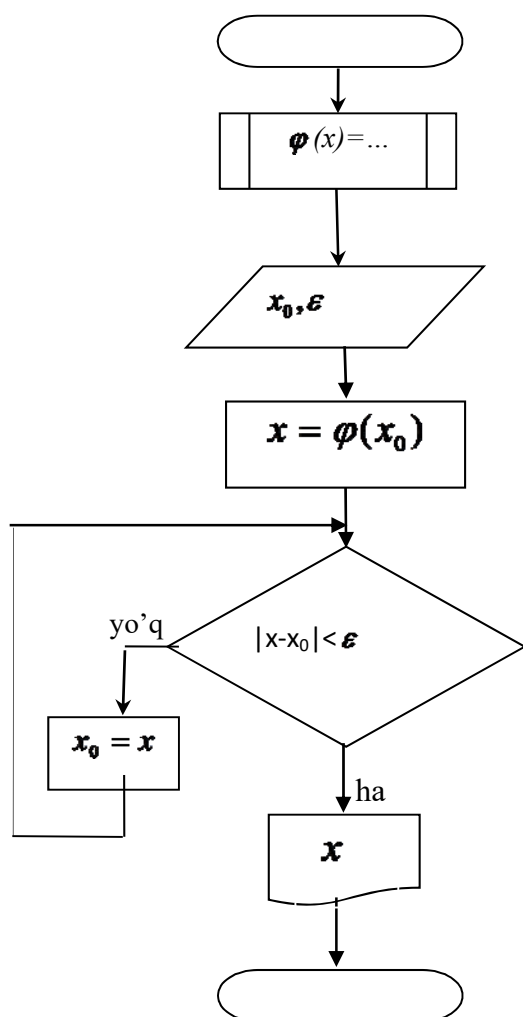
Misol. $0,1x^3 - 0,4x - 80 = 0$ tenglamaning eng katta musbat ildizini 0,0001 aniqlikkacha toping.

Yechish. Birinchi taqribiy yaqinlashish sifatida $x_0 = 10$ va $x_0 = 8$ ko‘rinishlarini olamiz va berilgan tenglamani quyidagi ko‘ri-nishda ifodalaymiz:

$$x = \sqrt[3]{4x+800}, \text{ ya'ni } \phi(x) = \sqrt[3]{4x+800} \text{ deb olsak, } |\phi'(x)| \leq \frac{4}{3\sqrt[3]{(4x+800)^2}} < 1.$$

Demak, iteratsiya jarayoni yaqinlashuvchidir.

Algoritm blok-sxemasi



Dastur matni

```

# Iteratsiya
import math
x, x0, eps, i = 0, 0, 0, 0
def ff(x):
    ff = 0.1*x*x*x-0.4-x-80
    return ff
def f(x):
    f = math.exp(math.log(4*x+800)/3)
    return f
x0 = float(input("X0 ni kiriting: "))
eps = float(input("Aniqlikni kiriting: "))
i = 1
x = f(x0)
while abs(x-x0)>eps:
    print(f"{i}, {x:6.5f}, {ff(x):8.6f}")
    i += 1
    x0 = x
    x = f(x0)
  
```

Ishlab chiqilgan algoritm dasturi asosida keyingi yaqinlashishlarni topamiz.

N	x_n	$f(x_n)$
0	8	32
1	9.405339	0.562136
2	9.426473	0.008454
3	9.426791	0.000127
N	x_n	$f(x_n)$
0	10	16
1	9.435388	0.225845
2	9.426924	0.003385
3	9.426797	0.000051

Natijalarni tahlil qilib, shunday xulosalarga kelish mumkin: agar iteratsion jarayonni tashkil etuvchi $\phi(x)$ funksiya to‘g‘ri tanlansa, yechim juda oson topiladi, jarayonning yaqinlashishi faqat shu funksiyaga bog‘liq, chunki ixtiyoriy dastlabki yaqinlashishda ham iteratsion qiymatlar o‘zini darhol o‘nglab oladi va yechimga intiladi.

Agar biz berilgan tenglamani $x = 0,1x^3 - 80$ ko‘rinishida yozib olsak, $\phi'(x) = 0,3x^2$ bo‘lib, $9 < x < 10$ qiymatlarida $\phi'(x) > 27$ bo‘ladi va iteratsiya jarayonining yaqinlashish sharti bajarilmaydi.

Demak, bundan ko‘rinadiki, berilgan tenglamani ixtiyoriy ko‘rinishda yozib olish har doim ham maqsadga yetkazavermaydi. Bu kamchilikni yo‘qotadigan usullardan biri Vegsteyn usulidir. U ham aslida iteratsion jarayon bo‘lib, o‘ziga xos yaqinlashish formulasiga egaligi bilan oddiy iteratsiya usulidan farq qiladi.

Vegsteyn usuli

Yuqorida keltirilgan mulohazalardan aytish mumkinki, iteratsiya usulining yaqinlashishi yoki uzoqlashishi ξ ildizning kichik atrofida $\phi'(x)$ hosilaning qiymatiga bog‘liq bo‘ladi. Bu esa qidirilayotgan ildizga har doim ham yaqinlashish imkoni mavjud emasligini bildiradi. Lekin, J. X. Vegsteyn 1958 yilda geometrik mulohazalar asosida, iteratsiya usulini shunday o‘zgartirishni taklif etadiki, uni qo‘llaganda $\phi'(x)$ ning qiymati har qanday bo‘lganda ham iteratsiya jarayoni

yechimga yaqinlashadi. Mabodo $|\phi'(x)| < 1$ tengsizlik bajarilsa, u vaqtda oddiy iteratsiya jarayoniga nisbatan Vegsteyn jarayoni tezroq yaqinlashadi. Vegsteyn usulida (2.2) tenglamaning aniq yechimi ξ ning dastlabki yaqinlashishi x_0 orqali,

$$z_0 = x_0, \quad z_1 = x_1 = \phi(x_0) \quad (2.5)$$

deb olib, ikkita $\{x_n\}$ va $\{z_n\}$ ketma-ketlikni quramiz:

$$\begin{aligned} x_{n+1} &= \phi(z_n), \\ z_{n+1} &= x_{n+1} - \frac{(x_{n+1} - x_n)(x_{n+1} - z_n)}{x_{n+1} - x_n - z_n + z_{n-1}} \end{aligned} \quad (2.6)$$

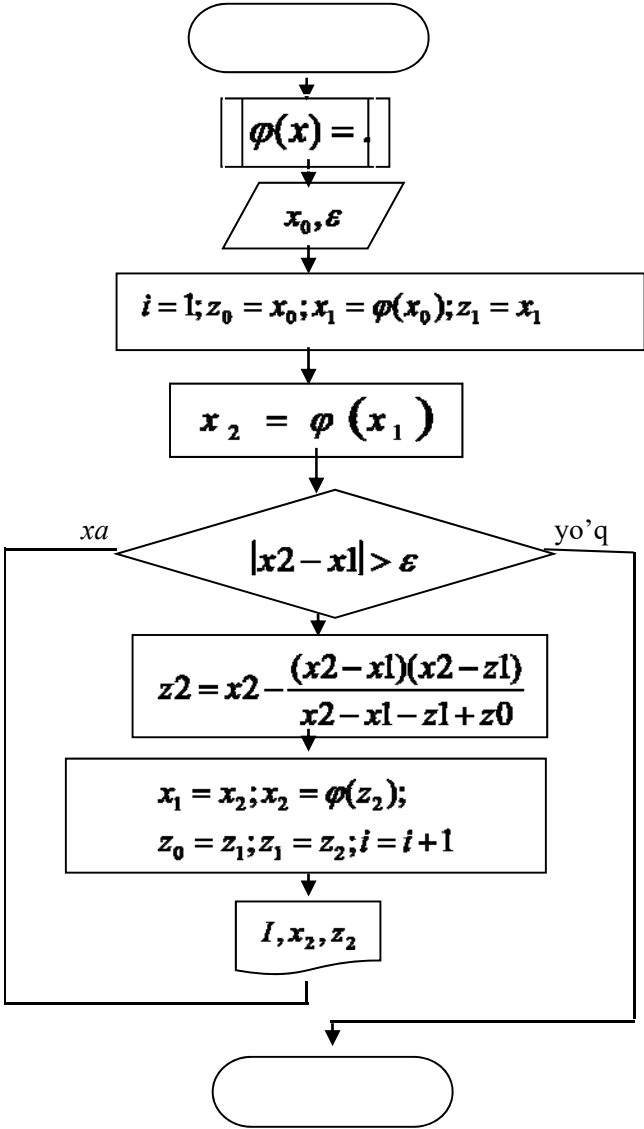
Vegsteyn usulini qo'llash uchun ildizning nolinchisi yaqinlashishi x_0 ga nisbatan bir marta oddiy iteratsiya usulini qo'llash kerak. Birinchi qadamdan so'ng, keyingi yaqinlashishlarni topish uchun (2.6) formulalarni qo'llaymiz. Bu usulda oddiy iteratsiya usuliga nisbatan yechimga qat'iy yaqinlashishni misol yordamida ko'ramiz.

Misol. Vegsteyn usuli yordamida yuqorida berilgan tenglamani

$x = 0,1x^3 - 80$ iteratsion yaqinlashishlar bilan 0,0001 aniqlikda yeching.

Yechish. Ma'lumki, oldingi paragrafda ko'rib o'tganimizdek $x = 0,1x^3 - 80$ hol uchun oddiy iteratsiya usulining yaqinlashish sharti bajarilmaydi. Nolinchisi yaqinlashish sifatida $x_0 = 10$, aniqlik uchun $\varepsilon = 0.0001$ deb olamiz. Vegsteyn usuli bilan topilgan ketma-ket yaqinlashishlarni aniqlaymiz. Natijalar jadvalidan ko'rinib turibdiki, oxirgi ustundagi qiymatlar oddiy iteratsiya usuli uchun olingan bo'lib, xattoki 0,01 aniqlikda ham yechimdan juda tez uzoqlashib ketiladi.

Algoritm blok-sxemasi



Dastur matni

```

# Vegsteyn
x1, x0, x2, z1, z0, eps, i = 0,0,0,0,0,0,0
def f(x):
    f = 0.1*x*x*x-80
    return f
eps = float(input("Aniqlikni kiriting: "))
x0 = float(input("X0 ni kiriting: "))
i=0
z0=x0
print(f"{i}, {x1:6.5f}, {z1:6.5f}")
x2 = f(z1)
while abs(x2-x1) > eps:
    z2=x2-((x2-x1) * (x2-z1))/(x2-x1-1+z0)
    x1=x2
    x2=f(z2)
    z0=z1
    z1=z2
    i+=1
print(f"{i}, {x2:6.5f}, {z2:6.5f}")
  
```

N	$x_{n+1} = \varphi(z_n)$ Vegsteyn	z_n	$x_{n+1} = \varphi(x_n)$ Oddiy iteratsiya
0	10	10	10
1	20	20	20
2	15.714881	9.855072	720
3	13.241184	9.769431	37324720
4	9.721507	9.644925	519983634610000
5	9.643155	9.642117	
6	9.642078	9.642078	
7	9.642078	9.642078	

Vegsteyn usuli o'zini juda tez «o'nglab» olganligi uchun, hisoblash jarayoni biroz uzoq bo'lsa ham, iteratsion jarayonlarda uni qo'llash maqsadga muvofiq degan xulosalarga kelish mumkin.



Chiziqsiz tenglama yechimining aniqligi usulning tanlanishiga bog'liqmi?



Nazorat savollari

1. Iteratsiya usulining geometrik ma'nosi qanday ifodalanadi?
2. Iteratsiya usulida dastlabki yaqinlashish qanday aniqlanadi?
3. Usulning ishchi formulasi qanday xosil qilinadi?
4. Iteratsion jarayonni yechimga yaqinlashishi qaysi formula yordamida tekshiriladi?
5. Iteratsiya usulining xatoligi qanday baholanadi?
6. Vegsteyn usulida dastlabki yaqinlashish qanday aniqlanadi?
7. Vegsteyn usulida ishchi formulalar qanday hosil qilinadi?
8. Vegsteyn usulining o'ziga xos va iteratsiya usuliga o'xshashlik xususiyati nimada?

6-§. Chiziqsiz tenglamalar sistemasini yechishning oddiy iteratsiya usuli



Tayanch so'z va atamalar

Aniq usullar, taqribiy usullar, iteratsiya usuli, kanonik shakl, dastlabki yaqinlashish, iteratsiya usulining yaqinlashish sharti, iteratsion jarayon, usulning ishchi algoritmi, dastur matni.

Ma'lumki, tenglamalar sistemasini yechish usullarini ikki guruhga bo'linadi: aniq va iteratsion. Aniq usullar yordamida sistemani yechgan bilan aniq yechimni har doim ham topa olmasligimiz mumkin. Chunki berilgan sistemadagi ayrim qiymatlar taqriban olingan bo'lishi, bundan tashqari, hisoblash jarayonida sonlarni yaxlitlashga to'g'ri kelishi mumkin. Iteratsion usullarda esa yechim cheksiz ketma-ketliklarning limiti sifatida olinadi. Lekin, bu usullarning o'ziga xos tomonlaridan biri shundan iboratki, ular o'z xatosini o'zi tuzatib boradi.

Agar aniq usullar bilan ishlayotganda biror qadamda xatoga yo‘l qo‘yilsa, bu xato oxirgi natijaga ham o‘z ta‘sirini o‘tkazadi. Yaqinlashuvchi iteratsion jarayonning biror qadamida yo‘l qo‘yilgan xato esa faqat bir necha iteratsiya qadamini ortiqcha bajarishgagina olib keladi, xolos. Ya‘ni, biror qadamda yo‘l qo‘yilgan xato keyingi qadamlarda tuzatib boriladi. Iteratsion usullarning hisoblash sxemalari juda sodda bo‘lib, ularni dasturlash juda qulaydir. Lekin, har bir iteratsion usulning qo‘llanish sohasi chegaralangandir. Chunki, iteratsiya jarayoni berilgan sistema uchun uzoqlashishi yoki, shuningdek, sekin yaqinlashishi mumkinki, amalda yechimni qoniqarli aniqlikda topib bo‘lmaydi. Shuning uchun ham, iteratsion usullarda faqat yaqinlashish masalasigina emas, balki yaqinlashish tezligi masalasi ham katta ahamiyatga egadir. Yaqinlashish tezligi dastlabki yaqinlashish vektorining qulay tanlanishiga ham bog‘liqdir. Aytib o‘tilgan mulohazalar chiziqli tenglamalar sistemasini iteratsion usullar yordamida yechishga tegishli bo‘lib, chiziqsiz tenglamalar sistemasini iteratsion usullar yordamida yechishda bu jarayon birmuncha boshqacharoq kechadi.

[illegible]

chiziqsiz tenglamalar sistemasini yechish masalasi qo'yilgan bo'lsin. Bu sistemani yechish uchun avval berilgan sistemani biror usul bilan quyidagi kanonik shaklga keltirib olinadi:

(2.8)

Masalan, $x^{(k)} = (x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)})$ yaqinlashish topilgan bo'lsa,

$$\begin{cases} x_1^{(k+1)} = \phi_1(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}) \\ x_2^{(k+1)} = \phi_2(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}) \\ \vdots \\ x_n^{(k+1)} = \phi_n(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}) \end{cases} \quad (2.9)$$

Iteratsiya jarayoni

sharti bajarilguncha davom ettiriladi. Bu yerda ε -izlanayotgan yechim aniqligi.

$x^{(0)}$ boshlang'ich yaqinlashishlarda topish imkoniyatini beradi. Bu shartlar yaqinlashish shartlari deyiladi. Muayyan aniqlikdagi yechimni olish uchun kerak bo'lgan iteratsiyalar soni dastlabki yaqinlashishlarga bog'liq bo'ladi. Dastlabki yaqinlashish topilayotgan taqribiy yechimga qancha yaqin bo'lsa, yechim shuncha kam iteratsiyalar bilan olinadi. Iteratsiya jarayonining yaqinlashish tezligi esa o'z

navbatida berilgan sistema koeffisientlari matrisasining xususiyatiga bog'liq bo'ladi.

Albatta, iteratsiya usuli bilan tenglamalar sistemasining taqribiy yechimi topiladi. Agar sistema koeffisientlari va ozod hadlari aniq sonlardan iborat bo'lsa, hisoblashlarni sonlarda verguldan keyin ixtiyoriy m ta xona aniqligida bajarish mumkin. Buning uchun, hisoblash amallari verguldan keyin $m+1$ ta xona aniqligida bajarilib, kerakli iteratsiyalar bajarilgach, $m+1$ xonadagi son yaxlitlanadi. Sistema koeffisientlari va ozod hadlar p aniqlikdagi sonlar bo'lsa, sistemani p dan katta aniqlikda yechish ma'noga ega emas. Bunda odatda sistema p dan katta bo'lmagan aniqlikda yechiladi. Misol sifatida iteratsiya usulining yaqinlashish shartlarini ikkinchi tartibli sistema uchun keltiramiz.

5-teorema: Ikkinchi tartibli sistemaning yagona yechimi $\{a < x_1 < b, c < x_2 < d\}$ to'g'ri to'rtburchakda joylashgan bo'lsin. Agar bu to'g'ri to'rtburchakda quyidagi

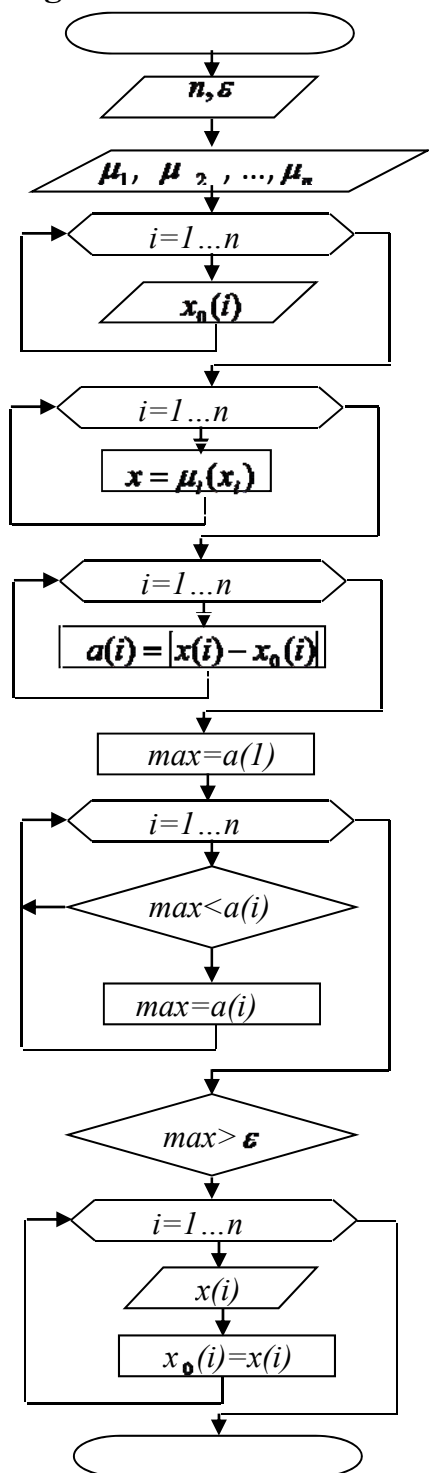
$$\left| \frac{\partial \varphi_1}{\partial x_1} \right| \leq p_1, \quad \left| \frac{\partial \varphi_1}{\partial x_2} \right| \leq q_1, \quad \left| \frac{\partial \varphi_2}{\partial x_1} \right| \leq p_2, \quad \left| \frac{\partial \varphi_2}{\partial x_2} \right| \leq q_2,$$

$$p_1 + p_2 < 1, \quad q_1 + q_2 < 1$$

tengsizliklar bajarilsa, iteratsiya jarayoni yaqinlashadi va nolinchi yaqinlashish sifatida to'g'ri to'rtburchakning ixtiyoriy nuqtasini olish mumkin.

Quyida iteratsiya usulining algoritmini blok-sxemalardagi ifodasi va dasturi keltirilmoqda. Ularning ishonchlilik darajasini aniqlash uchun esa aniq chiziqsiz sistemani yechish tashkil qilinadi.

Algoritm blok-sxemasi



Dastur matni

```

# Iterat sis
import math
N=2
EPS=0.001
vec, a, x, x0 = [0,0],[0,0],[0,0],[0,0]
i, k = 0,0
def f1(x):
    f1 = 4.6-math.pow(x[0],2)/10-
    math.log(x[1])/3
    return f1
def f2(x):
    f2 = 3.1-math.exp(-x[0])-
    math.sqrt(x[1])
    return f2
k=1
for i in range(N):
    x0[i] = float(input(f'X0[{i+1}]= '))
while True:
    x[0] = f1(x0)
    x[1] = f2(x0)
    for i in range(N):
        a[i] = abs(x[i]-x0[i])
    maxx = a[0]
    for i in range(N):
        if maxx < a[i]:
            maxx = a[i]
    if maxx > EPS:
        print(f'{k} ning yaqinlashishi',
        end=" ")
        for i in range(N):
            print(f'{x[i]:8.6f}', end=" ")
            x0[i]=x[i]
        print()
        k+=1
    else:
        break
  
```

Misol: Ushbu
$$\begin{cases} x_1 + \frac{x_1^2}{10} + \frac{\ln x_2}{3} = 4.6 \\ e^{-x_1} + \sqrt{x_2} + x_2 = 3.1 \end{cases}$$

chiziqsiz tenglamalar sistemasini oddiy iteratsiya usuli bilan 0,001 aniqlikda yeching.

Echish: Avvalo sistemaning ko‘rinishini o‘zgartirib olamiz, ya’ni ularni x_1 va x_2 larga nisbatan yechib olamiz:

$$\begin{cases} x_1 = 4,6 - \frac{x_1^2}{10} - \frac{\ln x_2}{3}; \\ x_2 = 3.1 - e^{-x_1} - \sqrt{x_2} \end{cases} \quad \text{U holda} \quad \begin{cases} \phi_1(x_1, x_2) = 4,6 - \frac{x_1^2}{10} - \frac{\ln x_2}{3} \\ \phi_2(x_1, x_2) = 3.1 - e^{-x_1} - \sqrt{x_2} \end{cases}$$

Endi qidirilayotgan o‘zgaruvchilar bo‘yicha hususiy hosilalar olinadi:

$$\frac{\partial \phi_1}{\partial x_1} = -\frac{x_1}{5}, \quad \frac{\partial \phi_1}{\partial x_2} = -\frac{1}{3x_2}, \quad \frac{\partial \phi_2}{\partial x_1} = e^{-x_1}, \quad \frac{\partial \phi_2}{\partial x_2} = -\frac{1}{2\sqrt{x_2}}$$

Aytaylik, boshlang‘ich yaqinlashish x_1 va x_2 lar bo‘yicha $[1,4]$ kesmada bo‘lsin.

U holda hosilalar uchun quyidagi tengsizliklar o‘rinli bo‘ladi:

$$\begin{aligned} \left| \frac{\partial \phi_1}{\partial x_1} \right| &\leq \frac{4}{5} = 0,8 = p_1 & \left| \frac{\partial \phi_1}{\partial x_2} \right| &\leq \frac{1}{12} = 0,083 = q_1 \\ \left| \frac{\partial \phi_2}{\partial x_1} \right| &\leq \frac{1}{e^4} \approx 0.0069 = p_2 & \left| \frac{\partial \phi_2}{\partial x_2} \right| &\leq \frac{1}{4} = 0,25 = q_2 \end{aligned}$$

Demak, qarayotgan kvadratda:

$$p_1 + p_2 = 0,8 + 0.0069 = 0.8069 < 1$$

$$q_1 + q_2 = 0,083 + 0,25 = 0.333 < 1$$

yaqinlashish shartlari bajariladi.

U holda dastlabki yaqinlashish sifatida $x_1^{(0)} = 3,5$ $x_2^{(0)} = 1,7$ ni olib, keyingi yaqinlashishlarni oddiy iteratsiya usuliga mos dastur ta’minoti yordamida aniqlaymiz.

Taqribiy yechimga n nchi yaqinlashish	x_1	x_2
	3.5	1.7
1	3.468951	1.635999
2	3.232553	1.789788
3	3.361027	1.722714
4	3.289049	1.752779
5	3.331148	1.738785
6	3.305950	1.745618
7	3.321367	1.742117
8	3.311819	1.744004
9	3.317791	1.742943
10	3.314034	1.743562

Natijalardan ko‘rinib turibdiki, berilgan chiziqsiz tenglamalar sistemasining $x_1^{(0)} = 3,5$ $x_2^{(0)} = 1,7$ dastlabki yaqinlashish bilan olingan $0,001$ aniqlikdagi yechimi $x_1 = 3,315$ va $x_2 = 1,743$ ga teng. Albatta, aniqlikni oshirish imkoniyati ε ning qiymatiga bog‘liq ravishda har doim mumkin va bu zamonaviy hisoblash mashinasida hisoblash vaqtini biroz orttiradi xolos.



Nazorat savollari

1. Iteratsiya usuli uchun dastlabki yaqinlashish qanday aniqlanadi?
2. Iteratsion usullarda yechimga yaqinlashish formulasi qanday hosil qilinadi?
3. Iteratsiya usulining yaqinlashish tezligi qaysi omilga bog‘liq?
4. Iteratsion jarayon qachon to‘xtatiladi?
5. Iteratsiya usulida har doim yechimga yaqinlashish holati sodir bo‘ladimi?

Nyuton usuli, Yakobi matrisasi, dastlabki yaqinlashish, usulning xatoligi, yechimga yaqinlashish tezligi, usulning ishchi algoritmi, dastur matni.

$$x = x^{(k)} + \varepsilon^{(k)} \quad (2.10)$$
$$f(x^{(k)} + \varepsilon^{(k)}) = 0 \quad (2.11)$$
$$f(x^{(k)} + \varepsilon^{(k)}) \approx f(x^{(k)}) + f'(x^{(k)}) \varepsilon^{(k)} \quad (2.12)$$

Bu tenglamalarni koordinatalar bo'yicha yoyib yozib,

$$\left\{ \begin{aligned} f_1(x^{(k)} + \varepsilon^{(k)}) &\approx f_1(x^{(k)}) + \frac{\partial f_1(x^{(k)})}{\partial x_1} \cdot \varepsilon_1^{(k)} + \frac{\partial f_1(x^{(k)})}{\partial x_2} \cdot \varepsilon_2^{(k)} + \dots + \frac{\partial f_1(x^{(k)})}{\partial x_n} \cdot \varepsilon_n^{(k)} = 0 \\ f_2(x^{(k)} + \varepsilon^{(k)}) &\approx f_2(x^{(k)}) + \frac{\partial f_2(x^{(k)})}{\partial x_1} \cdot \varepsilon_1^{(k)} + \frac{\partial f_2(x^{(k)})}{\partial x_2} \cdot \varepsilon_2^{(k)} + \dots + \frac{\partial f_2(x^{(k)})}{\partial x_n} \cdot \varepsilon_n^{(k)} = 0 \\ \vdots \\ f_n(x^{(k)} + \varepsilon^{(k)}) &\approx f_n(x^{(k)}) + \frac{\partial f_n(x^{(k)})}{\partial x_1} \cdot \varepsilon_1^{(k)} + \frac{\partial f_n(x^{(k)})}{\partial x_2} \cdot \varepsilon_2^{(k)} + \dots + \frac{\partial f_n(x^{(k)})}{\partial x_n} \cdot \varepsilon_n^{(k)} = 0 \end{aligned} \right.$$

51

$$[f'(x)] = [W(x)] = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

Yakobi matrisasini kiritib, uni

$$W(x^{(k)})\varepsilon^{(k)} = -f(x^{(k)}) \quad (2.13)$$

shaklga keltiramiz. Bu esa $\varepsilon^{(k)}$ larga nisbatan chiziqli algebraik tenglamalar sistemasidan iborat. Noma'lumlar oldidagi koeffisientlar $W(x^{(k)})$ -Yakobi matrisasini tashkil qiladi. Bu matrisani xos emas yani,

$$\det [W(x^{(k)})] \neq 0$$

deb faraz qilaylik. Unda (2.13) sistemaning yechimi

$$\varepsilon^{(k)} = -[W^{-1}(x^{(k)})]f(x^{(k)})$$

dan iborat bo'ladi.

(2.10) ni hisobga olib, yechimning $k+1$ yaqinlashishini

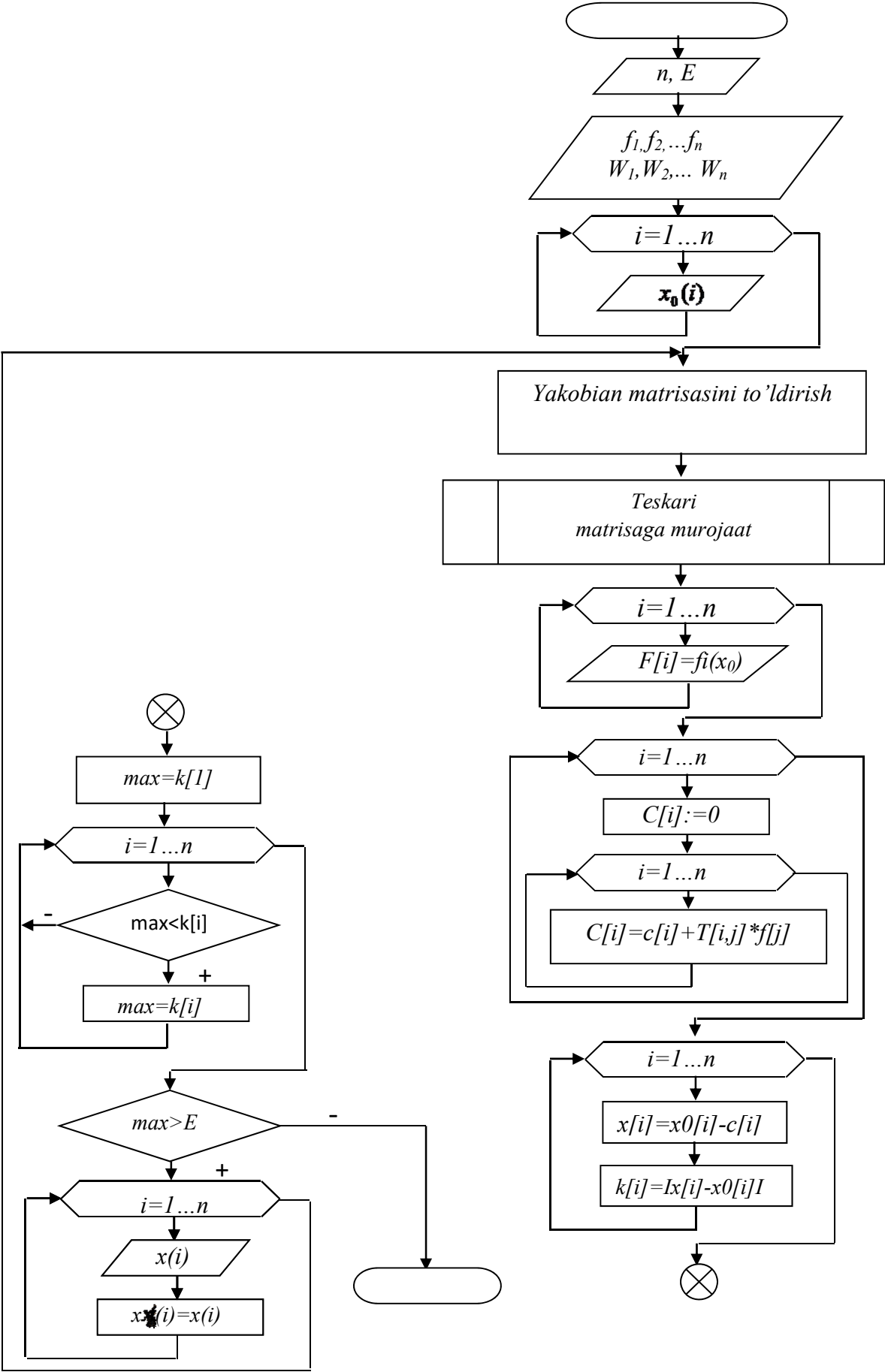
$$x^{(k+1)} = x^{(k)} - W^{-1}(x^{(k)})f(x^{(k)}), \quad k = 0, 1, \dots \quad (2.14)$$

ko'rinishda aniqlaymiz.

Nolinchi yaqinlashish sifatida ixtiyoriy $x^{(0)}$ vektorni olish mumkin.

Quyida usul algoritmining umumiy-strukturaviy blok-sxemasi va algoritmga mos ishchi dastur keltirilgan.

lgoritm blok-sxemasi



Dastur matni

```
# Nyuton
import math
N = 2
EPS = 0.001
mas = [[1, 1], [1, 1]]
vec = [1, 1]
x, x0, c, f, k = vec*1, vec*1, vec*1, vec*1, vec*1
w, t = mas*1, mas*1
maxx, i, j, p = 0, 0, 0, 0
def w1(x):
    w1 = 1+x[0]/5
    return w1
def w2(x):
    w2 = 1/(3*x[1])
    return w2
def w3(x):
    try:
        w3 = -1 * math.log(-x[0])
    except:
        w3 = 0
    return w3
def w4(x):
    w4 = 1/(2*math.sqrt(x[1]))+1
    return w4
def f1(x):
    f1 = x[0]+math.pow(x[0]/10+math.log(x[1])/3-4.6, 2)
    return f1
def f2(x):
    f2 = math.exp(-x[1])+math.sqrt(x[1])+x[1]-3.1
    return f2

def Gauss(a, b, n, x):
```

Ishlab chiqilgan usul algoritmining to'g'riligi va u asosida yaratilgan dasturning ishga sozligini tekshirish uchun yana aniq bir misol ko'rib chiqaylik.

Misol: Yuqorida iteratsiya usuli bilan yechilgan ikkinchi tartibli, chiziqsiz tenglamalar sistemasini Nyuton usuli bilan yeching.

Yechish: Avvalo berilgan sistemani

$$f_1(x_1, x_2) = x_1 + \frac{x_1^2}{10} + \frac{\ln x_2}{3} - 4,6 = 0$$

$$f_2(x_1, x_2) = e^{-x_1} + \sqrt{x_2} + x_2 - 3,1 = 0$$

ko‘rinishida yozib olamiz.

Nolinchi yaqinlashish sifatida $x(0) = (1; 1)$ ni olaylik. YAkobi matrisasining shu nuqtadagi qiymatini topish uchun mos hususiy hosilalarni olamiz:

$$\frac{\partial f_1}{\partial x_1} = 1 + \frac{x_1}{5}, \quad \frac{\partial f_1}{\partial x_2} = \frac{1}{3x_2}, \quad \frac{\partial f_2}{\partial x_1} = -e^{-x_1}, \quad \frac{\partial f_2}{\partial x_2} = \frac{1}{2\sqrt{x_2}} + 1.$$

Hosilalarni $x_1^{(0)} = 1, \quad x_2^{(0)} = 1$ boshlang‘ich qiymatlar uchun hisoblab

$$W(x^{(0)}) = \begin{pmatrix} 1.2 & 0.33 \\ -0.37 & 1.5 \end{pmatrix}$$

Yakobian matrisasini topamiz. Bu matrisa uchun $\det W(x^{(0)}) = 1,9221 \neq 0$ sharti bajarilganligi tufayli $W^{-1}(x^{(0)})$ ni topish mumkin. Keyingi yaqinlashishlar esa

$$x^{(k+1)} = x^{(k)} - W^{-1}(x^{(k)})f(x^{(k)})$$

formula orqali aniqlanadi. Har safar $\max |x^{(k+1)} - x^{(k)}| < \varepsilon$ yaqinlashish sharti yordamida yechimning kerakli aniqliqda ekanligini tekshirib boriladi. Yuqorida ishlab chiqilgan algoritm yordamida olingan natijalar quyidagi jadvalda berilgan.

Taqribiy yechimga n nchi yaqinlashish	X_1	X_2
0	3.5	1.7
1	3.603709	2.126648
2	3.317086	1.739434
3	3.315488	1.743330

Natijalarni tahlil qilib, shunday xulosalarga kelamiz. Nyuton usuli iteratsiya usuliga nisbatan yechimga juda tez yaqinlashadi. Oddiy iteratsiya usulida yuqorida 17 ta qadam bilan topilgan yechim bu usulda 3 tagina qadamda olindi va tabiiyki, bu hisoblash mashinasining juda oz vaqt sarflashiga olib keldi. Demak, Yakobi determinantini to‘g‘ri qurib olinsa, Nyuton usuli chiziqsiz tenglamalar sistemasini yechishdagi eng maqbul usullardan deya olamiz.



Nazorat savollari

1. Nyuton usulida dastlabki yaqinlashish qanday aniqlanadi?
2. Nyuton usulida tenglamalar sistemasini yechish uchun qo'llanadigan Yakobi matrisasi qanday tuziladi?
3. Nyuton usulida yechimga yaqinlashish formulasi qanday munosabatlar asosida shakllantiriladi?

XULOSA

Mazkur bobda chiziqsiz algebraik va transsendent tenglamalar, ularni yechish imkoniyatlari, tenglamani yechimlari soni, taqribiy yechish usullari haqida umumiy ma'lumotlar keltirildi.

Chiziqsiz tenglamani yechish uchun uning ildizlarini ajra-tishning analitik, grafik va algoritmik usullari, ularning mohiyati bayon qilindi va misollar bilan tushuntirildi.

Chiziqsiz tenglamani yechishning oraliqni teng ikkiga bo'lish, urinmalar, iteratsiya va vatarlar usullarining mohiyati, geometrik ma'nolari, yechimga yaqinlashish formulasi, usullarning xatoliklarini baholash jarayoni tavsiflanib, usullarga mos ishchi formulalar va dastur matnlari tavsiya qilindi. Har bir usul uchun na'muna sifatida misollar keltirildi va natijalar yaratilgan dastur ta'minoti yordamida olinib, tahlil etildi.

Chiziqsiz sistemani yechishning iteratsiya va Nyuton usullari uchun ishchi algoritm va dastur ta'minotlari ishlab chiqildi. Aniq misollar uchun usullarga mos dastlabki yaqinlashishlar tanlanib, natijalar olindi. Har ikkala usuldan olingan natijalar tahlil etildi.



Bobga doir muammoli vaziyatlar!

- + Chiziqsiz tenglamani taqribiy yechish uchun dastlab oraliqni ajratish shart deb o'ylaysizmi? Oraliqni ajratmay turib tenglamani yechish bo'yicha tavsiyalar bera olasizmi?
- + Ildiz yotgan $[a, b]$ oraliqni to'g'riligini tekshiruvchi asosiy shartda ko'paytma $f(a)f(b)=0$ tenglikni qanoatlantirsa, qanday mulohazalar yuritiladi?
- + Nima uchun taqribiy ildiz yotgan oraliqning chetki nuqtalarida funksiyaning turli ishorali bo'lishi va shu oraliqda birinchi tartibli xosilaning ishorasini o'zgarmas bo'lishligi talab qilinishini tushuntirib bera olasizmi? Aksincha bo'lsachi?
- + Ildiz yotgan oraliqni ajratishning qaysi usulini eng samarali va qulay deb hisoblaysiz? Nima uchun?
- + Ildiz yotgan oraliqni katta qilib tanlab olish qanday «salbiy oqibat»larga olib kelishi mumkin? Fikringizni ifodalovchi misollar keltira olasizmi?
- + Oraliqni teng ikkiga bo'lish usuli juda sodda bo'lganligi uchun tenglamani yechishda hech ikkilanmay shu usulni tanlagan bo'larmidingiz? Yoki bunga monelik qiluvchi biror sabab bormi?
- + Yechimga yaqinlashishda bir muncha ustunlikka ega bo'lgan urinmalar va vatarlar usullarini qo'llashdagi asosiy qiyinchilik qaerda paydo bo'ladi? Bu usullarni har doim ishlatish mumkin deb o'ylaysizmi?
- + Iteratsiya usulida yechimga yaqinlashish jarayonini nima uchun xosilaning qiymati bilan bog'lanishini tushuntirib bera olasizmi? yechimdan uzoqlashish jarayonini to'g'ri "boshqarish" uchun hisoblash algoritmidan muayyan shartlar kiritish zarur deb hisoblaysizmi?
- + Vegsteyn usulida iteratsiya usuliga qaraganda juda samarali bo'lgan xususiyatni ko'rsata olasizmi? Nima uchun bu usulda aksariyat hollarda yechimga yaqinlashish imkoni mavjudligining sababini bilasizmi?

- ✚ Chiziqsiz tenglamani taqribiy yechish zaruriyati tug'ilganda qaysi usulni tanlagan bo'lar edingiz? Qaysi usulda aniqlik yuqori bo'ladi deb o'ylaysiz?
- ✚ Tenglamalardagi singari tenglamalar sistemasini yechishda ham dastlabki yaqinlashishni tanlashda muayyan shartlarning bajarilishi yechimga yaqinlashishdagi asosiy omil sifatida qaraladimi? Dastlabki yaqinlashish izlanayotgan yechimga yaqinlashish tezligiga ta'sir etadimi?
- ✚ Tenglamalar sistemasini samarali yechishda yaqinlashish tezligi masalasi eng muhim omillardan ekanligining sababini ko'rsata olasizmi?
- ✚ Iteratsion jarayonining davomiyligi nima uchun aynan oldingi va keyingi yaqinlashishlarga mos tavofut miqdorlarning max qiymatini muayyan aniqlikka tekshirish orqali aniqlanishini tushuntirib bera olasizmi?
- ✚ Nyuton usulining iteratsiya usuliga ko'ra tezroq yechimga yaqinlashishiga nima sabab deb o'ylaysiz? Yakobi matrisasini har doim ham tuzib bo'ladimi? U yechimga yaqinlashishni doimo kafolatlaydimi?

3-BOB. CHIZIQLI ALGEBRAIK TENGLAMALAR SISTEMASINI YECHISH

Hayotda kuzatilayotgan har bir hodisalarning ro'y berish hollari ma'lum bir qonuniyatlarga bo'ysunadi. Bunday xodisalarning ro'y berishi aniq hisobga olingan faktorlar bilan bog'liq bo'lib, ularning sonli munosabatlari ma'lum bir aniq harakterga ega bo'ladi. Shunday munosabatlardan biri tenglamalar sistemasidir.

Chiziqli algebraik tenglamalar sistemasi(CHATS)ni yechish usullari sonli usullar orasida muhim o‘rin tutadi. Buning asosiy sababi, xalq ho‘jaligining juda ko‘p masalalari bunday sistemalarni yechish bilan bog‘liqdir. Shu bois ushbu bobda

CHATSni yechish usullari, aniq va iteratsion usullarning mohiyati, ularni hisoblash algoritmlari, dastur ta`minotlari keltirildi. Har bir hisoblash usuliga mos masalalar na`muna sifatida yechilib, natijalar tahlil etildi.

1-§. Chiziqli algebraik tenglamalar sistemasini yechishning umumiy qoidalari

Tayanch soʻz va atamalar



Chiziqli algebraik tenglamalar sistemasi, diagonal matrisa, simmetrik matrisa, yuqori uchburchak matrisa, uch diagonalli matrisa, sistemaning yechimi, sistemani yechish usullari, to'g'ri usullar, iteratsion usullar.

Chiziqli algebraning sonli usullariga chiziqli algebraik tenglamalar sistemasini yechish, matrisaning teskarisini topish, determinantlar hisoblash kabi sonli usullar kiradi.

Ushbu n -tartibli n ta chiziqli algebraik tenglamalarning sistemasi berilgan bo'lsin:

$$\left\{ \begin{array}{l} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \dots\dots\dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{array} \right. \quad (3.1)$$

Bu yerda $a_{ij}(i, j = \overline{1, n})$ lar ma'lum sonlardan iborat bo'lib, noma'lumlarning koeffisientlari deyiladi, $x_1, x_2, \dots, x_n$ - noma'lumlar, $b_1, b_2, \dots, b_n$ - (3.1) sistema tenglamalarining ozod hadlari, ular ham ma'lum sonlardan iborat.

Quyidagi belgilashlarni kiritamiz:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}, \quad X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}, \quad B = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix}. \quad (3.2)$$

Matrisa va vektorni bir-biriga ko'paytirish xossasidan foydalanib, (3.2) belgilashlarni hisobga olgan holda (3.1) sistemani matrisa ko'rinishda yozamiz:

$$AX = B \quad (3.3)$$

Umuman olganda A kvadrat matrisa turli xil ko'rinishlarda bo'lishi mumkin:

- 1) agar matrisadagi faqat $a_{ii}(i = \overline{1, n})$ hadlar noldan farqli bo'lib, boshqa hadlarning hammasi nolga teng bo'lsa, A matrisa diagonal matrisa deyiladi;
- 2) agar $a_{ij} = a_{ji}(i, j = \overline{1, n})$ shart bajarilgan bo'lsa, A matrisani simmetrik matrisa deyiladi;
- 3) agar matrisaning diagonal va undan yuqorida turgan hadlar noldan farqli, qolgan hadlar esa nolga teng bo'lsa, bu matrisani yuqori uchburchak matrisa deyiladi. Xuddi shunday ta'rifni pastki uchburchak matrisaga nisbatan ham berish mumkin;
- 4) agar matrisaning asosiy chap diagonal va unga parallel bo'lgan ikkita qo'shni diagonallardagi elementlarga noldan farqli, boshqa elementlar esa nolga teng bo'lsa, matrisani uch diagonalli matrisa deb ataladi.

Chiziqli algebraik tenglamalar sistemasini yechish deb (3.1) yoki (3.3) sistemalardan $x_1, x_2, \dots, x_n$ noma'lumlarni topishga aytiladi. Topilgan $x_1, x_2, \dots, x_n$ qiymatlar (3.1) yoki (3.3) sistemalariga qo'yilganda tenglamalarni ayniyatga aylantirsa, ular sistemaning yechimi deyiladi.

Sistema yagona yechimi mavjudligining zaruriy va yetarli sharti A kvadrat matrisa determinantining noldan farqli bo'lishidir, ya'ni

$$D = \det A = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix} \neq 0. \quad (3.4)$$

Agar $D = 0$ bo'lsa, sistemalar maxsus sistemalar deyiladi va ularning yechimi yoki mavjud emas, yoki cheksiz ko'p bo'ladi. $D \approx 0$ bo'lgan holda esa sistema yomon shartlangan deb hisoblanadi va bunday sistemalarni yechish masalasiga alohida ahamiyat bilan yondoshish lozim bo'ladi. Chiziqli algebraik tenglamalar sistemasini yechish usullari ikkita guruhga bo'linadi: to'g'ri (aniq) va iteratsion (taqribiy) usullar.

To'g'ri usullar yordamida sistemaning yechimiga chekli sondagi aniq arifmetik amallarni bajarish orqali erishiladi. Bu usullar keng sinfdagi sistemalarni yechish imkoniyatiga ega. Lekin, shu bilan birga, ular ayrim kamchiliklardan ham holi emas. Masalan, ular kompyuterda ishlatilganda hotira qurilmasida sistema koeffisientlari va ozod hadlarning barchasi saqlanishi kerak. Bundan tashqari, usullar asosida yotuvchi algoritmlar aniq bo'lishiga qaramasdan yechim ma'lum darajada taqribiy topiladi. Chunki, yaxlitlash xatoliklari ketma-ket bajariluvchi hisoblash bosqichlarida doimo jamlanib boradi. Ayniqsa, yuqori tartibli va yomon shartlangan sistemalar uchun bu butunlay yaroqsiz yechim olinishiga sabab bo'lishi mumkin. Shuning uchun, to'g'ri usullar yaxshi shartlangan, past tartibli, hadlari siyrak bo'lmagan matrisali sistemalarni yechishda ishlatiladi.

Iteratsion usullar - bu ketma-ket yaqinlashish usullaridir. Bu usullar to'g'ri usullarga nisbatan murakkabroq. Lekin, ko'p hollarda iteratsion usullarni ishlatish ma'qulroqdir. Chunki, bu usullarni ishlatganda kompyuterning xotira qurilmasida sistema matrisasining barcha hadlarini saqlab turishga hojat yo'q. Undan tashqari, xatoliklar ham iteratsion usullarda jamlanib bormaydi. Har bir iteratsiya qadamida hisob-kitob go'yo yangidan boshlangandek davom etib ketadi. Lekin, iteratsion usullarni hamma vaqt ham ishlataverish mumkin emas. Buning uchun, ma'lum shartlar bajarilishi kerak. Aks holda, iteratsiya jarayoni uzoqlashuvchi bo'lib, uning yordamida yetarli aniqlikdagi yechimni olish imkoniyati bo'lmaydi.

Bu shartlar haqida batafsil ma'lumotlar iteratsion usullar berilgan paragrafda keltirilgan. To'g'ri usullarga Kramer, Gauss, bosh elementlar, kvadrat ildizlar va shu kabi usullar kiradi. Iteratsion usullarga esa oddiy iteratsiya, Zeydel, relaksasiya va shu kabi usullar kiradi.

Nazorat savollari

1. Chiziqli algebraik tenglamalar sistemasi umumiy holda qanday ifodalanadi?
2. Qanday matrisaga diagonal matrisa deyiladi?
3. Simmetrik matrisaning ta'rifini bilasizmi?
4. Yuqori uchburchak matrisani tavsiflang.
5. Qanday matrisalarni uch diagonalli matrisalar deb yuritiladi?
6. Sistemaning yechimi nima?
7. Determinantni hisoblashning sistemaning yechimini aniqlashda qanday ahamiyati bor?
8. Qanday usullarni to'g'ri usullar deb ataladi?
9. Iteratsion usullarning mohiyatini bilasizmi?
10. To'g'ri va iteratsion usullarga qanday usullar kiradi?

2-§. Gauss usuli

Tayanch soʻz va atamalar



Gauss usuli, to'rtburchak sistema, uchburchak hol, to'g'ri yurish bosqichi, teskari yurish bosqichi, usulning ishchi algoritmi, dastur matni.

Gauss usuli bizga oddiy matematika kursidan ma'lum bo'lgan sistema noma'lumlarini ketma-ket yo'qotish usulining umumiy sxemasidan iboratdir. Bizga (3.1) ko'rinishidagi kabi ifodalangan quyidagi n -tartibli chiziqli algebraik tenglamalar sistemasi berilgan bo'lsin:

[illegible]

Gauss usuli ikki bosqichdan iborat: to‘g‘ri yurish va teskari yurish. Usulning to‘g‘ri yurish bosqichida (3.5) ko‘rinishdagi «to‘rtburchak» sistema «yuqori uchburchak» holiga keltiriladi. Teskari yurish bosqichida esa, xosil qilingan «uchburchak» sistema eng oxirgi tenglamasidan boshlab yuqoriga qarab ketma-ket yechib boriladi va sistemaning sonli yechimlari xosil qilinadi.

To'g'ri yurish bosqichi. Faraz qilaylik , (3.5) sistemadagi birinchi tenglamaning yetakchi elementi $a_{11}^{(1)} \neq 0$ bo'lsin, aks holda sistemadagi tenglamalarning o'rinlarini almashtirib, x_1 noma'lum oldidagi koeffitsienti noldan farqli bo'lgan tenglamani birinchi o'ringa ko'chiramiz. Sistemadagi birinchi tenglamaning barcha koeffitsientlarini $a_{11}^{(1)}$ ga bo'lib,

$$x_1 + a_{12}^{(2)} x_2 + \dots + a_{1n}^{(2)} x_n = b_1^{(2)} \quad (3.6)$$

ni hosil qilamiz, bu yerda $a_{1j}^{(2)} = a_{1j}^{(1)} / a_{11}^{(1)}$, $b_1^{(2)} = b_1^{(1)} / a_{11}^{(1)}$. (3.6) tenglamadan foydalanib (3.5) sistemaning qolgan tenglamalaridan x_1 ni yo‘qotish mumkin. Buning uchun, (3.6) tenglamani ketma-ket $a_{21}^{(1)}, a_{31}^{(1)}, \dots, a_{n1}^{(1)}$ larga ko‘paytirib, mos ravishda (3.5) sistemaning ikkinchi, uchinchi va x.k tenglamalaridan ayiramiz va yana shu tenglamalar sifatida yozib olamiz. Natijada, quyidagi sistema xosil bo‘ladi:

$$\begin{cases} a_{22}^{(2)}x_2 + \dots + a_{2n}^{(2)}x_n = b_2^{(2)} \\ \vdots \\ a_{n2}^{(2)}x_2 + \dots + a_{nn}^{(2)}x_n = b_n^{(2)} \end{cases} \quad (3.7)$$

bu yerda $a_{ij}^{(2)}$ koeffisientlar va $b_i^{(2)}$ ozod hadlar

$$a_{ij}^{(2)} = a_{ij}^{(1)} - \frac{a_{i1}^{(1)}}{a_{11}^{(1)}} a_{1j}^{(1)}, \quad b_i^{(2)} = b_i^{(1)} - \frac{a_{i1}^{(1)}}{a_{11}^{(1)}} b_1^{(1)}$$

formula yordamida hisoblanadi.

Yuqoridagi kabi amallarni (3.7) sistema uchun ham qo‘llaymiz va bu jarayonni ketma-ket davom ettirib, berilgan (3.5) sistemaga teng kuchli quyidagi «uchburchak» sistemani hosil qilamiz:

[illegible]

Xosil qilingan bu uchburchak sistemaning koeffisientlari va ozod hadlari quyidagi formulalar bilan hisoblanadi:

$$a_{ml}^{(k+1)} = a_{ml}^{(k)} - \frac{a_{mk}^{(k)}}{a_{kk}^{(k)}} a_{kl}^{(k)} \quad (3.9)$$

$$b_m^{(k+1)} = b_m^{(k)} - \frac{a_{mk}^{(k)}}{a_{kk}^{(k)}} b_k^{(k)}$$

bu yerda $k < m, l \leq n, 1 \leq k < n-1$

Teskari yurish bosqichi. Endi (3.8) uchburchak sistemaning oxirgi tenglamasidan boshlab yuqoriga qarab yurib sistema noma'lumlari $x_1, x_2, \dots, x_n$ larni ketma-ket topish mumkin:

$$\left\{ \begin{array}{l} x_n = b_n^{(n)} / a_{nn}^{(n)} \\ x_{n-1} = (b_{n-1}^{(n-1)} - a_{n-1,n}^{(n-1)} * x_n) / a_{n-1,n-1}^{(n-1)} \\ \dots\dots\dots \\ x_1 = (b_1^{(1)} - a_{12}^{(1)} x_2 - a_{13}^{(1)} x_3 - \dots - a_{1n}^{(1)} x_n) / a_{11}^{(1)} \end{array} \right. \quad (3.10)$$

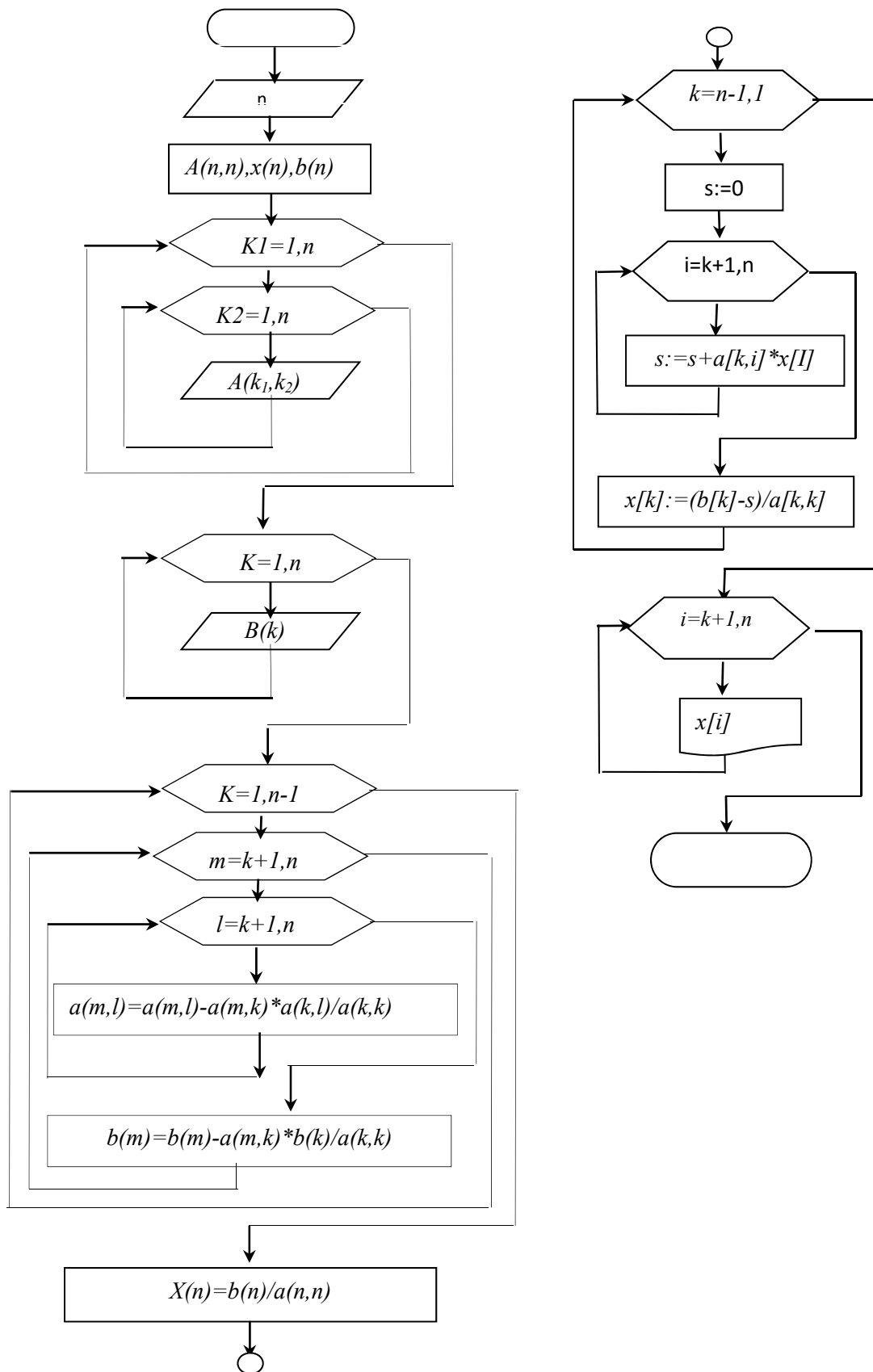
(3.10) ko‘rinishdagi sistema yechimlarini aniqlash formulalarini quyidagi ixcham ko‘rinishda yozish mumkin:

$$x_n = \frac{b_n^{(n)}}{a_{nn}^{(n)}}, \quad x_k = \frac{1}{a_{kk}^{(k)}} (b_k^{(k)} - \sum_{i=k+1}^n a_{ki}^{(k)} * x_i) \quad (3.11)$$

$$k = n-1, n-2, \dots, 1$$

Shunday qilib, ixtiyoriy n -tartibli chiziqli algebraik tenglamalar sistemasini ko'rsatib o'tilgan algoritm bo'yicha yechish mumkin. Faqat yetakchi hadlar $a_{kk}^{(k)}$ ni noldan farqliligiga yoki uni moduli bo'yicha eng kattaligiga erishish kerak. Buning uchun, Gauss usulini bosh hadni tanlash yo'li bilan qo'llaniladi, ya'ni noma'lum yo'qotiladigan ustundan moduli bo'yicha eng katta koeffisientli tenglama ishchi tenglama sifatida tanlab olinadi.

Gauss usulining algoritmining blok-sxemasi



Gauss usuli algoritmining dastur matni

```
#programm m1 Gauss usuli
N = 2
k1, k2, k = 0, 0, 0
a = [[0, 0], [0, 0]]
x, b = [0, 0], [0, 0]
for k1 in range(N):
    for k2 in range(N):
        a[k1][k2] = float(input(f"A[{k1}][{k2}] = "))
for k1 in range(N):
    b[k1] = float(input(f"B[{k1}] = "))
for k in range(N):
    for m in range(k+1, N):
        for l in range(k+1, N):
            a[m][l] = a[m][l] - a[m][k] * a[k][l] / a[k][k]
            b[m] = b[m] - a[m][k] * b[k] / a[k][k]
x[N-1] = b[N-1] / a[N-1][N-1]
for k in range(N-1, 0, -1):
    s = 0
    for i in range(k+1, N):
        s += a[k][i] * x[i]
    x[k] = (b[k] - s) / a[k][k]
for k in range(N):
    print(f"x[{k}] = ")
```

Dastur natijasi va uning tahlili.

Yuqoridagi algoritmnining to'g'riligini tekshirish uchun quyidagi sistemani yechishni tashkil qilamiz.

$$\left. \begin{aligned} 1,1x_1 + 2,3x_2 + 3,4x_3 - 2,0x_4 &= 6,5, \\ 2,8x_1 - 1,2x_2 - 2,3x_3 - 3,9x_4 &= 8,8, \\ 3,9x_1 + 2,8x_2 - 1,3x_3 + 2,8x_4 &= 4,1, \\ 2,7x_1 - 3,6x_2 + 2,6x_3 + 1,7x_4 &= -8,7. \end{aligned} \right\}$$

Gauss usuliga mos dastur ta'minotini ishlatib ko'rib, quyidagi natijalarga ega bo'lamiz.

$$x_1 = 2,6372176354E + 00 \quad x_2 = 7,93222977661E - 01$$

$$x_3 = -6,2607093330E - 01 \quad x_4 = -1,8050300306E + 00$$

Gauss usuli to'g'ri usullar guruhiga kirsa ham, haqiqiy sonlar ustida bajarilgan amallar tufayli ba'zi bir hisoblash xatoliklari kelib chiqishi mumkin. Shu xatolikni aniqlash uchun quyidagi formuladan foydalanamiz:

$$R_i = \sum_{j=1}^n x_j a_{ij} - b_i$$

Natijada quyidagi xisoblash xatoliklari kelib chiqadi:

$$R_1 = 0,0000000000 \quad E + 00 \quad R_2 = 0,0000000000E + 00$$

$$R_3 = 2,1456221112 \quad E - 11 \quad R_4 = 0,0000000000 \quad + 00$$

Xatolik miqdorining juda kichikligi algoritm va dasturning to'g'riligini va ishlatish uchun yaroqliligini ko'rsatadi.



Nazorat savollari

1. Gauss usuli necha bosqichdan iborat?
2. To'g'ri yurish bosqichining vazifasini bilasizmi?
3. To'g'ri yurishda qaysi ishchi formulalardan foydalaniladi?
4. Teskari yurishning mohiyatini tushuntirib bering.
5. Teskari yurishda qo'llanadigan asosiy ishchi formulalar qaysi?

3-§. Yuqori tartibli determinantlarni hisoblash



Tayanch so'z va atamalar

Matrisaning determinanti, yuqori tartibli matrisa, yetakchi hadlar, determinantni hisoblash algoritmi, dastur matni.

Ma'lumki, yuqori tartibli determinantlarni hisoblashga asoslangan Kramer qoidasi yordamida ham chiziqli algebraik tenglamalar sistemasini yechish mumkin. Lekin, bu usuldan amalda deyarli foydalanilmaydi. Chunki n -tartibli sistemani yechish uchun $n^2 * n!$ tartibidagi arifmetik amallarni bajarishga to'g'ri keladi. Shuning uchun, sistemani yechishga asosan Gauss usulini qo'llaniladi. Bu usulda bajariladigan amallar soni $\frac{2}{3}n^3$ atrofida bo'ladi.

Agar ahamiyat bergan bo'lsangiz, oldingi paragrafda, sistemani Gauss usuli bilan yechganda, bir vaqtning o'zida noma'lumlar oldidagi koeffisientlardan

tuzilgan matrisaning determinan-tini hisoblash imkoniyati ham mavjud edi. Xuddi shu imkoniyatdan yuqori tartibli quyidagi

$$\det A = \begin{vmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & \dots & a_{2n}^{(1)} \\ \dots & \dots & \dots & \dots \\ a_{n1}^{(1)} & a_{n2}^{(1)} & \dots & a_{nn}^{(1)} \end{vmatrix} \quad (3.12)$$

n -tartibli determinantni hisoblashda foydalanamiz. Buning uchun, Gauss usulini (3.9) to'g'ri yurish bosqichi ishchi formulalaridan foydalanib (3.12) determinantni quyidagi «uchburchak» xoliga keltirib olinadi:

$$\det A = \begin{vmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & a_{nn}^{(n)} \end{vmatrix} \quad (3.13)$$

Oliy matematika kursidan ma'lumki, bunday determinant natijasi asosiy diagonaldagi yetakchi hadlar ko'paytmasiga teng:

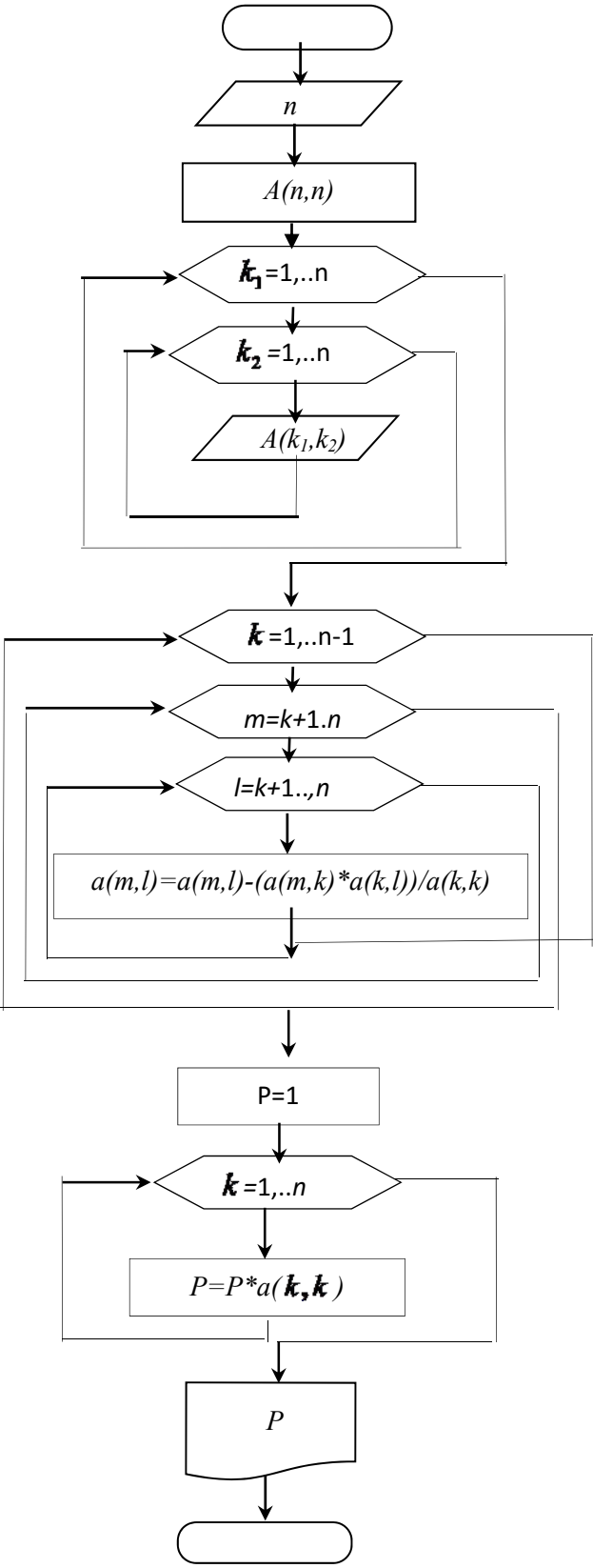
$$\det(A) = a_{11}^{(1)} \cdot a_{22}^{(2)} \cdot \dots \cdot a_{nn}^{(n)} = \prod_{k=1}^n a_{kk}^{(k)} \quad (3.14)$$

Bu yerda ham agar yetakchi hadlardan birortasi nolga teng bo'lsa, u xolda satr bo'yicha bosh hadni tanlash sxemasidan foydalanish mumkin. Lekin, bu xolda determinant ishorasini saqlab turish uchun yuqoridagi (3.14) formulani $(-1)^p$ ga ko'paytirish kerak, ya'ni

$$\det(A) = (-1)^p \prod_{k=1}^n a_{kk}^{(k)} \quad (3.15)$$

bu yerda r -determinant satrlarini almashtirishlar soni.

Determinant hisoblash algoritmining blok- sxemasi



Algoritmining dasturi

```

# program Determinant
N = 2
k1,k2,k,m,l,p=0,0,0,0,0,0
a=[[0,0],[0,0]]
for k1 in range(N):
    for k2 in range(N):
        a[k1][k2] =
float(input(f"A[{k1}][{k2}]= "))
for k in range(N-1):
    for m in range(k+1, N):
        a[m][l] = a[m][l]-
a[m][k]*a[k][l]/a[k][k]
p = 1
for k in range(N):
    p = p*a[k][k]
print(f"get A= {p}")
  
```

Shunday qilib, alohida ta'kidlash mumkinki, matrisalar determinantini Gauss usuli yordamida hisoblash uni bevosita algebraik to'ldiruvchi va minorlarga yoyib hisoblashga nisbatan ancha qulay va kam amallar bajarishni talab qiladi.

Yuqori tartibli determinantlarni Gauss usulining ishchi formulalari orqali hisoblash usuli algoritmining blok- sxemasi va dastur ta'minotlari keltirilmoqda. Yuqoridagi algoritmnining to'g'riligini tekshirish uchun avvalgi paragrafdagi sistemaning koeffisientlaridan tuzilgan determinantni yozib olamiz:

$$\det A = \begin{vmatrix} 1.1 & 2.3 & 3.4 & -2.0 \\ 2.8 & -1.2 & -2.3 & -3.9 \\ 3.9 & 2.8 & -1.3 & 2.8 \\ 2.7 & -3.6 & 2.6 & 1.7 \end{vmatrix}$$

Algoritmgaga mos hisoblash dasturini ishlatishdan olingan natija: $\det A = 794.4342$. Yuqori tartibli determinantni hisoblashning Gauss usuliga asoslangan algoritmnining hisoblash xatoligi qanday ekanligini tekshirish uchun yuqoridagi determinantni oliy matematika kursida o'rgangan an'anaviy- algebraik to'ldiruvchilar yordamida hisobladik.

Natija: 794,4345. Hisoblash xatoligi: 0,0001. Xatolik miqdorining unchalik katta emasligi algoritm va dasturning to'g'riligini va ishlatish uchun yaroqliligini ko'rsatadi.

Nazorat savollari

1. Yuqori tartibli determinantlarni hisoblash nima uchun kerak bo'ladi?
2. Determinantlarni hisoblashning qanday usullarini bilasiz?
3. Matrisaning qaysi hadlarini yetakchi hadlar deb ataymiz?
4. Determinantni hisoblovchi asosiy formula qaysi?
5. Yetakchi hadlardan birortasi nolga teng bo'lsa qanday choralar ko'riladi?

4-§. Teskari matrisani Gauss usuli bilan hisoblash

Tayanch soʻz va atamalar



Xosmas kvadrat matrisa, teskari matrisa, Kroneker belgisi, uchburchak sistemalar tizimi, teskari matrisani topish algoritmi, dastur matni.

Odatda, shunday bir qancha sistemalarni yechishga toʻgʻri keladiki, ularning koeffisientlari matrisasi bir xil, ozod hadlari esa turlicha boʻladi. Bunday xollarda koeffisientlar matrisasini teskarisini topib, ular orqali sistemalarni yechish maqsadga muvofiqdir. Bundan tashqari, turli xil statistik hisoblashlarda baʼzi parametrlarni baholash uchun teskari matrisalar katta ahamiyatga ega. Umuman olganda, turli xil amaliy ahamiyatga ega boʻlgan masalalarni yechishda teskari matrisalar bilan ishlashga toʻgʻri keladi. Shuning uchun, bu paragrafda teskari matrisani topishni Gauss usuli yordamida tashkil etamiz.

Faraz qilaylik, bizga xosmas, yaʼni $\det A \neq 0$, kvadrat matrisa $A = [a_{ij}]$ ($i, j = \overline{1, n}$) berilgan boʻlsin.

Umumiy usulda teskari matrisa

$$A^{-1} = \frac{1}{\det A} \begin{bmatrix} A_{11} & A_{21} & \dots & A_{n1} \\ A_{12} & A_{22} & \dots & A_{n2} \\ \dots & \dots & \dots & \dots \\ A_{1n} & A_{2n} & \dots & A_{nn} \end{bmatrix}$$

koʻrinishda topiladi. Bunda A_{ij} ($i, j = \overline{1, n}$) hadi berilgan matrisani a_{ij} -hadining algebraik toʻldiruvchisi boʻlib, u shu had minori bilan $(-1)^{i+j}$ ning koʻpaytmasiga tengdir.

Maʼlumki, n -tartibli A kvadrat matrisaga teskari A^{-1} matrisa taʼrifga koʻra $A \cdot A^{-1} = E$ (E -birlik matrisa) shart bajariladi.

Gauss usuli yordamida A^{-1} matrisani topish uchun uning hadlarini $x_{i,j}$ ($i, j = \overline{1, n}$) deb belgilab olamiz va $A \cdot A^{-1} = E$ munosa-batdan foydalanamiz. U xolda

A^{-1} matrisaning noma'lum had-lariga nisbatan yozilgan n ta n -tartibli tenglamalar sistemasiga ega bo'lamiz:

$$\sum_{k=1}^n a_{i,k} \cdot x_{kj} = \delta_{ij} \quad (i, j = 1, n) \quad (3.16)$$

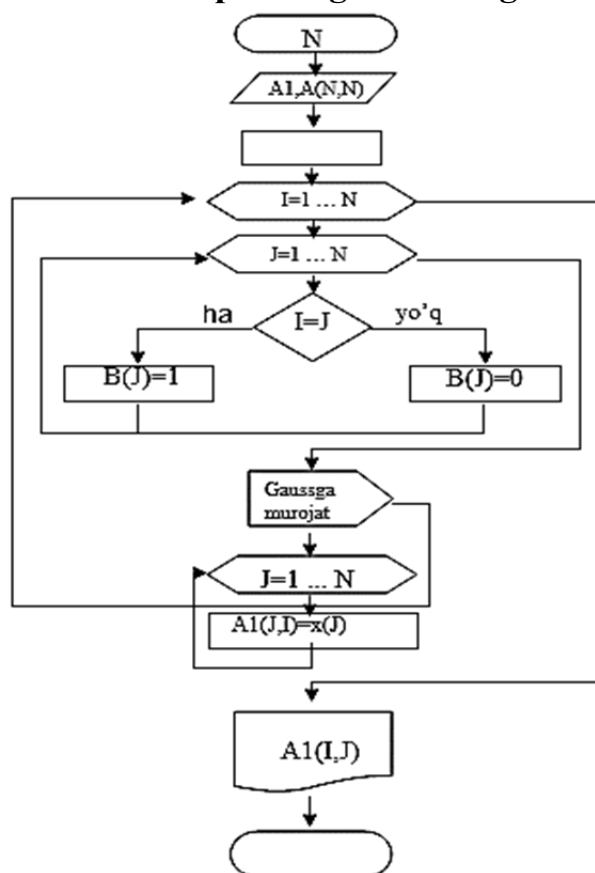
Bu formuladagi Kroneker belgisi quyidagicha aniqlanadi:

$$\delta_{i,j} = \begin{cases} 1 & \text{agar } i = j \text{ bo'lsa} \\ 0 & \text{agar } i \neq j \text{ bo'lsa} \end{cases}$$

Hosil qilingan n ta tenglamalar sistemalarining koeffitsientlar matrisalari bir xildir. Bu sistemalarda ozod hadlar farqli xolos. Shuning uchun, sistema koeffitsientlari matrisasini Gauss usuli yordamida bir marta uchburchak xoliga keltirib olamiz va so'ngra ularga moslashtirilgan turli xil ozod hadlar uchun uchburchak sistemalarni ketma-ket yechib, teskari matrisa hadlarini aniqlaymiz. Bunda asosiy vaqt uchburchak sistemani yechishga ketadi xolos.

Quyida teskari matrisani Gauss usuliga asoslanib hisoblanish algoritmining blok-sxemasi va dasturi keltirilgan.

Teskari matrisa topish algoritmining blok-sxemasi



```

import numpy as np
from numpy import linalg as LA
a=np.array([[8.2,-3.2,14.2,14.8],
            [5.6,-12,15,-6.4],
            [5.7,3.6,-12.4,-2.3],
            [6.8,13.2,-6.3,-8.7]])
print('Matritsa elementlari')
print(a)
s=np.linalg.inv(a)
print('Teskari matritsa')
print(s)
b=np.array([-8.4,4.5,3.3,14.3])
print('Ozod had elementlari')
print(b)
x=LA.solve(a,b)
print('x ning qiymatlari',x)

```

Olingan natijalarni tahlil qilish.

Yuqoridagi algoritmnining va dastur ta'minotining to'g'rili-gini tekshirish uchun quyidagi matrisani misol sifatida olamiz:

$$A = \begin{bmatrix} 1,1 & 2,3 & 5,5 & 2,3 \\ 3,3 & 1,3 & 1,8 & 3,1 \\ 2,6 & 4,3 & 1,1 & 1,7 \\ 1,1 & 3,8 & 2,9 & 2,7 \end{bmatrix}$$

Teskari matrisani topish algoritmiga mos dastur ta'mi-notini ishlatib ko'rib, quyidagi teskari matrisani hosil qilamiz.

$$A^{-1} = \begin{bmatrix} 0,16 & 0,16 & 0,34 & -0,54 \\ -0,05 & 0,19 & 0,20 & 0,12 \\ 0,31 & -0,07 & 0,04 & -0,21 \\ -0,33 & 0,27 & -0,47 & 0,64 \end{bmatrix}$$

Olingan natijaning to'g'riligini tekshirish uchun berilgan matrisani teskari matrisaga ko'paytirib ko'rdik. Natijada ko'paytma birlik matrisadan iborat bo'ldi. Bu jarayon dasturning tekshirish qismida o'z ifodasini topgan. Bu esa algoritm va dasturning to'g'riligini va ishlatish uchun yaroqliligini ko'rsa-tadi.



Nazorat savollari

1. Teskari matrisani topish uchun qaysi usuldan foydalaniladi?
2. Teskari matrisani hisoblashda ishlatiladigan Kroneker belgisi qanday aniqlanadi?
3. Teskari matrisani to'g'ri topilganligini qanday tekshirish mumkin?
4. Teskari matrisa yordamida tenglamalar sistemasini yechish qanday usullar guruhiga kiradi?

5-§. Iteratsiya usuli



Tayanch soʻz va atamalar

Diagonalelementlar, nolinchi yaqinlashish, iteratsiyalar soni, yaqinlashish tezligi, yaqinlashish shartlari, iteratsiya usulining hisoblash algoritmi, dastur matni.

Iteratsiya, ya'ni ketma-ket yaqinlashish usuli yuqori tartibli sistemalarni yechishda qo'1 keladi. Bundan tashqari, bu usul aniq usullardan farqli o'laroq xatoliklarning jamlanmasligi kabi xususiyatga egaki, bu yuqori tartibli sistemalarni yechishda hal qiluvchi omillardan biri bo'lishi mumkin.

Bizga (3.1) ko‘rinishidagi sistema berilgan bo‘lsin. Koeffisientlar matrisasi A da barcha diagonal hadlarni 0 dan farqli deb qaraymiz. (3.1) sistemaning birinchi tenglamasini x_1 ga nisbatan, ikkinchi tenglamasini x_2 ga nisbatan va hokazo n-tenglamasini x_n ga nisbatan yechib quyidagi berilgan tenglamalar sistemasiga teng kuchli sistemani hosil qilamiz:

[illegible]

Bunda agar $i \neq j$ bo'lsa, $\beta_i = \frac{b_i}{a_{ij}}$, $\alpha_{ij} = -\frac{a_{ij}}{a_{ii}}$, $a_{ii} \neq 0$; agar $i = j$ bo'lsa,

$$\alpha_{ij} = 0 \quad (i, j = \overline{1, n})$$

Quyidagi belgilashlar kiritib

$$\alpha = \begin{pmatrix} \alpha_{11} \alpha_{12} \dots \alpha_{1n} \\ \alpha_{21} \alpha_{22} \dots \alpha_{2n} \\ \alpha_{n1} \alpha_{n2} \dots \alpha_{nn} \end{pmatrix}, \quad \beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

(3.17) sistemani matrisa-vektorli ko‘rinishida yozib olishimiz mumkin:

$$X = \beta + \alpha X \quad (3.18)$$

(3.18) sistemani ketma-ket yaqinlashish usuli bilan yechamiz. Nolinchi yaqinlashish sifatida ixtiyoriy X vektor ustunni olishimiz mumkin. Masalan:

$$X^{(0)} = \beta \text{ yoki } X^{(0)} = 0$$

Nolinchi yaqinlashishni (3.18) sistemaning o‘ng tarafiga qo‘yib chap tarafda X vektor-ustunning birinchi yaqinlashishini topamiz:

$$X^{(1)} = \beta + \alpha X^{(0)}$$

Xuddi shuningdek,

$$X^{(2)} = \beta + \alpha X^{(1)}$$

ni topishimiz shu qonuniyat asosida yechimning ixtiyoriy $(k+1)$ -yaqinlashishini

$$X^{(k+1)} = \beta + \alpha X^{(k)} \quad (k = 0, 1, 2, \dots) \quad (3.19)$$

ko‘rinishida topishimiz mumkin.

Agar $X^{(0)}, X^{(1)}, \dots, X^{(k)}, \dots$ ketma-ketlik chekli limitga ega, ya‘ni

$$X = \lim_{k \rightarrow \infty} X^{(k)}$$

bo‘lsa, bu limit (3.1) yoki (3.17) sistemaning yechimi bo‘ladi.

Matrisa-vektorli ko‘rinishdagi yozuvdan koordinatalar ko‘rinishidagi yozuvga o‘tib, (3.19) formulani quyidagicha yozishimiz mumkin:

$$x_i^{(0)} = \beta_i, \\ x_i^{(k+1)} = \beta_i + \sum_{j=1}^n \alpha_{ij} x_j^{(k)}, \quad (3.20)$$

$$\alpha_{ii} = 0, \quad i = \overline{1, n}, \quad k = 0, 1, \dots$$

(3.19) yoki (3.20) formulalar bilan aniqlangan ketma-ket yaqinlashishlar asosida X yechimni topish iteratsiya usulini ma‘nosini tashkil qiladi.

Iteratsiya usuli ma'lum shartlar bajarilganda yetarli aniqlikdagi yechimni istalgan $X^{(0)}$ boshlang'ich yaqinlashishlarda topish imkoniyatini beradi. Bu shartlar yaqinlashish shartlari deyilib, ular to'g'risida keyinchalik fikr yuritamiz. Demak, yuqoridagi kabi dastlabki yaqinlashish sifatida ozod hadlarni olish umuman olganda shart emas. Ularning o'rniga nollar: $x_1^{(0)} = x_2^{(0)} = \dots = x_n^{(0)} = 0$ yoki istalgan sonlarni olish mumkin. Bunday ixtiyoriylik yaqinlashish jarayoniga kuchli ta'sir ko'rsatmaydi. Muayyan aniqlikdagi yechimni olish uchun kerak bo'lgan iteratsiyalar soni esa dastlabki yaqinlashishlarga bog'liq bo'ladi. Agar dastlabki yaqinlashish topilayotgan taqribiy yechimga qanchalik yaqin bo'lsa, yechim shuncha kam iteratsiyalar bilan olinadi. Iteratsiya jarayonining yaqinlashish tezligi koeffisientlar matrisasining xususiyatlariga bog'liq.

Yaqinlashuvchi iteratsiya jarayoni o'z-o'zini tuzatish kabi juda ajoyib xususiyatga ega. Ma'lum iteratsiya qadamida xatoliklarga yo'l qo'yilsa, keyingi iteratsiya qadamlarida ular o'z-o'zidan tuzatiladi. Lekin, shuni ham ta'kidlab o'tish lozimki, yo'l qo'yilgan xatoliklar ma'lum aniqlikdagi yechimlarni olish uchun zarur bo'lgan iteratsiyalar sonini ko'paytirib yuborishi mumkin. Xatoliklarning o'z-o'zidan tuzatilishi iteratsiya jarayonining har bir qadamida dastlabki yaqinlashish sifatida qaralayotgan oldingi iteratsiya qadamida olingan yechimning ishlatilishi tufayli yuz beradi.

Endi iteratsiya usulining yaqinlashish shartlarini keltiramiz. Agar keltirilgan (3.18) sistema uchun quyidagi

$$1) \sum_{j=1}^n |\alpha_{ij}| < 1, \quad i = \overline{1, n}, \quad 2) \sum_{i=1}^n |\alpha_{ij}| < 1, \quad j = \overline{1, n}, \quad 3) \sum_{i=1}^n \sum_{j=1}^n |\alpha_{ij}|^2 < 1 \quad (3.21)$$

shartlardan hech bo'lmaganda biri bajarilsa, (3.19) iteratsiya jarayoni boshlang'ich yaqinlashishni qanday olishdan qat'iy nazar, sistemaning yagona yechimiga yaqinlashadi.

Dastlabki (3.1) sistema esa (3.21) shartlarning birinchi va ikkinchisiga asosan

$$|a_{ii}| > \sum_{\substack{i=1 \\ j \neq i}}^n |a_{ij}|, \quad i = \overline{1, n}, \quad |a_{jj}| > \sum_{\substack{j=1 \\ i \neq j}}^n |a_{ij}|, \quad j = \overline{1, n}, \quad (3.22)$$

shartlarning hech bo'lmaganda bittasi bajarilganda tezroq yaqinlashadi.

Bu shartlar bajarilmasa (3.1) sistemaga bevosita iteratsiya usulini qo'llab bo'lmaydi. Buning uchun, sistemani chiziqli almashtirishlar yordamida (3.22) shartni qanoatlantiruvchi teng kuchli sistemaga keltirish kerak. Ayrim hollarda dastlabki sistemada tenglamalarning o'rnini almashtirishning o'ziyoq aytilgan teng kuchli sistemani beradi.

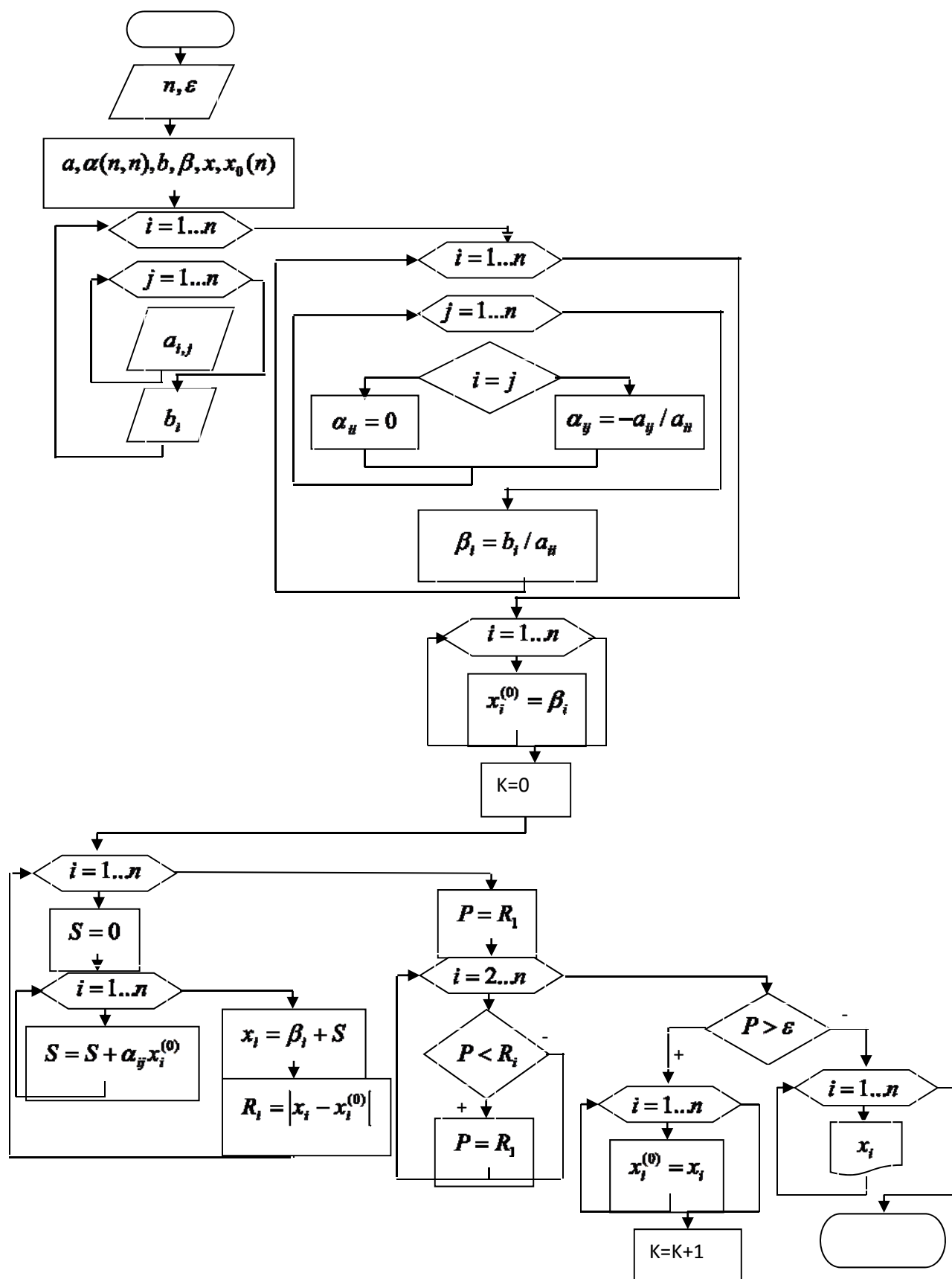
Har bir iteratsiya qadamidan so'ng olingan yechim oldingi iteratsiya qadamida olingan yechim bilan taqqoslanishi kerak. Ketma-ket ikkita iteratsiya qadamidagi yechimlarning o'zaro yaqinligi

$$\delta = \max_{1 \leq i \leq n} |x_i^{(k)} - x_i^{(k-1)}| \quad (3.23)$$

miqdor bilan baholanishi mumkin. Masalan, tenglamalar sistemasini avvaldan qabul qilingan ε aniqlikda yechish talab qilingan bo'lsa, iteratsiyalarni $\delta < \varepsilon$ shart bajarilguncha davom ettirish kerak.

Sistemaning taqribiy yechimi sifatida $\delta < \varepsilon$ shartni bajaruvchi oxirgi iteratsiya qadamida topilgan $\tilde{x} = x_1^{(k)}, \tilde{x}_2 = x_2^{(k)}, \dots, \tilde{x}_n = x_n^{(k)}$ yechim olinadi. Bu yerda $\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n$ lar berilgan sistemaning taqribiy yechimlaridir.

Iteratsiya usuli algoritmining blok-sxemasi



Algoritma dasturi

Program iterasiya;

$N = 4$

$EPS = 0.001$

$k, i, j, p, s = 0, 0, 0, 0, 0$

$r, x_0, x, b, beta = [0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0]$

$a, alfa = [[0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0]], [[0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0]]$

for i in range(N):

 for j in range(N):

$a[i][j] = \text{float}(\text{input}(f"A[\{i\}][\{j\}] = "))$

 if $i == j$:

$alfa[i][j] = 0$

 else:

$a[i][j] = -a[i][j]/a[i][i]$

$b[i] = \text{float}(\text{input}(f"B[\{i\}] = "))$

$beta[i] = b[i]/a[i][i]$

$x_0[i] = beta[i]$

$k = 0$

$br = \text{True}$

while br :

 for i in range(N):

$s = 0$

 for j in range(N):

$s += \text{float}(alfa[i][j]) * \text{float}(x_0[i])$

$x[i] = beta[i] + s$

$r[i] = \text{abs}(x[i] - x_0[i])$

$p = r[1]$

 for i in range(2, N):

 if $p < r[i]$:

$p = r[i]$

 if $p > EPS$:

 for i in range(N):

$x_0[i] = x[i]$

$k += 1$

 else:

$br = \text{False}$

 for i in range(N):

$\text{print}(f"X[\{i\}] = \{x[i]\}")$



Nazorat savollari

1. Iteratsiya usulida dastlab sistemaning ko‘rinishi qanday o‘zgartiriladi?
2. Iteratsiya usuli uchun dastlabki yaqinlashish qanday tanlanadi?
3. Iteratsiya usulining yaqinlashish shartlarini bilasizmi?
4. Iteratsion jarayon qachon to‘xtatiladi?

6-§. Zeydel usuli



Tayanch so‘z va atamalar

Zeydel usuli, keyingi yaqinlashish, avvalgi yaqinlashish, yaqinlashish shartlari, usulning ishchi algoritmi, dastur ta‘minoti.

Bu usulni 1847 yilda taniqli nemis matematigi F. L. Zeydel ishlab chiqqan. Mazkur usul oldingi paragrafda qaralgan iteratsiya usulining takomillashgan ko‘rinishidir. Shuning uchun, u ko‘pincha takomillashtirilgan iteratsiya usuli deb ham yuritiladi.

Zeydel usulining asosiy g‘oyasi shundan iboratki, x_i yechimning $(k+1)$ -yaqinlashishini hisoblash uchun $x_1, x_2, \dots, x_{i-1}$ yechimlar-ning $(k+1)$ -yaqinlashishi, qolgan $x_i, x_{i+1}, \dots, x_n$ yechimlarning esa k -yaqinlashishi ishlatiladi.

Bizga (3.18) sistema berilgan bo‘lsin. Koordinatalar bo‘yicha sistemani

$$x_i = \beta_i + \sum_{j=1}^n \alpha_{ij} x_j \quad (i = \overline{1, n}) \quad (3.24)$$

ko‘rinishda yozamiz. (3.24) sistemada umumiy holda $\alpha_{ii} \neq 0 \quad (i = \overline{1, n})$ bo‘lishi mumkinligi hisobga olingan.

Sistema yechimining k -yaqinlashishi topilgan bo‘lsin. Zeydel usuli bo‘yicha yechimning $(k+1)$ -yaqinlashishi quyidagicha topiladi:

$$x_1^{(k+1)} = \beta_1 + \sum_{j=1}^n \alpha_{1j} x_j^{(k)},$$

$$x_2^{(k+1)} = \beta_2 + \alpha_{21} x_1^{(k+1)} + \sum_{j=2}^n \alpha_{2j} x_j^{(k)},$$

$$x_i^{(k+1)} = \beta_i + \sum_{j=1}^{i-1} \alpha_{ij} x_j^{(k+1)} + \sum_{j=i}^n \alpha_{ij} x_j^{(k)},$$

$$x_n^{(k+1)} = \beta_n + \sum_{j=1}^{n-1} \alpha_{nj} x_j^{(k+1)} + \alpha_{nn} x_n^{(k)}, \quad k = 0, 1, 2, \dots$$

Zeydel usulining yaqinlashish shartlari oddiy iteratsiya usulining yaqinlashish shartlaridan, umuman olganda, farq qiladi. Zeydel usulining yaqinlashishi uchun quyidagi shartlarning hech bo'lmaganda bittasi bajarilishi yetarlidir:

$$1) \sum_{j=1}^n |\alpha_{ij}| < 1, i = \overline{1, n}; \quad 2) \sum_{i=1}^n |\alpha_{ij}| < 1, j = \overline{1, n}; \quad 3) \sum_{i=1}^n \sum_{j=1}^n |\alpha_{ij}|^2 < 1;$$

4) A matrisa simmetrik va musbat aniqlangan.

Yaqinlashish shartlari bajarilganda yechim boshlang'ich yaqinlashishning olinish usuliga bog'liq emas.

Yuqoridagi shartlardan ko'rinadiki, oddiy iteratsiya usuli-ning yaqinlashish shartlari bajarilganda Zeydel usuli yaqinlashadi.

Odatda Zeydel usuli oddiy iteratsiya usuliga nisbatan yechimga tezroq yaqinlashadi. Lekin, ayrim hollarda buning aksi ham bo'lishi mumkin.

Algoritm dasturi

Program Zeydel

$N=4$

$EPS=0.001$

$k, i, j, p, s=0, 0, 0, 0, 0$

$r, x0, x, b, beta =$

$[0]*N, [0]*N, [0]*N, [0]*N, [0]*N$

$a, alfa = [[0]*N]*N, [[0]*N]*N$

for i in range(N):

for j in range(N):

$a[i][j] = \text{float}(\text{input}(\text{"A[" + str(i) + "][" + str(j) + "]} = "))$

$alfa[i][j] = -a[i][j]/a[i][i]$

$b[i] = \text{float}(\text{input}(\text{"B[" + str(i) + "]} = "))$

$beta[i] = b[i]/a[i][i]$

for i in range(N):

$x0[i] = beta[i]$

$k=0$

$br=True$

while br :

$s = 0$

for i in range(2, N):

$s1=0$

for j in range($i-1$):

$s1=s1+alfa[i][j]*x[j]$

$s2=0$

for j in range(i, N):

$s2+=alfa[i][j]*x0[j]$

$x[i] = beta[i] + s1 + s2$

$r[i] = \text{abs}(x[i] - x0[i])$

$p=r[1]$

for i in range(2, N):

if $p < r[i]$:

$p=r[i]$

if $p > EPS$:

for i in range(N):

$x0[i] = x[i]$

else:

$br = False$

if $k > 15$:

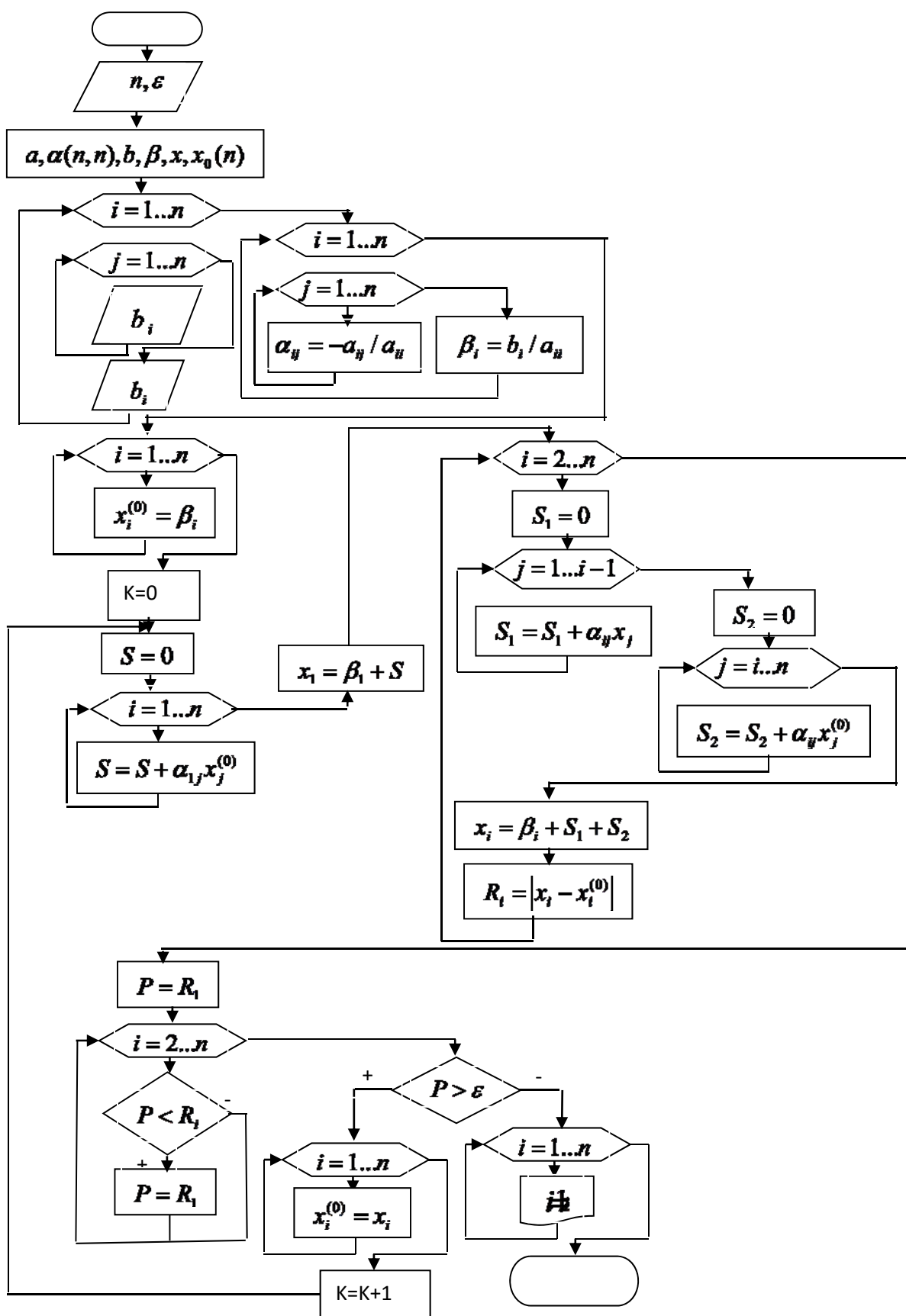
$br = False$

$k+=1$

for i in range(N):

$\text{print}(\text{"x[" + str(i) + "]} = {x[i]})"$

Zeydel usuli algoritmining blok-sxemasi





Nazorat savollari

1. Zeydel usuli qachon ishlab chiqilgan?
2. Zeydel usuli uchun kerakli Yakobi matrisasi qanday quriladi?
3. Zeydel usulining yaqinlashish shartlarini bilasizmi?
4. Zeydel usulida iteratsion jarayon qachon to'xtatiladi?

XULOSA

Mazkur bobda tenglamalar sistemasining yechimi, turli xil matrisalar va ularning tavsifi, CHATSni yechish usullari guruhi haqida umumiy ma'lumotlar keltirilgan.

Gauss usulining mohiyati, to'g'ri va teskari bosqichlar, ularning vazifasi, asosiy ishchi formulalar, usulga mos hisoblash algoritmi, algoritm blok-sxemasi, algoritmgga mos dastur ta'minoti, na'muna sifatida olingan misol va unga mos natijalar tahlili bayon qilingan.

Yuqori tartibli determinantlarni hisoblash usullari tahlil etilib, Gauss usuli yordamida matrisalar determinantini hisoblash algoritmi va dastur ta'minoti keltirilgan. Hisoblashdagi xatolik miqdorini aniqlash uchun na'muna sifatida to'rtinchi tartibli matrisaning determi-nanti ikki xil usulda hisoblanib, olingan natijalar taqqoslangan.

Berilgan matrisaga teskari matrisani Gauss usuli yordamida va Kroneker belgisi asosida hisoblanib, topilgan teskari matrisaning to'g'riligi an'anaviy usulda matrisalarni ko'paytirish orqali aniqlandi.

CHATSni iteratsion usullar yordamida yechish maqsadida iteratsiya va Zeydel usullari qo'llandi. Buning uchun dastlabki yaqinlashishni aniqlash, keyingi yaqinlashishlarni topish uchun ishchi formulalardan foydalanish, yaqinlashish tezligi va yaqinlashish shartlari masalasi har bir usul uchun alohida qaraldi.

Usullarga mos ishchi algoritmlar, dastur ta'minotlari, na'munaviy misollar va natijalar tahlili keltirildi.



Bobga doir muammoli vaziyatlar!

- + Amaldagi qaysi jarayonlarning matematik modellari chiziqli algebraik tenglamalar sistemasi orqali ifodalanadi deb o'ylaysiz? Shunday masalalarga misollar keltira olasizmi?
- + Tenglamalar sistemasini aniq usullar bilan yechganda olinadigan yechim har doim aniq bo'lavermasligiga nima sabab bo'lishi mumkin? Shunday holda aniq usullarni qo'llamaslik kerak degan fikrga qo'shilasizmi? Umuman olganda tenglamalar sistemasini aniq yechish imkoniyati mavjudmi?
- + Sistema yagona yechimi mavjudligining zaruriy va yagona sharti nima uchun determinantlar hisobi bilan bog'liqligini tushuntira olasizmi? Shu shart yechim mavjudligini aniqlash uchun yetarli deb o'ylaysizmi?
- + Har qanday CHATS uchun to'g'ri usullarni qo'llash maqsadga muvofiqmi? Ular har doim maqbul yechimlarni olishga yordam berishi mumkinmi?
- + Gauss usulini hamma CHATSlar uchun qo'llansa maqsadga muvofiq yechimlar olinishiga ishonasizmi? Usulni samarali qo'llash uchun qo'shimcha tavsiyalar qirita olasizmi?
- + Gauss usulining boshqa usullarda deyarli uchramaydigan alohida afzallik tomoni mavjud. Uni tavsiflay olasizmi?
- + Matrisa determinantini hisoblash siz bilgan qaysi masalalarni yechishda qo'llaniladi deb o'ylaysiz. Determinantlar hisoblash usuli to'g'ri usullar guruhiga kiradimi? Nima uchun?
- + Matrisa determinantini hisoblash zaruriyati tug'ilganda aynan qaysi usulni tanlar edingiz? Sababini tushuntira olasizmi?
- + Iteratsion usullarda aniq usullarda uchramaydigan xatoliklar bilan bog'liq "ajoyib xususiyat"ni aniqlay olasizmi? Bu holat iteratsion usullarning CHATSni yechishdagi "ustun"ligiga sabab bo'lishi mumkinmi?
- + Har qanday tenglamalar sistemasi uchun iteratsion usullarni qo'llash mumkin deb o'ylaysizmi? Bunda sistema koeffisientlari uchun alohida shartlarning bajarilishi talab etilmaydimi?

4-BOB. FUNKSIYALARNI INTERPOLYATSIYALASH

Amaliy faoliyatda ko‘pincha jarayon va hodisalarning matematik ifodasini topish bilan bog‘liq muammolarga duch kelinadi. Ayniqsa, iqtisodiyot va texnikada funksiya va jadval ko‘rinishida berilgan funksiya argumentining turli qiymatlariga mos funksiyaning analitik ko‘rinishini hamda matematik ifodasini topish masalasi birmuncha murakkab masala hisoblanadi. Shu bois ushbu bobda tajriba natijalarini qayta ishlashda qo‘llanadigan funksiyaning interpolatsiyalash masalasi qaraladi.

Ko‘phad bilan interpolatsiyalash, xususan Nyuton va Lagranj interpolatsion formulalari misol sifatida tavsiflanadi. Jadvalning qiymatlariga qarab funksiyaning analitik ko‘rinishini topish uchun eng kichik kvadratlar usuli tavsiya qilinadi.

1-§. Interpolatsiya masalasi



Tayanch so‘z va atamalar

Interpolatsiya, jadval funksiya, ekstrapolyatsiya, Vandermond determinanti, ko‘phad.

Interpolatsiyalash tajriba yoki kuzatish natijalarini umumlashtirish, analitik ko‘rinishini topish qiyin bo‘lgan murakkab funksiyalarni soddaroq funksiyalar bilan almashtirish va ularning qiymatlarini hisoblashda ishlatiladi.

Interpolatsiya deganda, erkli o‘zgaruvchi miqdor bilan funksiyaning diskret nuqtalardagi mos qiymatlari orasidagi munosabati ma‘lum bo‘lgan holda funksional bog‘lanishning taqribiy yoki aniq analitik ifodasini tuzish tushuniladi. Boshqacha aytganda, tajriba o‘tkazish yoki kuzatish natijasida quyidagi jadval funksiya

x_i	x_0	x_1	$\dots$	x_n
y_i	y_0	y_1	$\dots$	y_n

olingan bo'lsa, x va y o'zgaruvchilar orasida bog'lanish bormi, bor bo'lsa ular qanday qonuniyat bilan bog'langan degan muammoga javob izlanadi. Yoki, $x_1, x_2, \dots, x_n$ diskret nuqtalarda $y = f(x)$ funksiyaning qiymati ma'lum bo'lsa, boshqa x larda shu funksiyaning qiymatini topish tushuniladi. Umuman olganda, yuqorida aytgan qonuniyatni jadval ko'rinishida berilgan funksiya orqali bir qiymatli aniqlash mumkin emas.

Shuning uchun, bu masalaning klassik yechimi quyidagicha hal qilinadi: $F(x)$ funksiya $x = x_i, i = 0, 1, \dots, n$, nuqtalarda jadval funksiya $f(x)$ bilan ustma-ust tushadi deb qabul qilinadi, ya'ni

$$F(x_i) = f(x_i), \quad i = 0, 1, \dots, n \quad (4.1)$$

Yaqinlashuvchi funksiya $F(x)$ ni (4.1) shart asosida topishni interpolatsiya masalasi deyiladi. $F(x)$ funksiyaning interpolatsiya formulasi, $x_0, \dots, x_n$ nuqtalarni interpolatsiya nuqtalari, (4.1) shartni esa interpolatsiya shartlari deb ataladi.

$F(x)$ funksiyaning darajali ko'phad, trigonometrik ko'phad, rasional funksiya, splayn-funksiya ko'rinishida tanlab olish mumkin. Izlanayotgan ko'phad uchun quyidagi teorema doimo o'rinli bo'ladi.

Teorema: Agar $a = x_0 < x_1 < \dots < x_n = b$ bo'lsa, $P_n(x_i) = y(x_i), i = 0, 1, \dots, n$ shartlarni qanoatlantiruvchi yagona interpolatsiya ko'phadi $P_n(x)$ mavjud.

Argument va funksiya o'rtasidagi taqribiy munosabat biror bir usul bilan o'rnatilgach, bu munosabat orqali x o'zgaruvchining jadvalda berilmagan qiymatlarida $y = f(x)$ funksiya qiymatlarini hisoblash mumkin.

Agar funksiyaning taqribiy qiymati aniqlanayotgan x argument berilgan $[x_0, x_n]$ oraliqqa tegishli bo'lsa, u holda qo'yilgan masala interpolatsiyalash masalasi, agar hisoblanayotgan qiymat x argumentning $[x_0, x_n]$ oraliqdan tashqaridagi qiymatlari uchun bo'lsa, qo'yilgan masala ekstrapolyatsiyalash masalasi deyiladi.

$F(x)$ funksiyaning darajali funksiya ko'rinishida tanlansa, bunday interpolatsiya parabolik deb ataladi. Bunday yaqinlashish usuli, shunga

asoslanadiki, $f(x)$ funksiya uncha katta bo'lmagan oraliqlarda muayyan tartibli parabola uchun juda qulay approksimasiyalanadi.

Agar $f(x)$ funksiya davriy bo'lsa, u holda $F(x)$ ni ba'zi hollarda trigonometrik, ba'zan esa rasional funksiyalar sifatida tanlanadi. Shunday qilib, interpoliyatsiyada quyidagi vazifalarni hal etish lozim bo'ladi.

Har qanday aniq holat uchun interpoliyatsiya funksiyasini qurishning qulay usulini tanlash.

$[a, b]$ kesmada interpoliyatsiyalanadigan $F(x)$ funksiyaning xatoligini baholash, ya'ni $F(x)$ va $f(x)$ funksiya qiymatlarini $x_0, x_1, \dots, x_n$ tugun nuqtalarda qanchalik mos kelishini aniqlash.

Eng kichik xatolik bilan sodir bo'luvchi interpoliyatsiya tugunlarini maqbul tanlash.

Ko'phad bilan interpoliyatsiyalash. Interpoliyatsiyalash funksiyasini quyidagi ko'phad

$$F(x) = P_n(x) = a_0 + a_1x + \dots + a_nx^n = \sum_{j=0}^n a_jx^j \quad (4.2)$$

ko'rinishida izlaylik. $P_n(x_i) = y_i, i = 0, 1, \dots, n$ deb talab qilib, mazkur ko'phadning noma'lum koeffitsientlariga nisbatan quyidagi chiziqli algebraik tenglamalar sistemasini hosil qilamiz:

$$\sum_{j=0}^n a_jx_i^j = f(x_i), i = 0, 1, \dots, n \quad (4.3)$$

(4.3) sistemani to'g'ri yoki iteratsion usullar yordamida yechib, qiymatlari $x = x_i$ nuqtalarda $y = f(x)$ funksiya qiymatlari bilan ustma-ust tushadigan $y = P_n(x)$ ko'phadi hosil qilinadi. Lekin, (4.3) sistemani yechishda o'ziga xos muammolar hosil bo'ladi. Bu muammo ayniqsa, sistemaning tartibi katta bo'lganda o'zini namoyon qiladi. Chunki, yuqori tartibli sistemalarni yechishda kompyuter xotirasini yetishmasligi yoki hisoblashlardagi yaxlitlash xatosining ortib ketishi kabi muammolar hosil bo'ladi. Shunday xollarning oldini olish uchun $P_n(x)$ ko'phadi maxsus ko'rinishlarda tanlab olinadi.

$P_n(x)$ interpolatsiya ko'phadining I.Nyuton (1643-1727) va J.L.Lagranj (1736-1813) topgan qulay ko'rinishlari ana shu maxsus ko'rinishlarga misol sifatida qaralishi mumkin.



Nazorat savollari

1. Interpolyatsiyalash jarayoni qanday amalga oshiriladi?
2. Interpolyatsiya qanday funksiyalar uchun qo'llanadi?
3. Ekstrapolyasiya nima?
4. Interpolyatsiyada noma'lum ko'phadning koeffisientlari qanday aniqlanadi?

2-§. Nyuton interpolasyon formulasi



Tayanch so'z va atamalar

Tugun nuqtalar, n-darajali ko'phad, Nyuton interpolasyon formulalari, interpolatsiya xatoligi, interpolatsiyalash algoritmi, dastur matni.

Amalda Nyutonning ikkita interpolatsiyalash formulasi mavjud. Nyutonning birinchi interpolatsiya formulasida $y = f(x)$ funksiyaning qiymatini x_0 boshlang'ich qiymati atrofida hisoblash qulay bo'lib, funksiyaning jadvalning oxiridagi qiymatlari uchun hisoblash noqulay. Nyutonning ikkinchi interpolatsiya formulasi kesma oxirida x_n nuqtadan orqaga qarab bajariladi.

Aytaylik, $f(x)$ funksiya $x_0, x_1, \dots, x_n$ tugun nuqtalardagi $y_0 = f(x_0), y_1 = f(x_1), \dots, y_n = f(x_n)$ qiymatlar bilan berilgan bo'lsin, $P_n(x_0) = y_0$, $P_n(x_1) = y_1, \dots, P_n(x_n) = y_n$ tengliklarni qanoatlantiruvchi $P_n(x)$ ko'phadni qurish talab qilinmoqda. n -darajali interpolatsiya ko'phadining mavjudlik va yagonalik xossasidan kelib chiqib, uni quyidagi tenglama ko'rinishida ifodalanadi:

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \dots + a_n(x - x_0)(x - x_1)\dots(x - x_{n-1})$$

Bu yerda $a_0, a_1, \dots, a_n$ lar hozircha noma'lum koeffisientlardir. Ularni $P_n(x_i) = y_i, i = 0, \dots, n$, interpolatsiya shartlaridan topamiz. Interpolatsiya shartlari bizga quyi uchburchak matrisali chiziqli tenglamalar sistemasini beradi. Uni

yoʻzishdan avval biz funksiyaning $x_0, \dots, x_k, 1 \leq k \leq n$ nuqtalarda argumentlarga nisbatan simmetrik k -tartibli boʻlingan ayirmalar (k-t.b.a.) tushunchasini kiritamiz:

$$1\text{-t. b.a.: } f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = f[x_1, x_0],$$

$$2\text{-t. b.a.: } f[x_0, x_1, x_2] = \frac{f[x_0, x_2] - f[x_0, x_1]}{x_2 - x_1} = \frac{f[x_0, x_1] - f[x_0, x_2]}{x_1 - x_2},$$

$$k\text{-t. b.a.: } f[x_0, \dots, x_k] = \frac{f[x_0, \dots, x_{k-2}, x_k] - f[x_0, \dots, x_{k-1}]}{x_k - x_{k-1}} = \frac{f[x_1, \dots, x_{k-1}] - f[x_0, \dots, x_{k-1}]}{x_k - x_0}, k \leq n.$$

Endi $P_n(x_i) = y_i, i = 0, \dots, n$, interpolatsiya shartlarni bajarilishini soʻrab quyidagi munosabatlarni hosil qilamiz:

$$P_n(x_0) = y_0 \rightarrow a_0 = y_0 = f(x_0), \quad P_n(x_1) = y_1 \rightarrow f(x_0) + a_1(x_1 - x_0) = f(x_1) \rightarrow a_1 = f[x_0, x_1]$$

$$P_n(x_2) = y_2 \rightarrow f(x_0) + f[x_0, x_1](x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) = f(x_2) \rightarrow a_2 = f[x_0, x_1, x_2],$$

$$\dots \quad P_n(x_k) = y_k \rightarrow a_k = f[x_0, x_1, \dots, x_k], 1 \leq k \leq n.$$

Topilgan koeffisientlarni $P_n(x)$ formulaga qoʻyib Nyutonning 1-interpolatsiya formulasi (koʻphadi) ni topamiz:

$$P_n(x) = f(x_0) + \sum_{k=1}^n f[x_0, \dots, x_k](x - x_0) \dots (x - x_{k-1}). \quad (4.4)$$

Qoldiq had quyidagicha boʻladi:

$$R_n(f, x) = f(x) - P_n(x) = f[x, x_0, \dots, x_n](x - x_0) \dots (x - x_n)$$

Xuddi shunday interpolatsiya koʻphadini ushbu koʻrinishda

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \dots + a_n(x - x_0)(x - x_1) \dots (x - x_{n-1})$$

izlab, Nyutonning 2-interpolatsiya formulasini topamiz:

$$P_n(x) = f(x_n) + \sum_{k=1}^n f[x_{n-k}, \dots, x_n](x - x_{n-k}) \dots (x - x_{n-k+1}) \quad (4.5)$$

(4.4) va (4.5) formulalar mos holda jadval boshida va oxirida kerak boʻladi.

Ularni interpolatsiya nuqtalari teng uzoqlikda joylashgan hol uchun chiqaramiz:

$x_k - x_{k-1} = h, k = 1, 2, \dots, n$. Avvalo, k -tartibli chekli ayirmalar (k-t.ch.a) tushunchasini kiritamiz:

$$1\text{-t. ch.a.: } \Delta y_0 = \Delta f(x_0) = y_1 - y_0 = f(x_1) - f(x_0),$$

2-t. ch.a.: $\Delta^2 y_0 = \Delta^2 f(x_0) = \Delta y_1 - \Delta y_0 = \Delta f(x_1) - \Delta f(x_0) = y_2 - 2y_1 + y_0, \dots,$

k-t. ch.a.: $\Delta^k y_0 = \Delta^k f(x_0) = \Delta(\Delta^{k-1} y_0) = y_k - C_k^1 y_{k-1} + C_k^2 y_{k-2} - \dots - (-1)^{k-1} y_0.$

Bu yerda $C_k^i = \frac{k!}{i!(k-i)!}$ k elementdan i tadan kombinasiyalar soni.

Endi bo‘lingan ayirmalar va chekli ayirmalar orasida bog‘lanish o‘rnatamiz. Ravshanki,

$$f[x_0, x_1] = \frac{y_1 - y_0}{x_1 - x_0} = \frac{\Delta y_0}{h}, \quad f[x_0, x_1, x_2] = \frac{1}{h} \left[\frac{y_2 - y_0}{x_2 - x_0} - \frac{y_1 - y_0}{x_1 - x_0} \right] = \frac{\Delta^2 y_0}{2!h^2}, \dots,$$

$$f[x_0, \dots, x_k] = \frac{1}{kh} [f[x_1, \dots, x_{k-1}] - f[x_0, \dots, x_{k-1}]] = \frac{1}{kh} \left[\frac{\Delta^{k-1} y_1 - \Delta^{k-1} y_0}{(k-1)!h^{k-1}} \right] = \frac{\Delta^k y_0}{k!h^k}.$$

Bu formulalarni (4.4) ga qo‘yib quyidagi formulani olamiz:

$$P_n(x) = y_0 + \frac{\Delta y_0}{1!h}(x - x_0) + \frac{\Delta^2 y_0}{2!h^2}(x - x_0)(x - x_1) + \dots + \frac{\Delta^n y_0}{n!h^n}(x - x_0) \dots (x - x_{n-1}) \quad (4.5)$$

(4.5) formula Nyutonning birinchi interpoliyasion formulasi hisoblanadi. Nyutonning birinchi interpoliyasion formulasi xatoligi quyidagi miqdor bilan baholanadi:

$$R_n(x) = h^{n+1} \frac{f^{(n+1)}(\xi)}{(n+1)!} q(q-1)(q-2) \dots (q-n), \quad x_0 < \xi < x_n, \quad q = \frac{x - x_0}{h}.$$

Nyutonning ikkinchi interpoliyasion formulasi esa jadvalning oxirgi qiymatlari uchun hosil qilinadi. Buning uchun quyidagi interpoliyasion ko‘phadni (4.6) ko‘rinishda olib, koeffisientlar uchun quyidagi tengliklarni hosil qilamiz:

$$a_i = \frac{\Delta^i y_{n-i}}{i!h^i}, \quad i = 0, \dots, n.$$

Topilgan qiymatlarni va $q = \frac{x - x_n}{h}$ belgilashlarni hisobga olib, $P_n(x)$

ko‘phadni quyidagicha ifodalanadi:

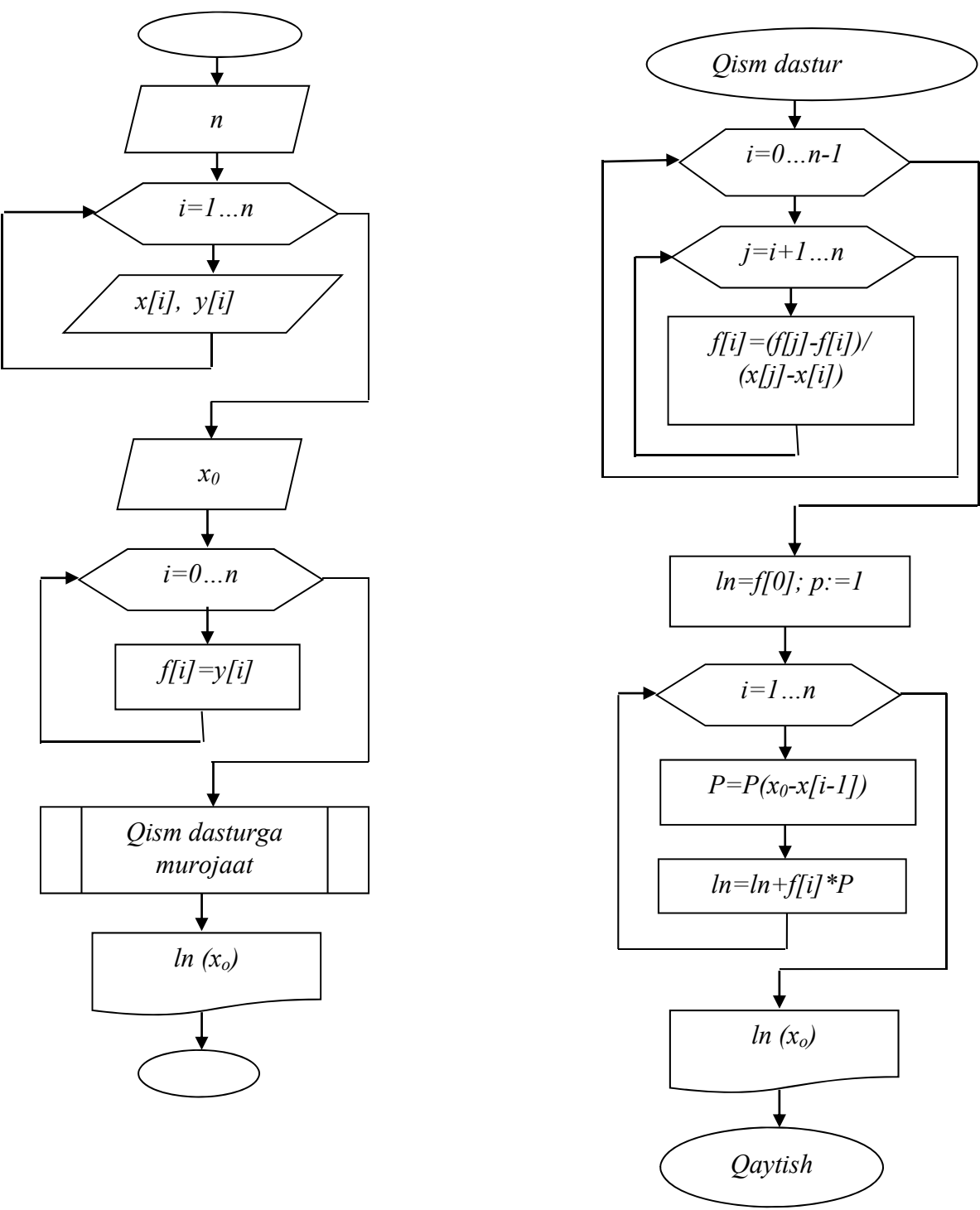
$$P_n(x) = y_n + q\Delta_{n-1} + \frac{q(q+1)}{2!}\Delta^2 y_{n-2} + \frac{q(q+1)(q+2)}{3!}\Delta^3 y_{n-3} + \dots + \frac{q(q+1) \dots (q+n-1)}{n!}\Delta^n y_0$$

Mazkur formula Nyutonning ikkinchi interpoliyasion formulasi deb yuritiladi.

Nyutonning ikkinchi interpoliyasion formulasi xatoligi quyidagi miqdor bilan baholanadi:

$$R_n(x) = h^{n+1} \frac{f^{(n+1)}(\xi)}{(n+1)!} q(q+1) \dots (q+n), x_0 < \xi < x_n, q = \frac{x - x_n}{h}$$

Quyida Nyuton interpolyasiyoni koʻphadini qurish algoritmining blok- sxemasi va dastur taʼminoti tavsiya qilinadi.



```

Nyuton interpoliyasion ko 'phadini qurish dasturi
# program Newtoninterpolation(i,o);
vec = [0]*20
i,j,n,p,ln,c,x0=0,0,0,0,0,0,0
x,y,f=vec, vec, vec
d=""
def tab(n, x,y):
    i=0
    for i in range(n):
        x[i],y[i] = map(float, input(f"x,y [{i}] = ").split())
    return x,y
def newton(m, x,f, x0,ln):
    p,c = 0,0
    for i in range(n-1):
        for j in range(i+1, n):
            f[i]=(f[j]-f[i])/x(x[j]-x[i])
        ln = f[0]
        p=1
        for i in range(n):
            p = p*(x0-x[i-1])
            ln=ln+f[i]*p
        return ln
n = int(input("n = "))
x,y = tab(n,x,y)
br = True
while br:
    x0 = float(input("x0= "))
    for i in range(n):
        f[i]=y[i]
    ln = newton(n,x,f,x0,ln)
    print(f"LN[{x0}] = {ln}")
    br = False

```

Olingan natijalar tahlili.

Quyidagi jadval ko'rinishidagi funksiya berilgan bo'lsin. nq5.

i	1	2	3	4	5
x_i	1,5	2	2,5	3	4
y_i	4	5,1	4,9	6	7,3

Ishlab chiqilgan hisoblash algoritmini va dastur ta'minotining to'g'ri ishlayotganligini aniqlash maqsadida qurilgan Nyuton interpolyasiyon ko'phadini bir nechta yangi qiymatlarda hisoblandi.

$$x_0 = 2.2 \qquad y = 5,0663296$$

$$x_0 = 3.1 \qquad y = 6,5542528$$

$$x_0 = 2.9 \qquad y = 5,557872$$

Sonli natijalar Nyuton interpolyasiyon ko'phadini qurish uchun ishlab chiqilgan dastur ta'minoti va hisoblash algoritmlarini ishonchli ekanligini ko'rsatadi.



Nazorat savollari

1. Nyuton interpolyasiyon ko'phadini qurish uchun qaysi formuladan foydalaniladi?
2. Nyutonning birinchi va ikkinchi interpolyasiyon formulalari bir-biridan qanday farqlanadi?
3. Nyuton interpolyasiyon ko'phadini qurish usulining xatoligi qanday baholanadi?

3-§ Lagranj interpolyasiyon formulasi



Tayanch so'z va atamalar

Tugun nuqtalar, n-darajali ko'phad, Lagranj interpolyasiyon formulasi, interpolyasiya xatoligi, interpolyasiyalash algoritmi, dastur matni.

Ma'lumki, Nyutonning interpolyasiyon formulalari x o'zgaruvchining teng uzoqlikda yotgan nuqtalari uchun yaroqli edi. Amaliyotda esa funksiyaning qiymatlarini o'zgaruvchining teng oraliqdagi qiymatlari uchun hisoblash murakkab kechishi mumkin. Bunday hollarda, ya'ni har xil uzoqlikda yotgan nuqtalar uchun

Lagranj interpolyasiyon formulasidan foydalanish samaraliroq kechadi.

Bu usulda $[a, b]$ oraliqdagi ixtiyoriy masofada yotgan $x_0, x_1, \dots, x_n$ tugun nuqtalar va ularga mos qiymatlar uchun shunday $L(x)$ ko'phadni qurish talab etiladiki, u

$$L(x_0) = y_0, \quad L(x_1) = y_1, \dots, \quad L(x_n) = y_n \quad (4,6)$$

ko‘rinishda tasvirlanadi. (4.8)ni boshqacharoq ko‘rinishda ifodalash mumkin:
 $L_n(x) = y_0 Q_0(x) + y_1 Q_1(x) + \dots + y_n Q_n(x)$. Bundan $Q_i(x)$ funksiyani quyidagi shartni qanoatlantirishi talab etiladi:

$$Q_i(x_j) = \begin{cases} 0, & \text{agar } i \neq j \\ 1, & \text{agar } i = j \end{cases}$$

Mazkur shartni qancha $Q_i(x)$ faqat quyidagi nisbatda bo‘lishini anglash qiyin emas.

$$Q_i(x) = \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)}.$$

$x_0, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n$ nuqtalarda $Q_i(x)$ funksiya 0 ga, x_i da esa 1 ga aylanadi.

Nihoyat, (4.8) tenglamani quyidagicha ifodalash mumkin bo‘ladi.

$$\begin{aligned} L_n(x) &= \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x_0-x_1)(x_0-x_2)\dots(x_0-x_n)} y_0 + \frac{(x-x_0)(x-x_2)\dots(x-x_{n-1})}{(x_1-x_0)(x_1-x_2)\dots(x_1-x_n)} y_1 + \dots \\ &+ \frac{(x-x_0)(x-x_1)\dots(x-x_{n-1})}{(x_n-x_0)(x_n-x_1)\dots(x_n-x_{n-1})} y_n = \sum_{i=0}^n \left(\prod_{\substack{j=0 \\ i \neq j}}^n (x-x_j) \cdot y_i / \prod_{\substack{j=0 \\ i \neq j}}^n (x_i-x_j) \right) \end{aligned} \quad (4.9)$$

Mazkur ko‘phad Lagranj interpoliyasion ko‘phadi deb ataladi.

Har xil uzoqlikda yotgan nuqtalar uchun yozilgan ushbu Lagranj interpoliyasion formulasini teng uzoqlikda yotgan nuqtalar uchun ham qo‘llaniladi. Yuqoridagi interpoliyatsiya formulasida x va y o‘zgaruvchilarining o‘rinlarini o‘zaro almashtirish ham mumkin.

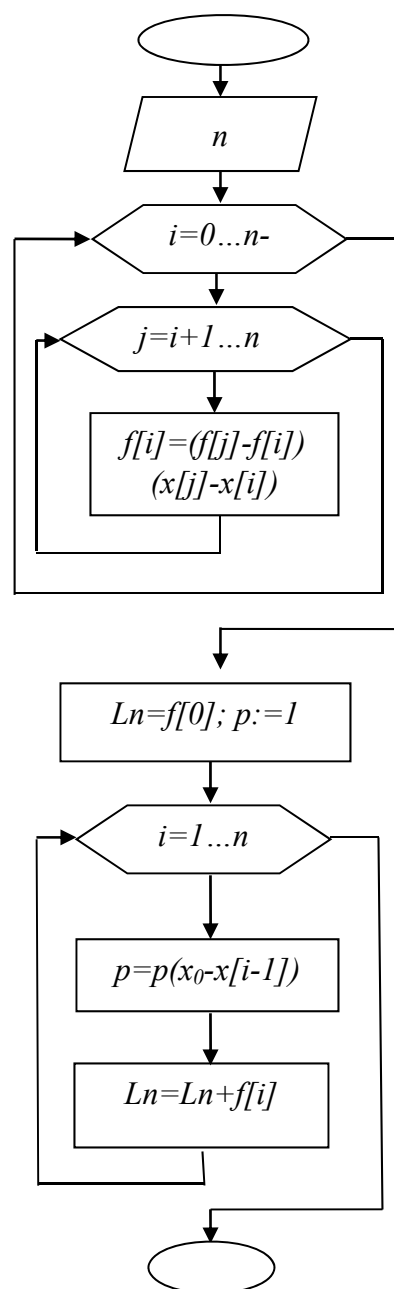
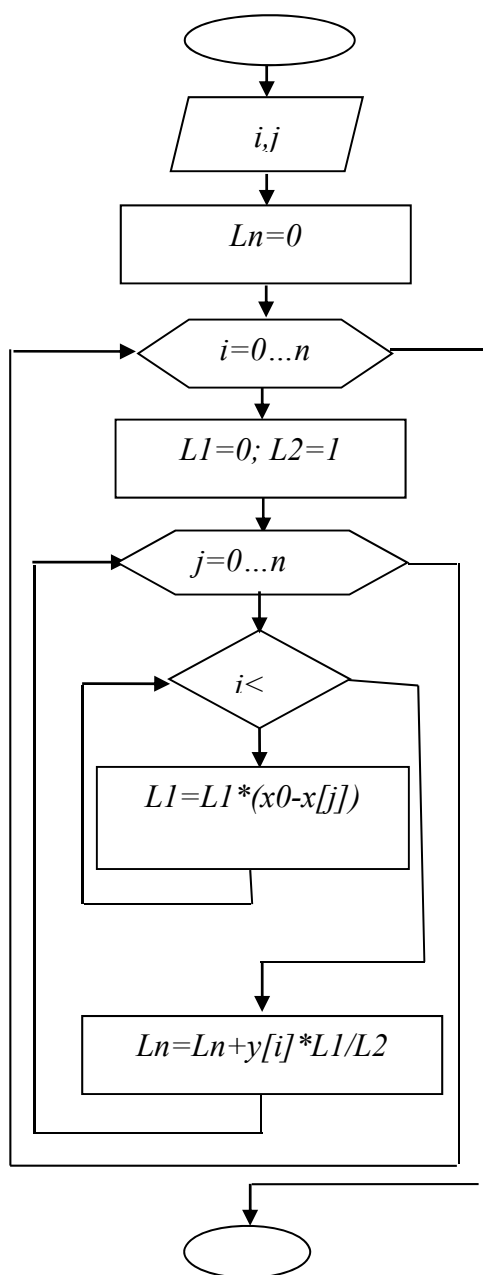
Lagranj interpoliyasion formulasi xatoligini quyidagi formula bilan baholanadi:

$$R_n(x) = \frac{\Delta^{n+1} y_0}{(n+1)!} q(q-1)(q-2)\dots(q-n)$$

bu yerda $q = \frac{x-x_0}{h}$, $\Delta y_0 = y_1 - y_0$. Quyida Lagranj interpoliyasion ko‘phadini

hisoblash algoritmining blok-sxemasi va dastur ta‘minotini tavsiya qilinadi.

Algoritm blok-sxemasi



Lagranj usulining dasturi

```

#program lagrangeinterpolation(i,o);
vec = [1]*20
i,j,n=1,1,1
x,y,f = vec*1,vec*1,vec*1
p,ln,e,x0=1,1,1,1
def tab(n, x,y):
    for i in range(n):
        x[i],y[i] = map(float, input(f"x,y [{i}] = ").split())
    return x,y
def lagrang(n, x,f,x0,ln):
    l1,l2=0,0
    
```

```

i,j=0,0
ln=0
for i in range(n):
    l1=1
    l2=1
    for j in range(n):
        if i!=j:
            l1=l1*(x0-x[j])
            l2=l2*(x[i]-x[j])
    try:
        ln=ln+y[i]*l1/l2
    except:
        ln+=0
return ln

```

```

n=int(input("n= "))
br=True
while br:
    x0=int(input("x0= "))
    for i in range(n):
        f[i]=y[i]
    ln = lagrang(n,x,f,x0,ln)
    print(f"Ln({x0})= {ln}")
    br = False

```

Olingan natijalar tahlili.

Lagranj usulidan olingan natijalarning to‘g‘riligini tekshirish uchun quyidagi jadval ko‘rinishidagi funksiyadan foydalaniladi. $n=6$.

i	1	2	3	4	5	6
x_i	1	2	3	4	5	6
y_i	5	6	6,5	6,8	8	8,8

Ishlab chiqilgan hisoblash algoritmini va dastur ta‘minotining to‘g‘ri ekanligini aniqlash maqsadida qurilgan Lagranj interpoliyasion ko‘phadini bir nechta yangi qiymatlarda hisoblandi.

$$\begin{array}{ll}
 x_0 = 2.5 & y = 6,35 \\
 x_0 = 4.7 & y = 7,543248 \\
 x_0 = 5.1 & y = 8,1574264
 \end{array}$$

Sonli natijalar Lagranj interpoliyasion ko‘phadini qurish uchun ishlab chiqilgan dastur ta‘minoti va hisoblash algoritmlarini ishonchli ekanligini ko‘rsatadi..



Nazorat savollari

1. Lagranj usuli qaysi hollarda qulay hisoblanadi?
2. Lagranj interpoliyasion ko‘phadini qurish uchun qaysi formuladan foydalaniladi?
3. Lagranj usulining xatoligi qanday baholanadi?

4-§. Eng kichik kvadratlar usuli



Tayanch so‘z va atamalar

n-darajali ko‘phad, Lagranj interpoliyasion formulasi, interpoliyatsiya xatoligi, interpoliyatsiyalash algoritmi, dastur ta‘minoti.

Eng kichik kvadratlar usuli birinchi marta 1874 yilda Gauss tomonidan ishlab chiqilgan bo‘lib, ayrim adabiyotlarda bu usul Gauss usuli deb ataladi.

Aytaylik, x erkli o‘zgaruvchining n ta qiymati berilgan bo‘lsin.

Bu usulda qidirilayotgan funksiyaning qiymatlari uni jadvaldagi mos qiymatlariga teng bo‘lmasada, ularning farqlar kvadratlarining yig‘indisi eng kichik bo‘lishi talab qilinadi, ya‘ni

$$\sum_{i=0}^n (y(x_i) - y_i)^2 \rightarrow \min$$

X	X_1	X_2	...	X_n
Y	Y_1	Y_1	...	Y_n

Jadvalda berilganlarga asosanib $P_m(x) = c_0 + c_1x + \dots + c_mx^m$ ko‘phadni qurish kerak, ya‘ni bu ko‘phadning $c_0, c_1, \dots, c_m$ koeffisientlarini aniqlash kerak.

Yaqinlik sharti $\sum_{i=0}^n (y_i - P_m(x_i))^2 = F(c_0, c_1, \dots, c_m)$ ko‘phadning noma‘lum

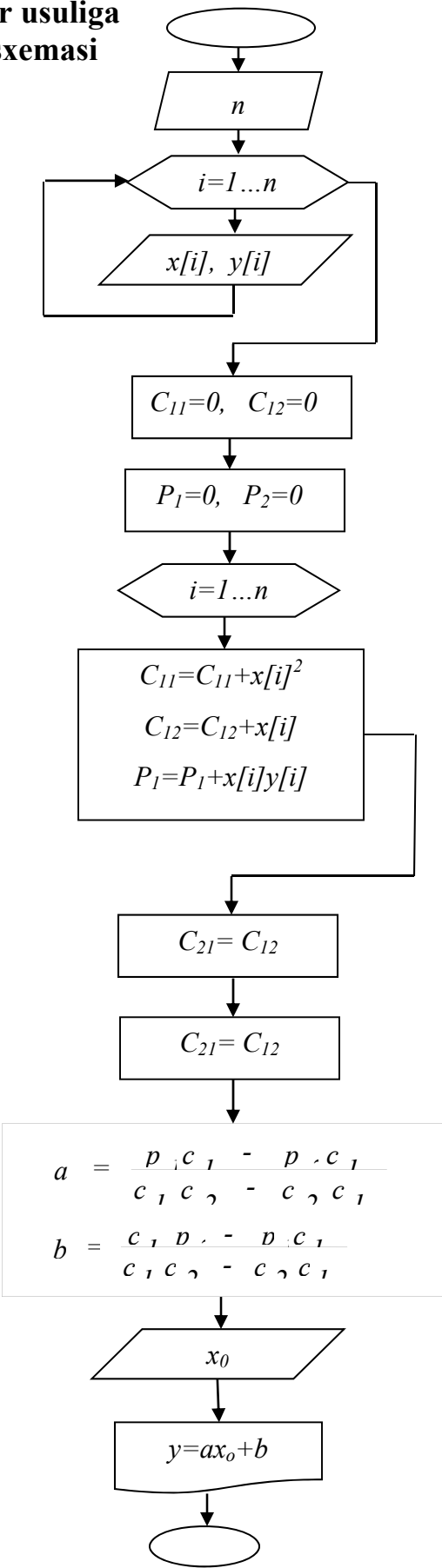
koeffisientlariga nisbatan funksiyadir. Ko‘p o‘zgaruvchili funksiya $F(c_0, c_1, \dots, c_m)$ ni

minimumga erishtirish uchun undan o'zgarmaslar c_i bo'yicha xususiy hosilalar olib nolga tenglaymiz $\partial F / \partial C_i = 0 \quad (i = \overline{0, m})$. Natijada $c_0, c_1, \dots, c_m$ larga nisbatan yozilgan quyidagi chiziqli algebraik tenglamalar sistemasini hosil qilamiz.

$$\left\{ \begin{aligned} (n+1)c_0 + c_i \sum_{i=0}^n x_i + \dots + c_m \sum_{i=0}^n x_i^m &= \sum_{i=0}^n y_i, \\ c_0 \sum_{i=0}^n x_i + c_1 \sum_{i=0}^n x_i^2 + \dots + c_m \sum_{i=0}^n x_i^{m+1} &= \sum_{i=0}^n y_i x_i, \\ \dots & \\ c_0 \sum_{i=0}^n x_i^m + c_1 \sum_{i=0}^n x_i^{m+1} + \dots + c_m \sum_{i=0}^n x_i^{2m} &= \sum_{i=0}^n y_i x_i^m, \end{aligned} \right. \quad (4.10)$$

Sistemani Gauss-Jordan usuli bilan yechib izlanayotgan $P_m(x)$ ko'phadning noma'lum koeffisientlarini hosil qilamiz.

**Eng kichik kvadratlar usuliga
mos algoritm blok-sxemasi**



Algoritmning dastur matni

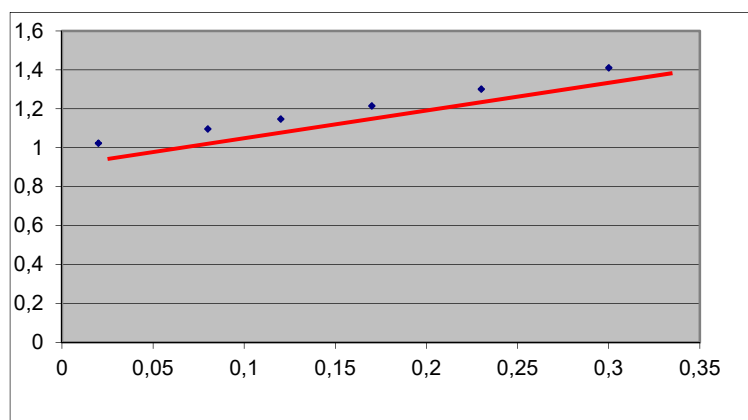
```
# Program Kvadusul;
n=6
x0,y0,a,b,c11,c12,c21,c22,p1,p2=1,1,1,1,1,1,1,1,1,1
x,y=[1]*6,[1]*6
x0 = float(input("Qaysi qiymat uchun hisoblaymiz: "))
print("Massiv elementlarini kiriting: ")
for i in range(n):
    x[i],y[i]=map(float, input("x, y = ").split())
c11=0
x12=0
p1=0
p2=0
for i in range(n):
    c11=c11+x[i]*x[i]
    x12=x12+x[i]
    p1=p1+x[i]*y[i]
    p2=p2+y[i]
a=(p1*c11-p2*c12)/(c11*c22-c21*c12)
b=(c11*p2-p1*c12)/(c11*c22-c21*c12)
print(f"a={a} , b={b}")
y0 = a*x0+b
print("y0= ", y0)
```

Olingan natijalar tahlili.

Yuqoridagi algoritmlarning to'g'riligini tekshirish uchun quyidagi nuqtalarga mos qiymatlarni olaylik.

X	0,02	0,08	0,12	0,17	0,23	0,30
Y	1,02316	1,09590	1,14725	1,21483	1,30120	1,40976

Dasturni ishlatib ko'rish natijasida quyida grafigi tasvirlangan chiziqli funktsiyani hosil qilindi. Bunda $a=1,431246$ $b=0,989741$ ga teng bo'ldi. Natijalardan ko'rinib turibdiki, hosil qilingan funktsiya grafigi berilgan jadval funktsiya qiymatlariga ancha yaqindir.



Quyidagi oraliq qiymatlar uchun natijalar alohida hosil qilindi.

$$x_0 = 0.15 \quad y = 1,20444$$

$$x_0 = 0.4 \quad y = 1,51927$$

Argument va funksiyaning qiymatlarini tahlil etish natijasida ularni tegishli oraliqlarga va yangi funksiyaning xususiyatiga mos ekanligini aniqlash mumkin. Bu esa ishlab chiqilgan algoritmlardan amaliy masalalar yechishda foydalanish mumkinligini ko'rsatadi.



Nazorat savollari

1. Eng kichik kvadratlar usulida nima uchun tafovut miqdorlarning kvadratlari qaraladi?
2. Eng kichik kvadratlar usulining geometrik ma'nosini tushuntirib bering.
3. Eng kichik kvadratlar usulida noma'lumlarni topish uchun tenglamalar sistemasini yechishning qaysi usulini qo'llash afzalroq?

XULOSA

Interpolyatsiyalash masalasining asosiy maqsadi va vazifasi, ko'phad bilan interpolyatsiyalash jarayoni, funksiya va argument o'rtasidagi o'zaro munosabatlar, izlanayotgan ko'phadni aniqlash yo'llari va imkoniyatlari bayon qilindi.

Ko'phad bilan interpolyatsiyalash usullaridan Nyuton va Lagranj interpolyasion ko'phadini qurish jarayoni, usullarning mohiyati, ishchi formulalari va hisoblash algoritmlari tavsiya qilindi.

Nyuton va Lagranj interpoliyasion ko'phadini qurish usullarining xatoligini baholash formulalari ifodalandi.

Nyuton va Lagranj interpoliyasion ko'phadini qurish usuliga mos dastur ta'minotlari yaratilib, natijalar olindi va tahlil etildi, ishlab chiqilgan algoritmlarning ishonchliligi tasdiqlandi.

Funksiyani qurishga oid interpoliyasion usullardan eng kichik kvadratlar usuli va uning mohiyati, usulning asosiy ishchi algoritmi va dastur ta'minoti, natijalar tahlili keltirildi.



Bobga doir muammoli vaziyatlar!

- ✚ Bir nechta argumentlarga mos funksiyaning qiymatlari berilgan holda funksional qonuniyatni bir qiymatli aniqlash mumkinmi?!
- ✚ Parabolik interpoliyatsiyalashning qulaylik jihatlari mavjud deb o'ylaysizmi? Izlayotgan funksiya davriy bo'lganda ham parabolik interpoliyatsiyalashni tavsiya etasizmi? Aksincha bo'lsa nima uchun?
- ✚ Interpoliyatsiyalash va ekstrapolyasiyalash jarayonida ko'phadniqurishning umumiy jihatlari bo'lishi mumkinmi?
- ✚ Nima uchun Lagranj interpoliyasion formulasi har xil uzoqlikda yotgan nuqtalar uchun qo'llanishga qulay deb hisoblanadi? Bu bilan Nyuton usulini turli uzoqlikdagi nuqtalar uchun qo'llash mumkin emas, degan mulohazalar kelib chiqishiga qo'shilasizmi? Fikringizni tushuntiring?
- ✚ Eng kichik kvadratlar usulida aynan tavofut miqdorlar kvadratlarining qaralishiga sabab nima deb o'ylaysiz? Bu usulning xatoligiga ta'sir qilishi mumkinmi? Xatolikni qamaytirish omillari nimalarga bog'liq?

5-BOB. INTEGRALLARNI TAQIRIBIY HISOBLASH

Ma'lumki, berilgan funksiyaning hosilasini topish amali differensiallash deb atalib, uning uchun boshlang'ich funksiyaning topishdan iborat teskari amal integrallash deb ataladi (lotincha-*integrare*-tiklash degan ma'noni bildiradi).

Amalda ko'pgina funksiylarning boshlang'ich funksiyalarini elementar funksiylarning kombinatsiyasi orqali ifodalab bo'lmaydi. Shuning uchun, bu funksiylarning aniq integrallarini ba'zan taqribiy usullar bilan hisoblash zaruriyati tug'iladi. Shuning uchun ushbu bobda aniq integralni taqribiy hisoblashning ayrim usullari, ularning mohiyati, geometrik ma'nosi, ishchi algoritmlari va dastur ta'minotlari tavsiya qilinadi.

1-§. Aniq integrallarni taqribiy hisoblash haqida umumiy tushunchalar



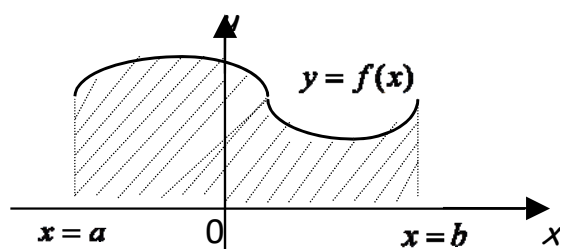
Tayanch so'z va atamalar

Integrallash, egri chiziqli trapetsiya, Nyuton- Leybnis formulasi, integral yig'indi, boshlang'ich funksiya, integrallash formulasi, integrallash usullari.

Aniq integrallarni taqribiy hisoblash egri chiziqli trapetsiyaning yuzi haqidagi masalaning geometrik yechimi bilan uzviy bog'liqdir. Quyidan OX o'qidagi $[a, b]$ kesma bilan, yuqoridan musbat qiymat qabul qiladigan $y = f(x)$ uzluksiz funksiyaning grafigi bilan, yon tomonlardan $x = a$ va $x = b$ to'g'ri chiziqlarning kesmalari bilan chegaralangan figurani egri chiziqli trapetsiya deyiladi. Egri chiziqli trapetsiyaning yuzini (14-rasm)

$$S = \int_a^b f(x)dx = F(x)\Big|_a^b = F(b) - F(a) \quad (5.1)$$

Nyuton-Leybnis formulasi orqali aniq hisoblash mumkin. Bunda $F(x)$ -berilgan $f(x)$ funksiyaning boshlang'ich funksiyasi. Yuqorida ta'kid-langaniidek, boshlang'ich funksiyaning integrallash qoidalari va formulalar yordamida hisoblash imkoni bo'lmaganda uni integral yig'indilar yordamida taqriban hisoblanadi.



14-rasm

Aniq integralni geometrik ma'nosini **14-rasmdagi** shtrixlangan sohaning yuzasini topish demakdir.

Odatda integral ostidagi funksiyaning boshlang'ich fuksiya-sini topish matematik jihatdan ancha qiyinchilik tug'diradi, ayrim funksiyalar uchun esa boshlang'ich funksiyaning topishning mutlaqo iloji yo'q. Shuning uchun, kerak bo'lgan aniqlikda aniq integrallarni hisoblash algoritmlarining ishlab chiqilishi va dasturlarining yaratilishi dolzarb masalalardan hisoblanadi.

Aniq integralni hisoblash ta'rifiga ko'ra, u

$$\lim_{\max \Delta x_k \rightarrow 0} \sum_{k=1}^n f(\xi_k) \Delta x_k \quad (5.2)$$

limitga teng. Bu yerda $\Delta x_k - [a, b]$ kesmani n -ta bo'lakka bo'lgandagi k -bo'lagining uzunligi, $\xi_k - k$ -kesma ichidagi biror nuqta.

Aniq integralni taqribiy hisoblashning barcha usullari (5.2) formulaga asoslangan bo'lib, Δx_k va ξ larni har xil tanlab olib, turli taqribiy integrallash formulalari hosil qilinadi.

Amalda eng ko'p qo'llaniladigani to'g'ri to'rtburchaklar, trapetsiyalar va Simpson formulalaridir.



Nazorat savollari

1. Integral so'zining ma'nosini bilasizmi?
2. Egri chiziqli trapetsiyaning yuzi qaysi formula bilan hisoblanadi?
3. Aniq integralni hisoblashda qaysi usullardan foydalaniladi?

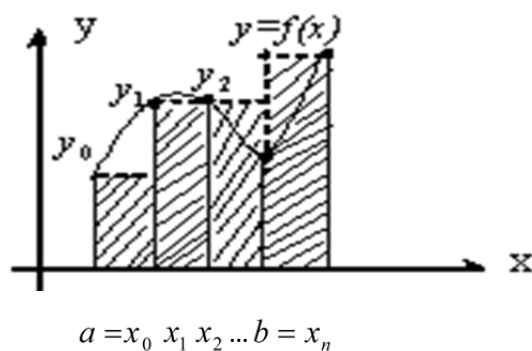
2-§. Tug'ri turtburchaklar formulasi

Tayanch so'z va atamalar



Egri chiziqli soha, bo'linish qadami uzunligi, o'ng to'g'ri to'rtburchaklar usuli, chap to'g'ri to'rtburchaklar usuli, hisoblash formulasi xatoligi, ishchi algoritmgaga mos blok-sxema, dastur ta'minoti.

Integral tarixan egri chiziqlar bilan chegaralangan figuralarning yuzini, xususan egri chiziqli trapetsiyaning yuzini hisoblash munosabati bilan kelib chiqqan. Mazkur usul ham yuzani to'g'ri to'rtburchaklar bilan to'ldirib taqriban hisoblashga asoslangan. Trapetsiyaning asosi bo'lgan $[a;b]$ kesmani $x_1, x_2, \dots, x_{n-1}$ nuqtalar bilan n ta kesmalarga bo'lamiz. U holda bo'linish oralig'i uzunligi $h = \frac{b-a}{n}$ formula bilan ifodalanadi. $x_0 = a$ deb, $x_i = x_{i-1} + h$ nuqtalarni belgilab olamiz, bunda $i = 1, 2, 3, \dots, n$. $[a, b]$ oraliqning tugun nuqtalari $x_1, x_2, x_3, \dots, x_n$ dan chegaraviy $y = f(x)$ egri chiziq bilan kesishgunga qadar vertikal parallel to'g'ri chiziqlar o'tkazamiz va kesishish nuqtalarining ordinatalarini $y(x_1), y(x_2), \dots, y(x_i), \dots$ kabi belgilaymiz. Har bir oraliqdagi ordinatasi uzunligi $y(x_i)$ ga teng to'g'ri to'rtburchakning yuzalarini topamiz $S_i = h \cdot y(x_i)$ (15-rasm):



15-rasm

Hosil qilingan to'g'ri to'rtburchaklarning yuzalarini qo'shamiz:

$$S = h \cdot (y(x_1) + y(x_2) + y(x_3) + \dots + y(x_n)) = h \cdot \sum_{k=1}^n y(x_k)$$

Yuzalarni hisoblashda $k=1,2,3,\dots,n$ deb olsak, vertikal to‘g‘ri chiziqlarga nisbatan o‘ng tomondagi to‘g‘ri to‘rtburchaklar olingani uchun o‘ng to‘g‘ri to‘rtburchaklar usulining formulasi kelib chiqadi:

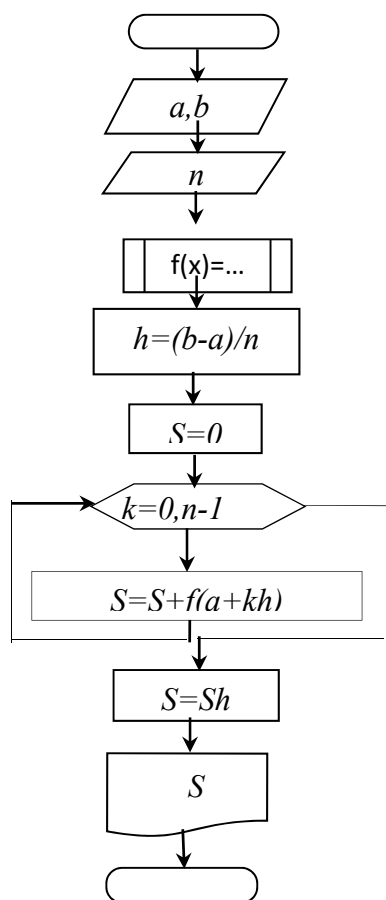
$$S = \int_b^a f(x)dx \approx h[f(a+h) + \dots + f(a+n \cdot h)] = h \cdot \sum_{k=1}^n f(a+k \cdot h) \quad (5.3)$$

$k=0,2,\dots,n-1$ deb olsak, vertikal to‘g‘ri chiziqlarga nisbatan chap tomondagi to‘g‘ri to‘rtburchaklar olingani uchun, chap to‘g‘ri to‘rtburchaklar usulining formulasi kelib chiqadi:

$$S = \int_b^a f(x)dx \approx h[f(a) + f(a+h) + \dots + f(a+(i-1)h)] = h \cdot \sum_{k=0}^{n-1} f(a+kh) \quad (5.4)$$

Agar $f(x)$ funksiya ikki marta differensiallanuvchi bo‘lsa ishchi formulani hisoblash xatoligi $R_n = \frac{(b-a)^3}{2n^2} f''(\xi)$, $a \leq \xi \leq b$ formula bilan aniqlanadi.

To‘g‘ri to‘rtburchaklar usulining ishchi algoritmi blok-sxemasi



Dastur ta`minoti

```

# program Turtburchakli yuza;
a,b,s,h=0,0,0,0
n,k=0,0
def f(x):
    f = 0
    # f = #funksiya kiritilari
    return f
a, b = map(float, input("a,b=").split())
n = int(input("n="))
h=(b-a)/n
s=0
for k in range(n-1):
    s+=f(a+k*h)
s=s*h
print("Natija: ", s)
  
```



Nazorat savollari

1. To'g'ri to'rtburchaklar usulining mohiyati nimada?
2. To'g'ri to'rtburchaklar usulining geometrik ma'nosi qanday tavsiflanadi?
3. Chap va o'ng to'g'ri to'rtburchaklar usulining ishchi formulasi qanday hosil qilinadi?

3-§. Trapetsiya usuli



Tayanch so'z va atamalar

Trapetsiya usulinig ishchi algoritmi, trapetsiya usulining geometrik ma'nosi, Trapetsiya usuli xatoligi, ishchi algoritmgaga mos blok-sxema, dastur ta'minoti.

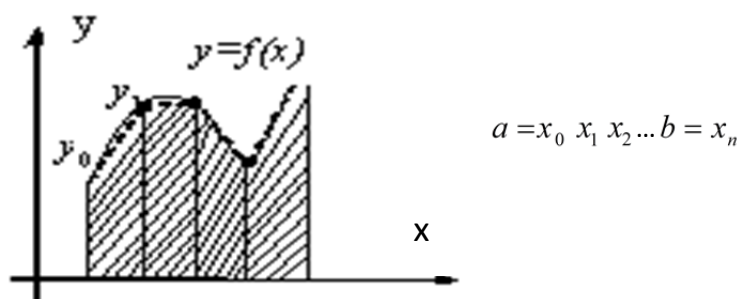
Bu usulda ham to'g'ri to'rtburchaklar usulidagi kabi $[a;b]$ kesmani $a = x_0, x_1, \dots, x_n = b$ nuqtalar bilan n ta teng bo'lakka bo'lamiz. Har bir tugun nuqtalar orasidagi masofa $h = \frac{b-a}{n}$.

$[a;b]$ kesmani bo'luvchi x_i nuqtalardan chegaraviy egri chiziq bilan kesishgunga qadar perpendikulyarlar o'tkazamiz. Egri chiziq mos nuqtalarining ordinatalarini $y_0 = f(x_0), y_1 = f(x_1), \dots, y_{n-1} = f(x_{n-1}), y_n = f(x_n)$ deb belgilaymiz.

Perpendikulyarlarning $y = f(x)$ chiziq bilan kesishgan qo'shni nuqtalarini vatarlar bilan birlashtiramiz va hosil qilingan har bir trapetsiyalarning yuzini topamiz (16-rasm):

$$\frac{y_0 + y_1}{2} \cdot h; \quad \frac{y_1 + y_2}{2} \cdot h; \dots; \frac{y_{n-1} + y_n}{2} \cdot h$$

Barcha n ta trapetsiya yuzini qo'shamiz: $S = h \left[\frac{y_0}{2} + y_1 + y_2 + \dots + \frac{y_n}{2} \right]$.



16-rasm

Demak, egri chiziqli trapetsiyaning yuzi taqriban quyidagiga teng:

$$\int_a^b f(x)dx \approx h \left(\frac{y_0 + y_n}{2} + y_1 + y_2 + \dots + y_{n-1} \right) \text{ yoki } y_0 = f(a), \quad y_n = f(b), \quad y_i = a + ih$$

desak, trapetsiya usulining formulasi

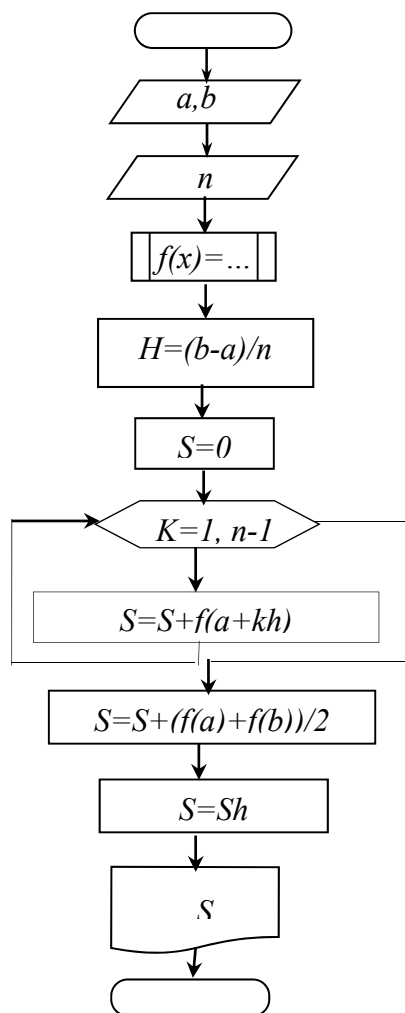
$$S = \int_a^b f(x)dx = h \cdot \left[\frac{f(a) + f(b)}{2} + \sum_{k=1}^{n-1} f(a + kh) \right] \text{ bo'ladi.} \quad \text{Usulning hisoblash}$$

xatoligi $R_n = -\frac{(b-a)^3}{12n^2} f''(\xi)$, $a \leq \xi \leq b$ formula bilan aniqlanadi.

Trapetsiya usuli algoritmining blok-

Dastur ta'minoti

sxemasi



Program Trapetsia;

a,b,h,s=0,0,0,0

n,k=0,0

def f(x):

#f= 0 #funksiya

return f

a,b = map(float, input("a,b = ").split())

n = int(input("n= "))

h = (b-a)/n

s=0

for k in range(n-1):

s+=f(a+k*h)

s+=(f(a)+f(b))/2

s*=h

print("Natija: ", s)



Nazorat savollari

1. Trapetsiya usulining mohiyati nimada?
2. Trapetsiya usulining geometrik ma'nosi qanday tavsiflanadi?
3. Trapetsiya usulining ishchi formulasi qanday hosil qilinadi?

4 -§. Simpson (parabolalar) formulasi



Tayanch soʻz va atamalar

Simpson usulining ishchi algoritmi, trapetsiya usulining geometrik maʼnosi, trapetsiya usulining xatoligi, ishchi algoritmgaga mos algoritm blok-sxemasi, dastur taʼmini.

Aniq integrolni Simpson usulida hisoblashda, oraliqni boʻlish (boʻlinishlar soni juft boʻlishi kerak) natijasida hosil qilingan yuzalarni yuqoridan parabolalar bilan chegaralangan deb faraz qilinadi va bunday yuzani hisoblash aniq integralni boshlangʻich funksiyasini topish hisobiga amalga oshiriladi.

$[a, b]$ kesma uzunligini $h = \frac{b-a}{2n}$ boʻlgan $2n$ ta juft boʻlakka $x_1, x_2, \dots, x_{2n-1}, x_{2n}$ nuqtalar orqali ajratamiz va $[x_0, x_2], [x_2, x_4], \dots, [x_{2n-2}, x_{2n}]$ kesmalarni hosil qilamiz.

Bu kesmalarning oʻrtalari mos ravishda $x_1, x_3, \dots, x_{2n-1}$ nuqtalar boʻladi. U holda hisoblanayotgan aniq integralni

$$\int_a^b f(x)dx = \int_{x_0}^{x_2} f(x)dx + \int_{x_2}^{x_4} f(x)dx + \dots + \int_{x_{2n-2}}^{x_{2n}} f(x)dx$$

koʻrinishidagi integral yigʻindiga ajratamiz. Har bir $[x_{2i}, x_{2i+2}]$ ($i = 0$ dan $n-1$ gacha) kesmalarda $(x_{2i}, y_{2i}), (x_{2i+1}, y_{2i+1}), (x_{2i+2}, y_{2i+2})$ nuqtalar orqali hamma vaqt parabola oʻtkazish mumkin, shu bilan birga bunday parabola $[x_{2i}, x_{2i+2}]$ kesmada yagona boʻladi. Yordamchi parabola bilan chegaralangan egri chiziqli trapetsiya yuzi taqriban berilgan egri chiziqli trapetsiyaning yuziga teng

$$\int_{x_{2i}}^{x_{2i+2}} f(x)dx = \int_{x_{2i}}^{x_{2i+2}} (ax^2 + bx + c)dx$$

Parabola tenglamasiga tegishli har uchta a, b, c nomaʼlum uchun quyidagi sistemani hosil qilamiz:

$$\begin{cases} ax_{2i}^2 + bx_{2i} + c = y_{2i} \\ ax_{2i+1}^2 + bx_{2i+1} + c = y_{2i+1} \\ ax_{2i+2}^2 + bx_{2i+2} + c = y_{2i+2} \end{cases}$$

Hosil boʻlgan a, b, c nomaʼlumli uchta tenglamalar sistemasini yechib, a, b, c larning qiymatini integral ifodaga qoʻyib, aniq integralni Nyuton-Leybnis formulasi bilan hisoblaymiz. Har bir kesmalar uchun ularning qiymatini qoʻshib,

parabolalar usuliga mos ishchi formulani hosil qilamiz.

Usulning ishchi formulasi quyidagi ko‘rinishda yoziladi:

$$S = \int_a^b f(x)dx = \frac{h}{3} \left[f(a) + f(b) + 4 \cdot \sum_{k=1}^n f(a + (2k-1)h) + 2 \cdot \sum_{k=1}^{n-1} f(a + 2kh) \right].$$

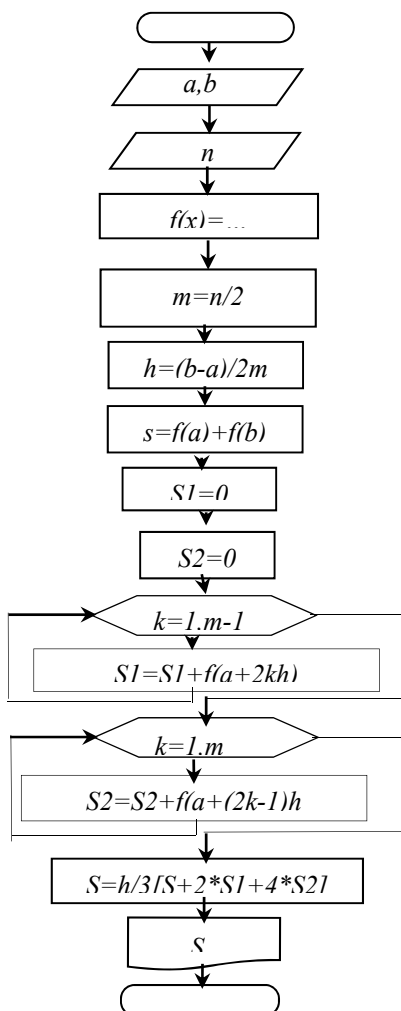
Nazariy tomondan Simpson formulasi yuqoridagi ikki formulaga nisbatan ancha aniqdir, chunki bunda xato

$$R_n(x) = -\frac{(b-a)^5}{180n^4} f^{IV}(\eta), \quad a \leq \eta \leq b$$

formula bilan aniqlanadi. Ammo, xatolik funksiyasi integral ostidagi funksiyaning 4-tartibli hosilasi mavjudligini talab qiladi. Shuning uchun, baʼzi bir funksiyalar uchun Simpson formulasi to‘g‘ri to‘rtburchaklar va trapetsiyalar formulalaridan yomonroq natija berishi mumkin.

Taqribiy qiymatni aniqligini tekshirish uchun aniq integ-rallanadigan funksiya uchun u yoki bu formulani qo‘llab ko‘rish foydali bo‘ladi.

Simpson (marabolalar) usuli Dastur taʼminoti algoritmining blok-sxemasi



```

# Program Simpson;
a,b,h,s,s1,s2=0,0,0,0,0,0
k,m,n=0,0,0
def f(x):
    f = x #functsiya
    return f
a,b = map(float, input("a,b = ").split())
n = float(input("n= "))
m=n*2
h=(b-1)/m
s=f(a)+f(b)
s1=0
s2=0
for k in range(m-1):
    s1+=f(a+2*k*h)
for k in range(m):
    s2+=f(a+(2*k-1)*h)
s=h/3*(s+2*s1+4*s2)
print("s = ", s)
  
```

Olingan natijalar va ularning tahlili.

Ishlab chiqilgan algoritmlarning va yaratilgan dastur-larning to'g'riligini tekshirib ko'rish uchun test misolini tanlab olaylik va uning qiymatini aniqlaylik:

$$\begin{aligned} I &= \int_0^1 (x^3 + 2x^2 - x + 5) dx = \left[\frac{x^4}{4} + 2 \cdot \frac{x^3}{3} - \frac{x^2}{2} + 5x \right] \Big|_0^1 = \\ &= \frac{1}{4} + \frac{2}{3} - \frac{1}{2} + 5 = 5 + \frac{3+8}{12} - \frac{1}{2} = 5 + \frac{11}{12} - \frac{1}{2} = \\ &= 5 + \frac{11-6}{12} = 5 \frac{5}{12} \end{aligned}$$

Demak, integralning aniq qiymati $5\frac{5}{12}$ yoki 5,41(6) ga teng ekan.

Dastur ishlashi uchun zarur bo'lgan boshlang'ich qiymatlar:

$$a=0, \quad b=1, \quad n=20$$

Berilgan qiymatlarni kiritib, yuqoridagi algoritmlar asosida dastur ta'minotini ishlatib ko'ramiz. Ulardan olingan natijalar:

- 1) To'g'ri to'rtburchaklar usulida: $S=5,4236673$
- 2) Trapetsiya usulida: $S=5,41566723$
- 3) Simpson usulida: $S=5,4166666$

Olingan natijalarning barchasi aniq yechimga yaqindar. Lekin, usullardan eng yaxshi natija bergani Simpson usuli bo'lsa, eng yomon natija to'g'ri to'rtburchaklar usulidan olindi. Mantiqan ham olingan natijalar rostdir. Demak, yuqorida berilgan algoritm va dasturlar to'g'ri, amalda ishlatish uchun yaroqli.



Nazorat savollari

1. Simpson usulining mohiyati nimada?
2. Simpson usulining geometrik ma'nosi qanday tavsiflanadi?
3. Simpson usulining ishchi formulasi qanday hosil qilinadi?

XULOSA

Ushbu bobda aniq integralning mohiyati, egri chiziqli trapetsiyaning yuzini topish masalasi, aniq integralni hisoblash usullari va ularning imkoniyatlari haqida zarur ma'lumotlar berildi.

Aniq integralni taqribiy hisoblashning to'g'ri to'rtburchaklar, trapetsiya va Simpson usullari uchun hisoblash algoritmlari va dastur ta'minotlari ishlab chiqildi.

Barcha usullarning mohiyati ularning geometrik ma'nolari yordamida tushuntirildi.

Barcha taqribiy usullar uchun ishlab chiqilgan dastur ta'minotlari aniq masalalar uchun ishlatib ko'rildi, olingan natijalar tahlil etildi.



Bobga doir muammoli vaziyatlar!

- + Matematik modeli aniq integrallar bilan ifodalanadigan amaliy jarayonlarni bilasizmi? Ularga misollar keltira olasizmi?
- + Aniq integralni hisoblovchi Nyuton-Leybnis kabi formulalar mavjud bo'la turib, taqribiy usullarni qo'llash zaruriyati paydo bo'lishining sababi nimada deb o'ylaysiz?
- + Aniq integralni taqribiy hisoblashning qaysi usulida aniqlik yuqori bo'lishi mumkin. Umuman olganda, natijaning aniqligi usulning turiga bog'liqmi?
- + Integral osti funksiyaning xususiyati aniq integralni taqribiy hisoblash usulini tanlashda muhim omil bo'la oladimi? Fikringizni tushuntiring.

6-BOB. ODDIY DIFFERENSIAL TENGLAMALARNI SONLI YECHISH USULLARI

1-§. Differensial tenglamalar va ularni yechish usullari haqida umumiy ma'lumotlar



Tayanch so'z va atamalar

Differensial tenglamalar, oddiy differensial tenglamalar, differensial tenglamaning tartibi, chiziqli differensial tenglamalar, umumiy yechim, xususiy yechim, differensial tenglamani yechish usullari, aniq usullar, taqribiy analitik usullar, sonli usullar.

Ma'lumki, ko'pincha amaliy masalalarni yechishda, dastlab uning matematik modeli fizik, mexanik, kimyoviy va boshqa qonuniyatlar asosida tuziladi. Matematik model asosan algebraik, differensial, integral va boshqa tenglamalardan iborat bo'ladi. Ayniqsa, oddiy differensial tenglamalar juda ko'p muhandislik masalalarini yechishda matematik model rolini o'ynaydi. Shuning uchun, differensial tenglamalarning ma'lum shartlarni qanoatlantiruvchi yechimlarini topish katta ahamiyatga ega.

Differensial tenglamalar ikkita asosiy sinfga bo'linadi: oddiy differensial tenglamalar va xususiy hosilali differensial tenglamalar.

Xususiy hosilali differensial tenglamalarga keyinroq batafsil to'xtalamiz.

Oddiy differensial tenglamalarda faqat bir o'zgaruvchiga bog'liq funksiya va uning hosilalari qatnashadi, ya'ni

$$f(x, y, y', \dots, y^{(n)}) = 0 \quad (6.1)$$

(6.1) tenglamada qatnashuvchi hosilalarning eng yuqori tartibi differensial tenglamaning tartibi deyiladi. Agar tenglama izlanuvchi funksiya va uning hosilalariga nisbatan chiziqli bo'lsa, unga chiziqli differensial tenglama deyiladi.

Differensial tenglamaning umumiy yechimi deb, uni ayniyatga aylantiruvchi x va n ta $c_1, c_2, c_3, \dots, c_n$ o'zgarmaslarga bog'liq ixtiyoriy funksiya aytiladi. Masalan (6.1) tenglamaning umumiy yechimi $y = \varphi(x, c_1, c_2, \dots, c_n)$ ko'rinishdagi funksiyalardan iborat. Agar $c_1, c_2, c_3, \dots, c_n$ o'zgarmaslarga muayyan qiymatlar

berilsa, umumiy yechimdan xususiy yechim hosil qilinadi. Xususiy yechimni topish uchun $c_1, c_2, c_3, \dots, c_n$ o'zgarmlarining mos qiymatlarini aniqlash lozim. Buning uchun esa yechim qanoatlantiruvchi qo'shimcha shartlarga ega bo'lishimiz kerak. Agar differensial tenglama n -tartibli bo'lsa, yagona xususiy yechimni topish uchun xuddi shuncha qo'shimcha shartlar kerak. Hususan, birinchi tartibli tenglama $F(x, y, y') = 0$ ning umumiy yechimi $y = \varphi(x, c)$ dagi c o'zgarmlarni topish uchun bitta qo'shimcha shartning berilishi kifoya.

Qo'shimcha shartlar berilishiga ko'ra differensial tenglamalar uchun ikki xil masala qo'yiladi:

- 1) *Koshi masalasi*
- 2) *Chegaraviy masala.*

Agar qo'shimcha shartlar bitta $x = x_0$ nuqtada berilsa, differensial tenglamani yechish uchun qo'yilgan masalani Koshi masalasi deyiladi. Koshi masalasidagi qo'shimcha shartlar boshlang'ich shartlar, $x = x_0$ nuqta esa boshlang'ich nuqta deb ataladi.

Agar qo'shimchi shartlar erkli o'zgaruvchi argumentlarning ikki yoki undan ko'p qiymatlarida berilsa, bunday masalaga chegaraviy masala deyiladi. Qo'shimcha shartlar esa chegaraviy shartlar deb ataladi.

Oddiy differensial tenglamalarni yechishning chizma, analitik, taqribiy va sonli yechish usullari mavjud.

Chizma usullarda differensial tenglamaning integral chiziqlarini geometrik tasviri yasaladi. Bunda hosila o'zgarmlar bo'lgandagi integral chiziqlar-izoklinalar tuziladi. Bu usuldan asosan sodda ko'rinishdagi differensial tenglamalarni yechishda foydalaniladi.

Analitik usullarda differensial tenglamaning yechimlari aniq formulalar orqali aniqlanadi.

Taqribiy usullarda differensial tenglama va qo'shimcha shartlar u yoki bu darajada soddalashtirilib, masala osonroq masalaga keltiriladi.

Sonli usullarda esa yechim analitik shaklda emas, balki sonlar jadvali ko‘rinishida olinadi. Albatta, bunda differensial tenglamalar oldin diskret tenglamalar bilan almashtirib olinadi. Natijada, sonli usullar vositasida olingan yechim taqribiy bo‘ladi.

Umuman olganda, oddiy differensial tenglamalarning yechimlarini analitik usul yordamida topish imkoni juda kam bo‘lganligi uchun, amalda ko‘pincha ularni sonli usullar yordamida taqribiy hisoblanadi.

Quyida Koshi masalasini sonli yechish usullaridan na‘muna sifatida Eyler va Runge-Kutta usullarini ko‘rib chiqamiz.



Nazorat savollari

1. Differensial tenglamalar qanday sinflarga bo‘linadi?
2. Qanday tenglamalar oddiy differensial tenglamalar hisoblanadi?
3. Differensial tenglamaning umumiy yechimi nima?
4. Differensial tenglamalarnig xususiy yechimi qanda aniqlanadi?
5. Differensial tenglamalar yechishning qanday usullarini bilasiz?

2-§. Koshi masalasini yechishning Eyler usuli

Tayanch so‘z va atamalar



Koshi masalasi, boshlang‘ich shart, Eyler usulining ishchi formulasi, usulning geometrik ma‘nosi, Eyler usulining xatoligi, usulga mos algoritm blok-sxemasi, dasturi

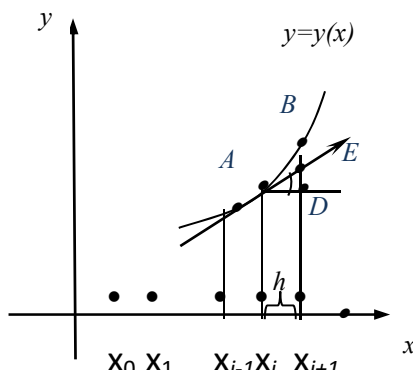
Bizga quyidagi birinchi tartibli oddiy differensial tenglama(Koshi masalasi)ni

$$y' = f(x, y) \quad (6.2)$$

$[a, b]$ oraliqdagi $y_0 = y(x_0)$ boshlang‘ich shartni qanoatlantiruvchi aniq yechimi $\bar{y} = \bar{y}(x)$ ni topish lozim bo‘lsin.

Koshi masalasini Eyler usuli yordamida yechish uchun, dastlab differensial tenglamaning yechimi qidiriladigan $[a, b]$ kesmani $x_0, x_1, \dots, x_n$ tugun nuqtalar bilan bo‘laklarga bo‘lamiz. Tugun nuqtalarning koordinatalari $x_{i+1} = a + (i + 1)h$, (

$i = \overline{0, n-1}$) formula orqali aniqlanadi. Har bir tugunda $y(x_i)$ echimning qiymatlarini chekli ayirmalar yordamida taqribiy y_i qiymatlar bilan almashtiriladi.



17-rasm

Ma'lumki, $\bar{y} = \bar{y}(x)$ funksiyaning $x = x_i$ nuqta atrofida Teylor qatoriga yoyilmasini quyidagicha yozish mumkin:

$$\bar{y}(x_{i+1}) = \bar{y}(x_i) + h \cdot \bar{y}'(x_i) + \frac{1}{2} h^2 \cdot \bar{y}''(x_i) + \dots$$

Ushbu cheksiz qatorning boshidagi ikkita had bilan chegaralanib, birinchi tartibli hosila qatnashgan hadni aniqlash natijasida quyidagi chekli ayirmali formulani hosil qilamiz:

$$\bar{y}'(x_i) = \frac{\bar{y}(x_{i+1}) - \bar{y}(x_i)}{h} + O(h) \quad (6.3)$$

Ushbu almashtirishning geometrik ma'nosi haqida fikr bildiraylik. Hosilaning geometrik ma'nosiga ko'ra (17-rasm)

$$\bar{y}'(x_i) = \tan \beta = \frac{ED}{AD} = \frac{ED}{h}$$

(6.3) dan

$$\bar{y}'(x_i) \approx \frac{\overbrace{y_{i+1} - y_i}^{BD}}{h} = \frac{BD}{h} = \frac{ED}{h} + \frac{BE}{h} = \bar{y}'(x_i) + \frac{BE}{h}$$

Demak, chekli ayirmalar formulasi hosilaning asl qiymatidan BE/h ga farq qiladi, ya'ni BE qancha kichik bo'lsa, chekli ayirma $\bar{y}'$ hosilaga shuncha yaqin bo'ladi. Rasmdan $h \rightarrow 0$ da $BE \rightarrow 0$ ekanini ko'rish mumkin. (6.2) va (6.3) dan

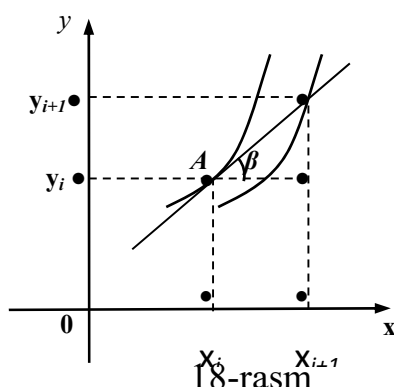
$\bar{y}'_i = f(x_i, \bar{y}_i)$ ekanini hisobga olib, quyidagini hosil qilamiz:
 $\bar{y}_{i+1} = \bar{y}_i + h \cdot f(x_i, \bar{y}_i) + O(h^2)$.

Yangi taqribiy $y_i \approx \bar{y}_i$ qiymatlarni ushbu formula bilan kiritamiz:

$$y_{i+1} = y_i + h \cdot f(x_i, y_i) \quad (6.4)$$

Hosil qilingan (6.4) formula Eyler usulining asosiy ishchi formulasi bo'lib, uning yordamida tugun nuqtalarga mos bo'lgan differensial tenglamaning y_i xususiy yechimlarini topish mumkin. Yuqoridagi formuladan ko'rinib turibdiki, y_{i+1} yechimni topish uchun y_i yechimnigina bilish kifoya.

Eyler usulining geometrik ma'nosi bilan tanishib chiqaylik:



A nuqta $x = x_i$ nuqtaga mos keluvchi yechim bo'lsin. Bu nuqtadan integral chiziqqa o'tkazilgan urinma x_{i+1} nuqtada boshqa integral chizig'ida y_{i+1} yechimni aniqlaydi.

Urinmaning og'maligi $\beta \cdot y'_i = f(x_i, y_i)$ hosila bilan aniqlanadi. Demak, Eyler usulidagi yo'l qo'yilgan asosiy xatolik yechimni bir integral chizig'idan boshqasiga o'tkazib yuborishi bilan xarakterlanadi. $y = f(x)$ funksiyaning $x = x_i$ nuqta atrofida Teylor qatoriga yoyib, bu qatordagi chiziqli xadlar bilangina cheklanganimiz uchun (6.4) formulani

$$\bar{y}_{i+1} = \bar{y}_i + h \cdot f(x_i, \bar{y}_i) + O(h^2)$$

kabi yozish mumkin. Har bir qadamda Eyler usuli xatoligi $O(h^2)$ bo'ladi.

Qidirilayotgan yechimni n -ta nuqtada topilishini inobatga olsak bu xatoliklar jamlanib borib $n \cdot O(h^2)$ ga teng bo'ladi. Lekin, $h = L/n$, L -integrallanuvchi kesma uzunligi, bo'lganligi uchun usulning yakuniy xatosi

$$n \cdot 0(h^2) = L / n \cdot 0(h^2) = L \cdot 0(h) = 0(h)$$

ga teng bo'ladi.

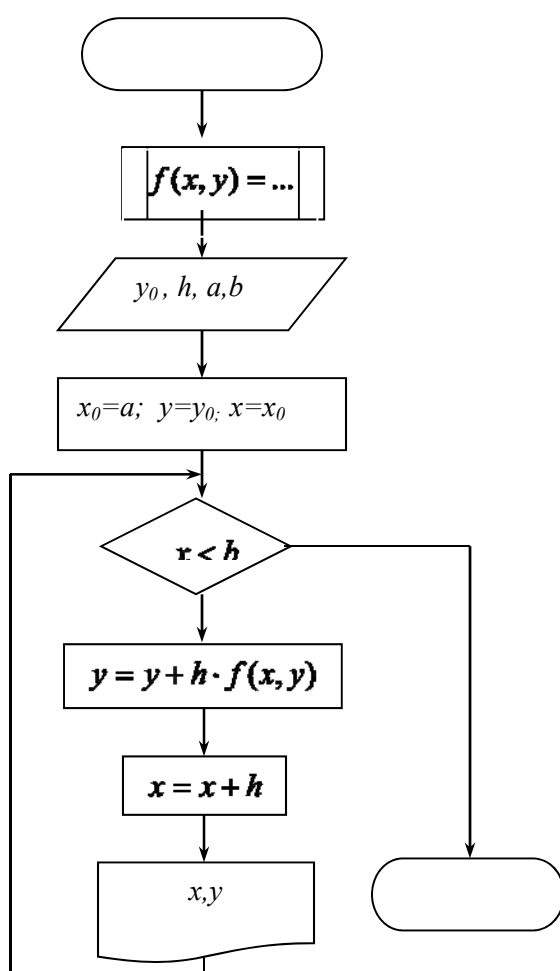
Shunday qilib, usulning aniqlik darajasi integrallash oralig'ini bo'linish qadamiga bog'liq bo'lib, bu qadam qanchalik kichik bo'lsa, yo'l qo'yilgan xatolik ham shunchalik kichik bo'ladi.

Taylor qatoridagi hadlarni ko'proq ushlab qolish evaziga turli xil ko'rinishdagi takomillashgan formulalarga ega bo'lish mumkin.

Eyler usuliga mos algoritm blok-

Algoritmning dastur matni

sxemasi



Program Eyler;

a,b,x0,y0,x,y,h=0,0,0,0,0,0,0

def f(x):

f = x

return f

a,b = map(float, input("a,b = ").split())

y0 = float(input("y0= "))

x=x0

y=y0

while x<b:

y+=h*f(x,y)

print(f"x={x} y={y}")

x+=h



Nazorat savollari

1. Eyler usulining ishchi formulasi qanday hosil qilinadi?
2. Eyler usulining geometrik ma'nosini ifodalay olasizmi?
3. Eyler usulining xatoligi qanday baholanadi?



3-§. Koshi masalasini yechishning Runge-Kutta usuli

Tayanch soʻz va atamalar

Runge – Kutta usuli, usulning ishchi formulasi, algoritm blok-sxemasi, datur tavminoti.

Bir qadamli oshkor usullarning boshqa bir necha xillari ham mavjud boʻlib, ularning ichida amalda eng koʻp ishlatiladigani Runge-Kutta usuli hisoblanadi. Usul shartiga koʻra har bir yangi x_{i+1} tugun nuqtadagi y_{i+1} yechimni topish uchun $f(x, y)$ funksiyani toʻrt marta har xil argumentlar uchun hisoblash kerak. Bu jihatdan

Runge-Kutta usuli hisoblash uchun nisbatan koʻp vaqt talab qiladi. Lekin, Eyler usuliga koʻra aniqligi yuqori boʻlganligi uchun, undan amalda keng foydalaniladi. Usulning ishchi formulasi quyidagicha yoziladi:

$$y_{i+1} = y_i + \frac{h}{6}(k_0 + 2k_1 + 2k_2 + k_3), \quad i = 0, 1, \dots, n-1,$$

bu yerda

$$k_0 = f(x_i, y_i);$$

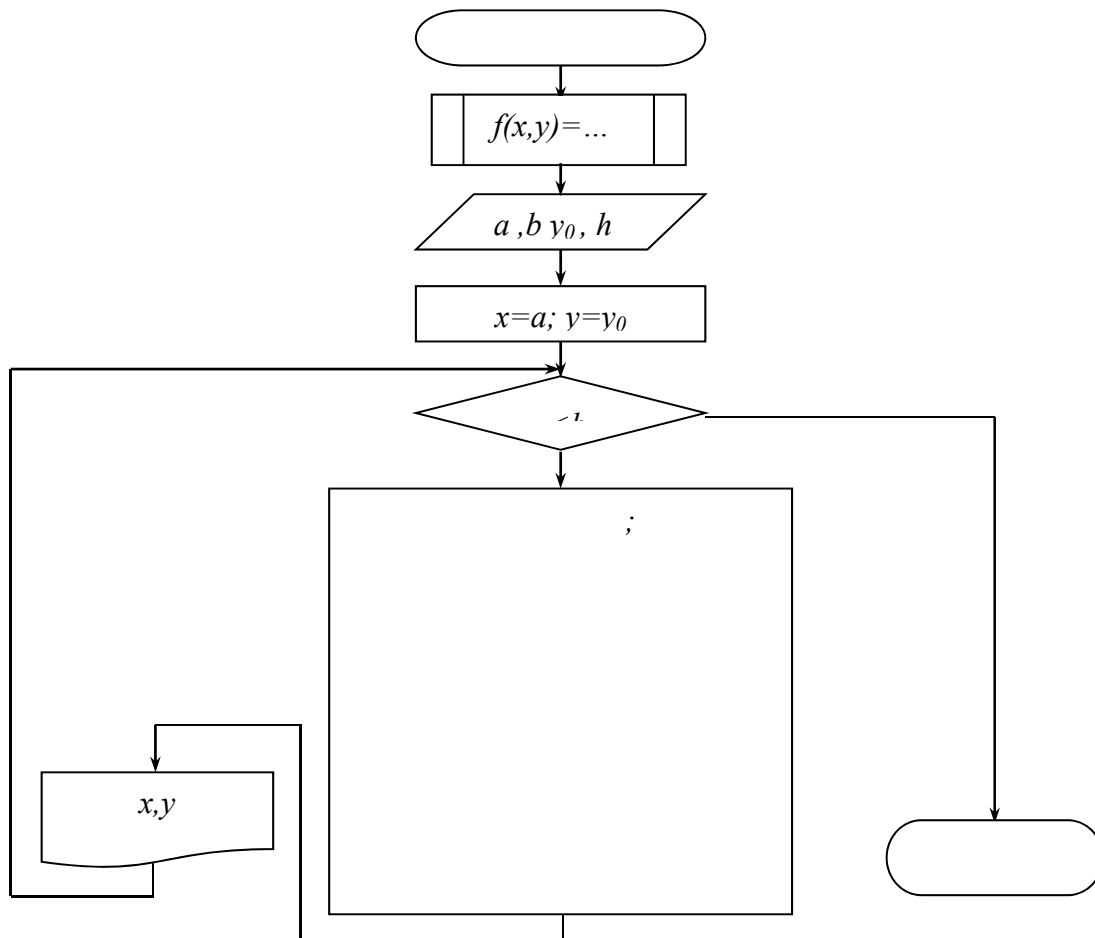
$$k_1 = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2} \cdot k_0\right);$$

$$k_2 = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2} \cdot k_1\right);$$

$$k_3 = f(x_i + h, y_i + h \cdot k_2);$$

Demak, formulalardan koʻrinib turibdiki, Eyler usuli birinchi tartibli Runge-Kutta usuliga mos keladi. Runge-Kutta usulining aniqligi $O(h^4)$ kichik miqdor bilan baholanadi, yaʼni Eyler usuliga nisbatan toʻrt marta yuqori aniqlikka egadir.

Runge-Kutta usuliga mos blok-sxema



Algoritmning dastur matni

```
# Program R_kutta;
a,b,x0,y0,x,y,k0,k1,k2,k3=0,0,0,0,0,0,0,0,0,0,0
def f(x):
    f = x #funksiya
    return f
a,b = map(float, input("a,b = ").split())
y0,h = map(float, input("y0,h = ").split())
x=a
y=y0
while x<b:
    k0=f(x,y)
    k1=f(x+h/2,y+h*k0/2)
    k2=f(x+h/2,y+h*k1/2)
    k3=f(x+h,y+h*k2)
    y=y+(k0+2*k1+2*k2+k3)*(h/6)
    x+=h
print(f"x={x} y={y}")
```

Yuqorida ko‘rib chiqilgan dasturlarning to‘g‘riligini va usullarning anqlik darajasini tekshirish uchun bitta ixtiyoriy differensial tenglama olamiz. Aniq yechimni analitik usulda hisoblash qulay bo‘lishi uchun quyidagi tenglamani ko‘rib chiqamiz:

$y' = \cos(x)$ tenglamaning $[0,1]$ oraliqda $h = 0,1$ qadam bilan $y(0) = 1$ boshlang‘ich shartni qanoatlantiruvchi yechimini topish kerak.

Yuqoridagi dasturlarga kerakli qiymatlarni kiritamiz. $x_0 = 0$; $y_0 = 1$;
 $f(x) = \cos x$; $a = 0$; $b = 1$; $h = 0,1$

Yuqoridagi tenglama uchun aniq yechim sifatida $y = \sin x + c$ ni olamiz. Uni boshlang‘ich shartlarga qo‘yib: $1 = \sin 0 + c$ noma‘lum o‘zgarmasni aniqlaymiz: $c = 1$. Demak, differensial tenglamaning xususiy yechimi: $y = \sin x + 1$.

Olingan natijalarga mos qiymatlardan iborat jadval tuzamiz.

xi	Eyler usuli uchun	Runge-Kutta usuli uchun	Aniq yechim
0.1	1,1000	1,0998	1,0998
0.2	1,1995	1,1986	1,1986
0.3	1,2975	1,2955	1,2955
0.4	1,3930	1,3894	1,3894
0.5	1,4851	1,4794	1,4794
0.6	1,5729	1,5646	1,5466
0.7	1,6554	1,6442	1,6442
0.8	1,7319	1,7173	1,7173
0.9	1,8015	1,7833	1,7833
1	1,8637	1,8414	1,8414

Natijalardan ko‘rinib turibdiki, haqiqatan ham Runge- Kutta usulidan olingan natijalar Eyler usulidan olingan natijalarga ko‘ra aniq yechimga ancha yaqindir.



Nazorat savollari

1. Runge – Kutta usulining ishchi formulasi qanday hosil qilinadi?
2. Runge – Kutta usulining aniqligi boshqa usullarga qaraganda nima uchun aniqroq deb hisoblanadi?
3. Runge – Kutta usulida yechim aniqligini yanada oshirish imkoni mavjudmi?

4-§. Chegaraviy masala va uni yechish usullari



Tayanch soʻz va atamalar

Chegaraviy masala, chegaraviy shart, chegaraviy masalani yechish usullari, analitik usullar, sonli-taqribiy usullar, taqribiy-analitik usullar.

Oldingi paragraflarda taʼkidlab oʻtganimizdek, differensial tenglamalar orqali juda ham koʻp va turli-tuman jarayonlarning matematik modellari ifodalanadi. Maʼlumki, amaliyotchilarni differensial tenglamalarning umumiy yechimlari emas, balki qandaydir qoʻshimcha shartlarni qanoatlantiruvchi xususiy yechimlari koʻproq qiziqtiradi. Qoʻshimcha shartlar esa oʻzlarining qoʻyilish maʼnosiga koʻra boshlangʻich va chegaraviy shartlarga boʻlinadi. Boshlangʻich shartli differensial tenglamalarni yechish yoʻllari bilan oldingi paragrafda tanishib oʻtdik.

Chegaraviy masalalarda differensial tenglamalarni qaralayotgan sohaning chegaralaridagi shartlarni qanoatlantiruvchi xususiy yechimlarini topish masalasi oʻrganiladi. Odatda, chegaraviy shartlar integrallash sohasini chegaralarida berilib quyidagi masalalarga boʻlinadi: Dirixle masalasi, Neyman masalasi va aralash masala. Endi chegaraviy masalalarni qoʻyilishi va ularni yechish usullari boʻyicha batafsil toʻxtalib oʻtaylik. Odatda, chegaraviy masalani yechishni oʻrganishni ikkinchi tartibli, oʻzgaruvchan koeffisientli oddiy differensial tenglamalarni turli xil chegaraviy shartlarda yechish orqali amalga oshiriladi.

Shunday qilib, bizga quyidagi ikkinchi tartibli oddiy differensial tenglamaning

$$y''(x) + P(x)y'(x) + Q(x)y(x) = f(x) \quad , \quad a < x < b \quad (6.5)$$

integral oraligʻining chetki nuqtalari $x = a$ va $x = b$ larda berilgan

$$m_0 y(a) + m_1 y'(a) = m_2, g_0 y(b) + g_1 y'(b) = g_2 \quad (6.6)$$

chegaraviy shartlarni qanoatlantiruvchi aniq yechimi $\bar{y} = \bar{y}(x)$ ni topish kabi chegaraviy masalani yechish masalasi qo'yilgan bo'lsin. Bu yerda $P(x), Q(x), f(x)$ - $[a, b]$ oraliqda berilgan uzluksiz funksiyalar, $m_0, m_1, m_2, g_0, g_1, g_2$ - berilgan sonlar, ularni chegaraviy shart belgilari deb ham ataladi. Bu o'zgarmaslar baravariga nolga teng emas, ya'ni

$$|m_0| + |m_1| \neq 0 \text{ va } |g_0| + |g_1| \neq 0$$

Chegaraviy shart belgilariga turli xil qiymatlarni berish orqali, berilgan masalani yechish uchun har xil chegaraviy shartlar hosil qilinishi mumkin.

Ayrim paytlarda yechilishi lozim bo'lgan masalalarning matematik modellari to'rtinchi tartibli oddiy differensial tenglamalar orqali ham ifodalanishi mumkin.

Amalda ko'pincha to'rtinchi tartibli differensial tenglamalarning quyidagi ko'rinishi uchraydi:

$$y^{(4)}(x) = k \cdot f(x)$$

bu yerda k qiymati beriluvchi koeffisient hisoblanadi. Bu differensial tenglama uchun ma'lum belgilashlarni kiritib, uni ikkinchi tartibli differensial tenglamalar sistemasiga keltirish mumkin.

$y(x)$ noma'lum funksiyani $y_1(x)$ funksiya orqali belgilab olib, quyidagi almashtirishlar qilamiz, ya'ni:

$$\begin{cases} y_1''(x) = y_2(x) \\ y_2''(x) = k \cdot f(x) \end{cases}$$

Shunday qilib, to'rtinchi tartibli differensial tenglamaning o'rniga ham ikkita, ikkinchi tartibli differensial tenglamalar sistemasini hosil qilishimiz mumkin. Shuning uchun, biz asosan ikkinchi tartibli differensial tenglamalarning chegaraviy shartlarni qanoatlantiruvchi xususiy yechimlarini topishni o'rganamiz.

Avval ta'kidlab o'tganimizdek, ikkinchi tartibli differensial tenglamalarda xususiy yechimni ajratib olish uchun ikkita qo'shimcha shart, yani chegaraviy shartlar berilgan bo'lishi lozim. Bundan buyon, qulaylik uchun ikkinchi tartibli

oddiy differensial tenglamalarni berilgan chegaraviy shartlarni qanoatlantiruvchi yechimlarini topish masalasini oddiygina qilib chegaraviy masala deb yuritamiz.

Chegaraviy masalalarni yechish usullarini quyidagi guruhlarga bo'lish mumkin:

Analitik usullar;

Sonli-taqribiy usullar;

Taqribiy-analitik usullar.

1. **Analitik usullar.** Oddiy differensial tenglamalarni analitik usullar bilan yechishni oliy matematika kursida o'rganganmiz. Unda chiziqli, bir jinsli bo'lmagan differensial tenglamaning umumiy yechimi bu tenglamaning xususiy yechimi va mos bir jinsli tenglamaning umumiy yechimi yig'indisidan iboratdir. Chiziqli, bir jinsli tenglamalarning umumiy yechimini topish uchun esa uning xususiy yechimlari fundamental sistemasini topish kerak bo'ladi. Xususiy yechimlarni differensial tenglamalarga mos xarakteristik tenglamalar yordamida topiladi. Yuqoridagi barcha bajariladigan amallar differensial tenglamaning ko'rinishi juda sodda bo'lgandagina biror-bir natija berishi mumkin.

Demak, analitik usullar bilan barcha ikkinchi tartibli differensial tenglamalarni yechish imkoni deyarli yo'q.

2. **Sonli-taqribiy usullar.**

Sonli-taqribiy usullarda yechim sonlar yoki sonlar jadvali ko'rinishida olinadi. Albatta, bunda differensial tenglamalar oldin diskret tenglamalar bilan almashtirib olinadi. Sonli usullarning imkoniyatlari boshqa taqribiy usullarga qaraganda ancha kengdir. Sonli usullar ikki guruhga bo'linadi:

Chegaraviy masalalarni Koshi masalasiga keltiruvchi usullar;

Chekli ayirmalar usuli.

Biz ko'proq e'tiborni sonli usullar ichida eng ko'p ishlatiladigani - chekli ayirmali usullarga qaratamiz.

3. **Taqribiy-analitik usullar.**

Bu usulda differensial tenglama va qo'shimcha shartlar u yoki bu darajada soddalashtirilib, masala osonroq masalaga keltiriladi. Taqribiy-analitik usullarga

Galyorkin, eng kichik kvadratlar , kollokasiya , Rits va boshqa usullar kiradi. Amalda eng ko‘p ishlatiladigan usullardan biri bu Galyorkin usulidir.



Nazorat savollari

1. Chegaraviy masalani yechish uchun qanday qo‘shimcha shartlardan foydalanish yetarli hisoblanadi?
2. Chegaraviy masalalarni yechish usullarini qaysi guruhlariga bo‘linadi?

5-§. Chekli ayirmalar usuli

Tayanch so‘z va atamalar



Chekli ayirmali formula, o‘ng chekli ayirmali formula, markaziy chekli ayirmali formula, uch diaganalli tenglamalar sistemasi, haydash usuli, noma‘lum haydash koefitsientlari, to‘g‘ri va teskari bosqichlar.

Bizga quyidagi

$$y''(x) + p(x)y'(x) + q(x)y(x) = f(x) \quad (6.7)$$

Ikkinchi tartibli, o‘zgaruvchan koefitsientli oddiy differensial tenglamaning $x \in [a, b]$ oraliqning chetki nuqtalarida qo‘yilgan

$$\{m_0 y(a) + m_1 y'(a) = m_2, g_0 y(b) + g_1 y'(b) = g_2 \quad (6.8)$$

chegaraviy shartlarni qanoatlantiruvchi aniq yechim $\bar{y} = \bar{y}(x)$ ni topish lozim bo‘lsin. Bu yerda $p(x), q(x), f(x)$ lar $[a, b]$ oraliqda uzluksiz funksiyalar sinfiga kiradi. $m_0, m_1, m_2, g_0, g_1, g_2$ - o‘zgarmaslar, ya‘ni chegaraviy shart belgilari. Ixtiyoriy taqribiy yechimni $y = y(x)$ deb belgilaymiz.

Yuqoridagi masalani sonli-taqribiy usul hisoblanmish chekli ayirmalar usuli bilan yechish uchun yechim qidiriladigan $[a, b]$ oraliqda quyidagi to‘rni kiritamiz, ya‘ni oraliqni koordinatalari $x_i = a + i \cdot h$ formula bilan aniqlanuvchi tugun nuqtalar

bilan bo‘laklarga bo‘lamiz, bu yerda $h = \frac{b-a}{n}$, n -tugun nuqtalar soni. Belgilashlar

kiritamiz: $\bar{y}_i = \bar{y}(x_i)$, $y_i = y(x_i)$.

x_i nuqtalar uchun yuqoridagi (6.7) tenglama o‘rinli bo‘lgani uchun, uni shu nuqtalarda yozib olamiz:

$$\bar{y}''(x_i) + p(x_i)\bar{y}'(x_i) + q(x_i)\bar{y}(x_i) = f(x_i)$$

Qulaylik uchun, bu tenglamani quyidagi ko‘rinishda qayta yozamiz:

$$\bar{y}''_i + p_i\bar{y}'_i + q_i\bar{y}_i = f_i \quad (6.9)$$

Ma'lumki, izlanuvchi y_i funksiyaning x_i nuqta atrofidagi Teylor qatoriga yoyilmasini quyidagicha ifodalash mumkin:

$$\bar{y}(x_{i+1}) = \bar{y}(x_i) + h\bar{y}'(x_i) + \frac{h^2}{2!}\bar{y}''(x_i) + \dots \quad (6.10)$$

yoki

$$\bar{y}(x_{i-1}) = \bar{y}(x_i) - h\bar{y}'(x_i) + \frac{h^2}{2!}\bar{y}''(x_i) + \dots \quad (6.11)$$

(6.10) va (6.11) qatordagi ikki va undan yuqori tartibli hosilalar qatnashgan hadlarni tashlab yuborsak, izlanuvchi funksiyaning x_i nuqtadagi hosilalari uchun quyidagi taqribiy hisoblash formulalari hosil bo‘ladi.

(6.10) formuladan

$$\bar{y}'(x_i) = \frac{\bar{y}(x_{i+1}) - \bar{y}(x_i)}{h} + O(h) \quad (6.12)$$

(6.11) formuladan

$$\bar{y}'(x_i) = \frac{\bar{y}(x_i) - \bar{y}(x_{i-1})}{h} + O(h) \quad (6.13)$$

(6.12)-formula o‘ng chekli ayirmali formula, (6.13)-formula chap chekli ayirmali formula deb ataladi. Bu formulalar $O(h)$ miqdorli xatoliklar bilan baholanadi.

Endi (6.10) va (6.11) Teylor qatoridagi uchinchi va undan yuqori tartibli hosilalar qatnashgan hadlarni tashlab yuborib, hosil bo‘lgan taqribiy tengliklarni

ayirish hisobiga birinchi tartibli hosilani taqribiy hisoblashning markaziy chekli ayirmali formulasini hosil qilamiz:

$$\bar{y}'(x_i) = \frac{\bar{y}(x_{i+1}) - \bar{y}(x_{i-1}))}{2h} + O(h^2) \quad (6.14)$$

bu almashtirishning xatolik darajasi $O(h^2)$ miqdor bilan belgilanadi.

Agar yuqoridagi (6.10) va (6.11) formulalardagi ikkinchi tartibli hosila qatnashgan hadni ham qo'shib olib, hosil bo'lgan tengliklarni hadlab qo'shsak:

$$\bar{y}_i'' = \frac{\bar{y}(x_{i+1}) - 2\bar{y}(x_i) + \bar{y}(x_{i-1}))}{h^2} + O(h^2) \quad (6.15)$$

dan iborat izlanuvchi y_i funksiyaning x_i nuqtalari uchun ikkinchi tartibli hosilasini taqribiy hisoblash formulasi kelib chiqadi. Bu almashtirishning xatoligi ham $O(h^2)$ miqdor bilan baholanadi.

(6.9) differensial tenglamadagi $\bar{y}_i', \bar{y}_i''$ lar o'rniga hosil qilingan chekli ayirmali formulalarni qo'yamiz:

$$\frac{\bar{y}_{i+1} - 2\bar{y}_i + \bar{y}_{i-1}}{h^2} + p_i \frac{\bar{y}_{i+1} - \bar{y}_{i-1}}{2h} + q_i \bar{y}_i + O(h^2) = f_i.$$

Cheksiz kichik miqdorlarni tashlab yuborib, hosilalar qatnashmagan va $y_i \approx \bar{y}(x_i)$ noma'lumlardan iborat tenglamalarni hosil qilamiz:

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + p_i \frac{y_{i+1} - y_{i-1}}{2h} + q_i y_i = f_i, i = 1, \dots, n-1.$$

Hosil bo'lgan tenglamani har ikkala tomonini h^2 ga ko'paytiramiz va mos hadlarni guruhlaymiz. Hamda ushbu belgilashlar kiritib:

$$A_i = 1 + \frac{h}{2} p_i, \quad B_i = 2 - h^2 q_i, \quad C_i = 1 - \frac{h}{2} p_i, \quad D_i = h^2 f_i \quad (6.16)$$

quyidagi tenglamalar sistemasini hosil qilamiz:

$$A_i y_{i+1} - B_i y_i + C_i y_{i-1} = D_i \quad (6.17)$$

Bu yerda $i = \overline{1, n-1}$ bo'lgani uchun i ga mos qiymatlarni berib, (6.17) sistemaning yoyib yozilgan xolini hosil qilamiz:

(6.18)

Uch diagonalli bo'lishiga sabab, sistemadagi har bir tenglamada faqat uchta noma'lum qatnashgan hadlar mavjud bo'lib, sistemada ularning joylashgan o'ri asosiy diagonal, uni pasti va yuqorisidagi diagonallarga mos keladi.

$x_0 = a$ va $x_n = b$ oraliqning chetki nuqtalari uchun (6.8) shartlarni quyidagicha yozib olamiz:

$$\begin{cases} m_0 y_0 + m_1 y_0' = m_2 \\ g_0 y_n + g_1 y_n' = g_2 \end{cases}$$

y'_0, y'_n -larni mos ravishda (6.11) va (6.12) chekli ayirmali formulalari bilan almashtiramiz, ya'ni $y(x)$ ni $x = x_0$ yoki $x = a$ nuqtadagi hosilasi uchun o'ng chekli ayirma formulasini, $x = x_n$ yoki $x = b$ nuqtadagi hosilasi uchun chap chekli ayirma formulasini qo'yamiz:

$$\begin{cases} m_0 y_0 + m_1 \cdot \frac{y_1 - y_0}{h} = m_2 \\ g_0 y_n + g_1 \frac{y_n - y_{n-1}}{h} = g_2 \end{cases}$$

Hosil bo‘lgan tenglamalarni h ga ko‘paytirib, o‘xshash hadlarni ixchamlaymiz:

$$(6.19)$$

Quyidagicha belgilashlarni kiritib:

$$A_0 = hm_0 - m_1, D_0 = hm_2, B_n = -g_1, B_0 = m_1, A_n = hg_0 + g_1, D_n = hg_2 \quad (6.20)$$

hosil qilingan tenglamalarni (6.17) tenglamalar sistemasiga “ulaymiz” va natijada $(n+1)$ ta noma'lumli, $(n+1)$ ta tenglamadan iborat $y_0, y_1, \dots, y_n$ noma'lumlarga nisbatan yozilgan quyidagi uch dioganalli chiziqli algebraik tenglamalar sistemasiga ega bo'lamiz:

$$\begin{cases} B_0 y_0 + C_0 y_1 = D_0 \\ A_i y_{i+1} - B_i y_i + C_i y_{i-1} = D_i \quad (i = \overline{1, n-1}) \\ A_n y_n + B_n y_{n-1} = D_n \end{cases} \quad (6.21)$$

Ma'lumki, qidirilayotgan taqribiy yechimning aniqlik darajasini oshirish uchun $[a, b]$ oraliqda kiritilgan $x_i = a + ih$ to'ring h qadamini kichraytirish lozim. Bu miqdorni kichraytirish esa o'z navbatida tugun nuqtalar x_i ning sonini keskin oshishiga olib keladi. Shunday qilib, qo'yilgan masalani zarur aniqlikda yechish uchun hosil qilingan (6.21) sistemaning tartibi ming, ayrim hollarda esa o'n mingdan ham ortiq bo'lishi mumkin. Yuqorida eslatganimizdek, sistemaning har bir tenglamasida faqat uchta tangina noma'lum qatnashgan xadlar mavjud. Qolgan noma'lumlarning koeffisientlari esa nolga teng. Agarda biz bunday sistemani an'anaviy usullar (Gauss, Kramer, teskari matrisa kabi) yordamida yechmoqchi bo'lsak, nollar ustida ma'nosiz bo'lgan ko'p hajmdagi amallarni bajarishimizga to'g'ri keladi. Shuning uchun, bunday maxsus sistemalarni yechishning maxsus usullari ishlab chiqilgan. Bu usullarning eng soddasi, dasturlashga qulayi, xatolar yig'ilmasini hosil qilmaydigani “haydash” usuli hisoblanadi.

Quyida “Haydash” usulining qisqacha mohiyati bilan tanishib chiqamiz.

Maxsus, dioganalli sistemalarni yechishga mo'ljallangan “Haydash” usuli ikki bosqichdan iborat:

noma'lum koeffisientlarni aniqlash (to'g'ri bosqichi)

sistemaning yechimlarini aniqlash (teskari bosqichi).

1-bosqichda (6.21) sistemaning noma'lum y_i yechimini quyidagi ko'rinishda qidiramiz:

$$y_i = \alpha_{i+1}y_{i+1} + \beta_{i+1} \quad (6.22)$$

bu yerda α_{i+1} va β_{i+1} noma'lum haydash koefitsientlari. Noma'lum $\alpha_{i+1}, \beta_{i+1}$ koefitsientlarni topish uchun (6.22) tenglikni $x = x_i$ va $x = x_{i-1}$ nuqtalardagi ko'rinishini (6.21) formuladagi ikkinchi tenglamaga ketma-ket qo'yib,

$$A_i y_{i+1} - B_i(\alpha_{i+1}y_{i+1} + \beta_{i+1}) + C_i(\alpha_i(\alpha_{i+1}y_{i+1} + \beta_{i+1}) + \beta_i) = D_i$$

yoki

$$(A_i - B_i\alpha_{i+1} + C_i\alpha_i\alpha_{i+1})y_{i+1} + (-B_i\beta_{i+1} + C_i\alpha_i\beta_{i+1} + C_i\beta_i - D_i) = 0$$

ni hosil qilamiz.

Bu chiziqli ifoda aynan 0 ga teng bo'lishi uchun, barcha koefitsientlar 0 ga teng bo'lishi kerakligini hisobga olib, quyidagi tengliklarni hosil qilamiz:

$$\begin{aligned} A_i - B_i\alpha_{i+1} + C_i\alpha_i\alpha_{i+1} &= 0 \\ -B_i\beta_{i+1} + C_i\alpha_i\beta_{i+1} + C_i\beta_i - D_i &= 0 \end{aligned}$$

Hosil qilingan tengliklardan $\alpha_{i+1}, \beta_{i+1}$ noma'lum koefitsientlarni topish unchalik qiyin emas, ya'ni

$$\alpha_{i+1} = \frac{A_i}{B_i - C_i\alpha_i}; \quad \beta_{i+1} = \frac{C_i\beta_i - D_i}{B_i - C_i\alpha_i}; \quad i = \overline{1, n-1} \quad (6.23)$$

Mazkur rekkurent formuladagi barcha α_{i+1} va β_{i+1} larni aniqlash uchun yoki boshqacha aytganda rekkurent formulani "yurishi" uchun dastlabki α_1 va β_1 qiymatlarni topishimiz kerak. Bu qiymatlarni topishimiz uchun $x = a$ nuqtadagi chegaraviy shartdan hosil qilingan (6.21) formuladagi birinchi tenglamadan foydalanamiz.

$B_0y_0 + C_0y_1 = D_0$ tenglamani har ikkala tomonini A_0 ga bo'lib, y_0 ni topamiz:

$$y_0 = \frac{C_0}{B_0}y_1 - \frac{D_0}{B_0};$$

Keltirib chiqarilgan formulani (6.22) formulaning $i=0$ dagi qiymatida hosil qilingan $y_0 = \alpha_1y_1 + \beta_1$ bilan solishtirish natijasida $\alpha_1 = \frac{C_0}{B_0}; \quad \beta_1 = -\frac{D_0}{B_0}$ ekanligi kelib chiqadi.

Eslatib o'tamiz, B_0, C_0, D_0 larning qiymati oldinroq (6.20) formulalar orqali aniqlangan edi.

α_1, β_1 lar ma'lum bo'lgach, barcha keyingi $\alpha_{i+1}, \beta_{i+1}$ lar (6.23) rekkurent formuladan topiladi. Bu jarayon "haydash" usulining to'g'ri bosqichini tashkil etadi.

2-bosqichda α_i, β_i noma'lum koeffisientlarning barcha qiymat-lari topilgach (6.22) rekkurent formula yordamida qidirilayotgan yechim y_i larni topish mumkin, bu yerda ham rekkurent formulaning ishlashi uchun dastlabki qiymat sifatida y_n ni aniqlash lozim. Bu ishni bajarish uchun $x=b$ nuqtadagi chegaraviy shartdan hosil qilingan (6.21) sistemaning uchinchi tenglamasi

$$A_n y_n + B_n y_{n-1} = D_n$$

va (6.22) formulaning $i=n-1$ nuqtadagi ko'rinishi $y_{n-1} = \alpha_n y_n + \beta_n$ dan foydalanamiz, ya'ni ularni sistema deb qarab, bu sistemadan y_n ni aniqlaymiz.

$$y_n = \frac{C_n \beta_n - D_n}{B_n - C_n \alpha_n}$$

Qidirilayotgan y_n hisoblangach, $y_i = \alpha_{i+1} \cdot y_{i+1} + \beta_{i+1}$ rekkurent formulasi yordamida ($i = \overline{n-1, 0}$) barcha qolgan yechimlar topiladi.

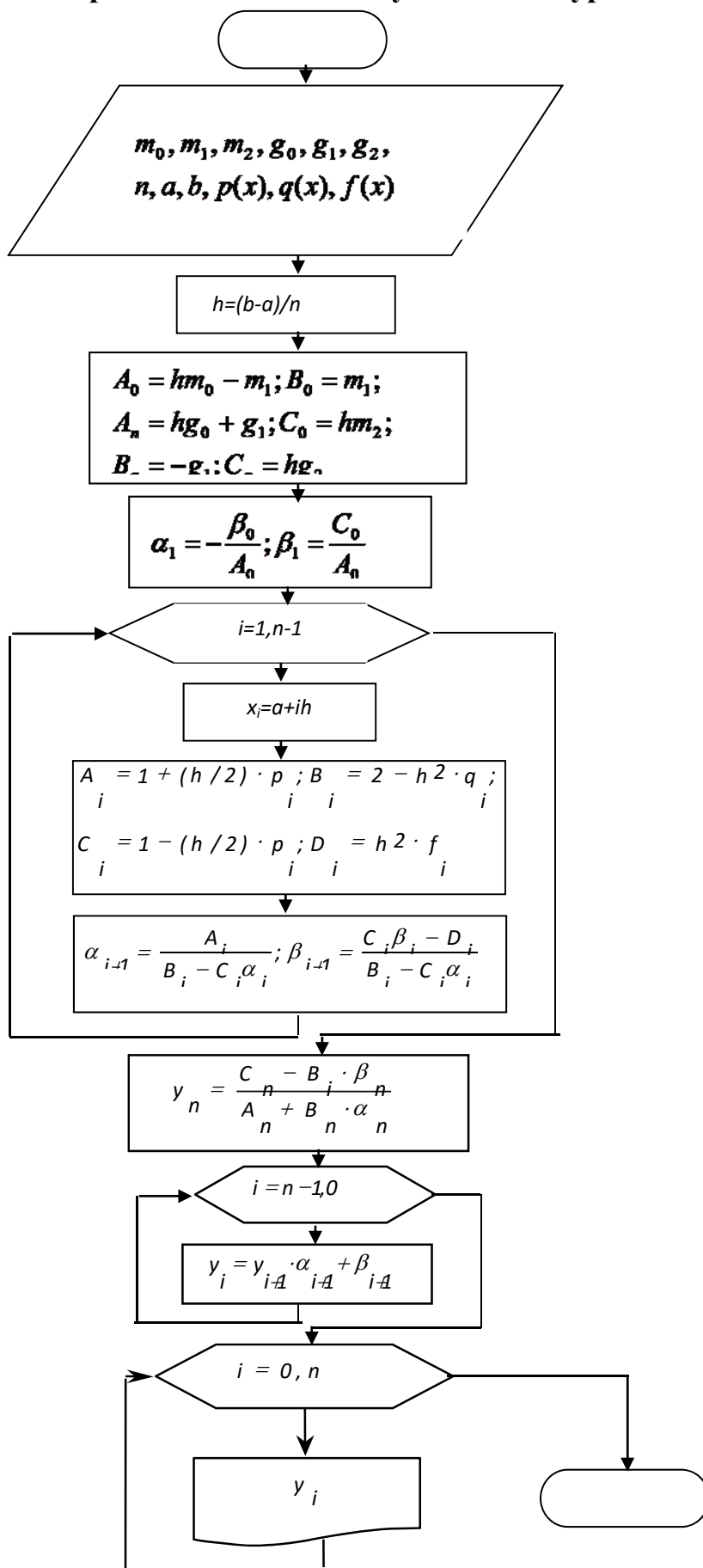
Bu jarayon i ga nisbatan teskari tartibda bo'lgani uchun, uni haydashning teskari bosqichi deb ataymiz.

(6.21) sistemaga xaydash usulini qo'llash uchun quyidagi turg'unlik shartlari bajarilishi kerak:

$$A_i \neq 0, C_i \neq 0, |B_i| \geq |A_i| + |C_i|, i = \overline{1, n-1}, \left| \frac{C_0}{B_0} \right| \leq 1, \left| \frac{C_n}{B_n} \right| < 1.$$

Shunday qilib, oldimizga qo'yilgan masalani, ya'ni o'zgaruvchan koeffisientli, ikkinchi tartibli, oddiy differensial tenglamani chekli ayirmali formulalar yordamida sonli-taqribiy usulda yechish uchun ishchi algoritm hosil qildik.

Usulga mos algoritm blok-sxemasi quyidagicha ko‘rinishda bo‘ladi
 Усулга мос алгоритм блок-схемаси куйидагича кўринишда бўлади



Algoritmning dastur matni

```
#program chekli_a;
import math
n=10
a0,b0,c0,an,bn,cn,ai,bi,ci,di=0,0,0,0,0,0,0,0,0,0
i=0
al,be=[0]*n,[0]*n
y=[0]*n
def p(x):
    p=-2
    return p
def f(x):
    f = 12*x*x-8*x*math.pow(x,2)+math.exp(7*math.log(x))
    return f
m0, m1, m2, g0, g1, g2= map(float, input("m0, m1, m2, g0, g1, g2 = ").split())
a,b = map(float, input("a,b = ").split())
h=(b-a)/n
b0=-h*m0+m1
a0=m1
d0=h*m2
bn=-h*g0-g1
cn=-g1
cn=h*g2
al[0]=c0/b0
be[0]=-d0/b0
for i in range(n-1):
    x=a+i*h
    ai=1+(h/2)*p(x)
    bi=2-h*h*f(x)
    ci=1-(h/2)*p(x)
    di=h*h*f(x)
    al[i+1]=ai/(bi-ci*al[i])
    be[i+1]=(ci*be[i]-di)/(bi-ci-al[i])
y[n-1]=(an-cn*be[n-1])/(-bn+cn*al[n-1])
for i in range(n-2, 0, -1):
    y[i]=y[i+1]*al[i+1]+be[i+1]
for i in range(n):
    print(f" {y[i]:2.8f}          {math.pow(h*i,2):2.8f}          {abs(y[i]-math.pow(i*h,
2)*math.pow(i*h,2))}")
```

Ishlab chiqilgan algoritm va dasturning ishonchlilik daraja-sini tekshirish uchun quyidagi chegaraviy masalani yechishni tashkil qilib ko‘raylik:

$$y''(x) - 2 \cdot y'(x) + x^3 \cdot y(x) = 12x^2 - 8x^3 + x^7$$

differentensial tenglamani $y(0) = 0$, $y(1) = 1$

chegaraviy shartni qanoatlantiruvchi yechimini topish.

Dasturdan olingan natijalar quyidagi jadvalda keltirilgan.

x	Taqribiy	Aniq	Xatolik
0.0	0.00000000	0.00000000	0.00000000
0.1	0.00006966	0.00010000	0.00016966
0.2	0.00108964	0.00160000	0.00051036
0.3	0.00712884	0.00810000	0.00097116
0.4	0.02411037	0.02560000	0.00148963
0.5	0.06051107	0.06250000	0.00198893
0.6	0.12722581	0.12960000	0.00237419
0.7	0.23757174	0.24010000	0.00252826
0.8	0.40729307	0.40960000	0.00230693
0.9	0.65456559	0.65610000	0.00153441
1.0	1.00000000	1.00000000	0.00000000

Natijalardan va xatolik miqdorini kam ekanligidan ishlab chiqilgan algoritmdan amaliy masalalarni yechishda keng foydalanish mumkin degan xulosa kelib chiqadi.



Nazorat savollari

1. Chekli ayirmali formulalar qanday hosil qilinadi?
2. Chekli ayirmalar usulida hosil qilinadigan mahsus tenglamalar sistemasi qanday yechiladi?
3. Haydash usulidagi to‘g‘ri va teskari bosqichlarning mohiyatini tushuntirib bering?
4. Haydash usulini qo‘llash uchun qanday turg‘unlik shartlari bajarilishi kerak?

6-§. Galyorkin usuli



Tayanch soʻz va atamalar

Galyorkin usuli, izlanayotgan funksiya, tafovut miqdor, bir jinsli boʻlmagan chegaraviy shartlar, oʻzaro bogʻliq boʻlmagan (ortogonal) bazis funksiyalar.

Yuqorida taʼkidlab oʻtganimizdek. Galyorkin usuli taqribiy-analitik usullar guruhiga kiradi. Mazkur usullar guruhi tarkibiga kiruvchi barcha usullarda berilgan differensial tenglamaning yechimi taqribiy aniqlangan formulalar yordamida topiladi va albatta, yechim ham analitik koʻrinishda ifodalanadi. Ayniqsa, koʻpgina fizika va mexanika masalalarining yechimini analitik koʻrinishda qidirish lozimligi, taqribiy analitik usullarni oʻrganishga katta ehtiyoj tugʻdiradi.

Deyarli barcha taqribiy-analitik usullarning algoritmlari bir-biriga oʻxshash boʻlgani uchun quyida Galyorkin usulini oʻrganish bilan cheklanamiz.

Bizga yana avvalgi mavzudagi kabi, quyidagi chegaraviy masala berilgan boʻlsin, yaʼni:

$$y''(x) + p(x) \cdot y'(x) + q(x) \cdot y(x) = f(x) \quad (6.24)$$

tenglama va

$$\begin{cases} m_0 y(a) + m_1 y'(a) = m_2 \\ g_0 y(b) + g_1 y'(b) = g_2 \end{cases} \quad (6.25)$$

chegaraviy shartni qanoatlantiruvchi yechimni topish kerak.

Maʼlumki, ixtiyoriy uzluksiz funksiyaning cheksiz qator koʻrinishida ifodalash mumkin:

$$y(x) = u_0(x) + \sum_{i=1}^{\infty} c_i u_i(x)$$

Galyorkin usulida ushbu qatordagi “ n ” ta chekli xad bilan chegaralanib, chegaraviy masalaning yechimini quyidagi ko‘rinishda qidirish taklif etiladi:

$$y(x) = u_0(x) + \sum_{i=1}^n c_i u_i(x) \quad (6.26)$$

Bu yerda shuni eslatib o‘tish lozimki, Galyorkin usulida yo‘l qo‘yilgan yagona va asosiy xatolik cheksiz hadli qatorni chekli hadli qatorga almashtirishdan iboratdir. Qatordagi hadlar sonini qancha ko‘p olsak, shunchalik olingan natijalar ishonchli va aniq yechimga yaqin bo‘ladi. Lekin, ikkinchi tomondan, qatordan ko‘proq had olishga intilish qo‘lda bajariladigan matematik almashtirishlar va amallar sonini keskin orttirib yuboradi. Bu esa yo‘l qo‘yilishi mumkin bo‘lgan xatoliklar ehtimolini keskin orttiradi.

Endi e‘tiborimizni yana yechimni qidirishga qaratsak, (6.26) formuladagi $c_1, c_2, \dots, c_n$ -lar qiymatlari noma‘lum bo‘lgan o‘zgarmaslar hisoblanadi. $u_0(x), u_1(x), \dots, u_n(x)$ lar esa tanlab olinadigan $[a, b]$ kesmada ikki marta uzluksiz differensallanuvchi, chiziqli bog‘liq bo‘lmagan funksiyalar hisoblanadi, ya‘ni ular bazis sistemasini tashkil qilishi kerak.

Bazis funksiyalarni shunday tanlash lozimki, (6.26) formula bilan aniqlanuvchi masalaning yechimi $c_1, c_2, \dots, c_n$ o‘zgarmaslarning ixtiyoriy tanlangan qiymatlarida ham chegaraviy masalaning (6.25) chegaraviy shartlarini qanoatlantirsin. Buning uchun bazis funksiyalarni tanlash quyidagicha amalga oshiriladi.

Avval quyidagi operatorlarni muomalaga kiritaylik:

$$L[y(x)] \equiv y''(x) + p(x)y'(x) + q(x)y(x),$$

$$L_a[y(x)] \equiv m_0 y(x) + m_1 y'(x)$$

$$L_b[y(x)] \equiv g_0 y(x) + g_1 y'(x)$$

1) $u_0(x)$ funksiya – berilgan (6.25) chegaraviy shartni qanoatlantiruvchi funksiya bo‘lishi lozim, ya‘ni:

$$\begin{cases} m_0 u_0(a) + m_1 u_0'(a) = m_2 \\ g_0 u_0(b) + g_1 u_0'(b) = g_2 \end{cases}, \quad \text{yoki} \quad \begin{cases} L_a[u_0(a)] = m_2 \\ L_b[u_0(b)] = g_2 \end{cases}.$$

2) $u_1(x), u_2(x), \dots, u_n(x)$ funksiyalari esa berilgan (6.25) chegaraviy shartning bir jinsli holatini qanoatlantiruvchi funksiyalar bo‘lishi lozim, ya‘ni:

$$\begin{cases} m_0 u_i(a) + m_1 u_i'(a) = 0 \\ g_0 u_i(b) + g_1 u_i'(b) = 0 \end{cases}, \quad i = \overline{1, n} \quad \text{yoki} \quad \begin{cases} L_a[u_i(a)] = 0 \\ L_b[u_i(b)] = 0 \end{cases}$$

Bazis funksiyalarni tanlash yo‘llarini quyidagi misolda ko‘rib chiqaylik: Chegaraviy masalaning chegaraviy shartlari quyidagicha berilgan bo‘lsin:

$$\begin{cases} y(0) = 1 \\ y(1) = 0 \end{cases}$$

$u_0(x)$ ni shunday tanlaymizki, $u_0(0) = 1$, $u_0(1) = 0$, ya‘ni berilgan chegaraviy shart qanoatlansin:

$$u_0(x) = 1 - x$$

Xuddi shunga o‘xshash, boshqa bazis funksiyalar $u_k(x)$ lar esa bir jinsli chegaraviy shartlarni qanoatlantirishi va chiziqli bog‘liqsiz bo‘lishi kerak.

$$\begin{cases} u_1(0) = 0 \\ u_1(1) = 0 \end{cases} \Rightarrow u_1(x) = x(1 - x)$$

$$\begin{cases} u_2(0) = 0 \\ u_2(1) = 0 \end{cases} \Rightarrow u_2(x) = x^2(1 - x)$$

Topshiriq: Chegaraviy shartlar $y(0)=1$ va $y'(\pi/2)=-1$ bo'lgan hol uchun bazis funksiyalarni mustaqil tanlang.

Yuqorida, soddalik uchun (6.26) da $n=2$ deb hisoblandi.

Biz $u_0(x), u_1(x), u_2(x), \dots, u_n(x)$ bazis funksiyalarni tanlashni o'rgandik, endi chegaraviy masalaning yechimi faqat $c_1, c_2, \dots, c_n$ noma'lum koefitsientlarga bog'liq bo'lib qoldi. Galyorkin, Rits, kollokasiya, eng kichik kvadratlar kabi boshqa taqribiy-analitik usullar bir-biri bilan aynan shu noma'lum koefitsientlarni aniqlash yo'llarini turlichaligi bilan o'zaro farq qiladi holos. Bu o'zgarmaslarni Galyorkin taklif etgan usul bilan aniqlashni tashkil qilamiz.

Buning uchun, dastlab (6.26) formulani (6.24) differensial tenglamaga qo'yib, quyidagi tafovut funksiyasini hosil qilamiz:

$$R(x, c_1, c_2, \dots, c_n) = L[u_0(x)] + \sum_{k=1}^n c_k \cdot L[u_k(x)] - f(x) \quad (6.27)$$

Bu funksiya chegaraviy masala taqribiy yechimining aniq yechimdan farqini xarakterlovchi miqdor bo'lib, u $c_1, c_2, \dots, c_n$ o'zgarmaslarga chiziqli bog'liqdir.

Tafovut funksiyani minimallashtirish sharti Galyorkin usulida quyidagicha ifodalanadi.

$$\begin{cases} \int_a^b R(x, c_1, c_2, \dots, c_n) \cdot u_1(x) \cdot dx = 0 \\ \int_a^b R(x, c_1, c_2, \dots, c_n) \cdot u_2(x) \cdot dx = 0 \\ \int_a^b R(x, c_1, c_2, \dots, c_n) \cdot u_n(x) \cdot dx = 0 \end{cases} \quad (6.28)$$

Yani, tafovut funksiyani $u_i(x)$, $(i=\overline{1, n})$ bazis funksiyalarga ortogonallik shartidan foydalaniladi.

(6.27) formuladagi R –tafovut funksiyasini (6.28) sistemaga qo‘yamiz.

$$\left\{ \begin{array}{l} \int_a^b (L[u_0] + c_1 \cdot L[u_1] + c_2 \cdot L[u_2] + \dots + c_n \cdot L[u_n] - f(x)) \cdot u_1(x) \cdot dx = 0 \\ \int_a^b (L[u_0] + c_1 \cdot L[u_1] + c_2 \cdot L[u_2] + \dots + c_n \cdot L[u_n] - f(x)) \cdot u_2(x) \cdot dx = 0 \\ \dots\dots\dots \\ \int_a^b (L[u_0] + c_1 \cdot L[u_1] + c_2 \cdot L[u_2] + \dots + c_n \cdot L[u_n] - f(x)) \cdot u_n(x) \cdot dx = 0 \end{array} \right. \quad (6.29)$$

Natijada $c_1, c_2, \dots, c_n$ noma'lumlarga nisbatan (6.29) ko'rinishidagi chiziqli algebraik tenglamalar sistemasi hosil bo'ladi. Uning koeffitsientlarini yuqoridagi aniq integrallarni hisoblash yordamida topiladi. Sistemani yechib (odatda Gauss usulidan foydalaniladi) $c_1, c_2, \dots, c_n$ noma'lum o'zgarmlarni aniqlash mumkin. U holda chegaraviy masalaning taqribiy analitik yechimini

$$y(x) = u_0(x) + c_1 \cdot u_1(x) + c_2 \cdot u_2(x) + \dots + c_n \cdot u_n(x)$$

ko‘rinishida yoza olamiz.

$n=3$ bo'lgan hol uchun tafovut funksiyasini va unga mos tenglamalar sistemasini hosil qiling.

Yuqorida ko‘rib, o‘rganib chiqilgan nazariy amallarni quyidagi chegaraviy masala ustida bajarishni tashkil qilaylik. ($n=2$ deb faraz qilaylik).

Chegaraviy masalaning differensial tenglamasi quyidagicha koʻrinishda berilgan boʻlsin:

$$y'' - 2y' + x^3 \cdot y = 12x^2 - 8x^3 + x^7$$

Differensial tenglamaning yechimiga qo'yilgan chegaraviy shartlar esa:

$$\begin{cases} y(0) = 0 \\ y(1) = 1 \end{cases}$$

Yuqorida ta'kidlaganimizdek, Galyorkin usuli bilan ishlashdan oldin berilgan masalaning chegaraviy shartlarini qanoatlantiradigan bazis funksiyalarni tanlab olishimiz lozim:

1) $u_0(x)$ ni berilgan chegaraviy shart, ya'ni $u_0(0)=0$ va $u_0(1)=1$ shartni qanoatlantiradigan qilib, quyidagicha tanlab olamiz: $u_0(x)=x$.

2) $u_1(x)$ va $u_2(x)$ larni esa berilgan chegaraviy shartga mos bir jinsli shartlarni, ya'ni $u_1(0)=0, u_1(1)=0$ va $u_2(0)=0, u_2(1)=0$ shartni qanoatlantiradigan va o'zaro chiziqli bog'liqsiz qilib, quyidagicha tanlab olamiz:

$$u_1(x) = x(x-1) = x^2 - x; \quad u_2(x) = x^2(x-1) = x^3 - x^2.$$

Ishchi formulalarda foydalaniladigan quyidagi operatorlarni hisoblashni tashkil qilaylik.

$$L[u_1] \equiv u_1'' - 2u_1' + x^3 u_1 = 2 - 2(2x-1) + x^3(x^2-x) = 2 - 4x + 2 + x^5 - x^4 = x^5 - x^4 - 4x + 4$$

$$L[u_2] \equiv u_2'' - 2u_2' + x^3 u_2 = 6x - 2 - 2(3x^2 - 2x) + x^3(x^3 - x^2) = 6x - 2 - 6x^2 + 4x + x^6 - x^5 = x^6 - x^5 - 6x^2 + 10x - 2.$$

$$L[u_0] \equiv u_0'' - 2u_0' + x^3 u_0 = 0 - 2 + x^4 = x^4 - 2$$

Endi quyidagi (6.29) sistemaga mos tenglamalar sistemasining koeffisientlari va ozod hadlarini hisoblashni tashkil etaylik.

$$\begin{cases} m_{11}c_1 + m_{12}c_2 = b_1 \\ m_{21}c_1 + m_{22}c_2 = b_2 \end{cases} \quad (6.30)$$

bu yerda

$$m_{11} = \int_0^1 L[u_1] \cdot u_1(x) dx = \int_0^1 (x^5 - x^4 - 4x + 4) \cdot (x^2 - x) dx;$$

$$m_{12} = \int_0^1 L[u_2] \cdot u_1(x) dx = \int_0^1 (x^6 - x^5 - 6x^2 + 10x - 2) \cdot (x^2 - x) dx ;$$

$$m_{21} = \int_0^1 L[u_1] \cdot u_2(x) \cdot dx = \int_0^1 (x^5 - x^4 - 4x + 4) \cdot (x^3 - x^2) \cdot dx ;$$

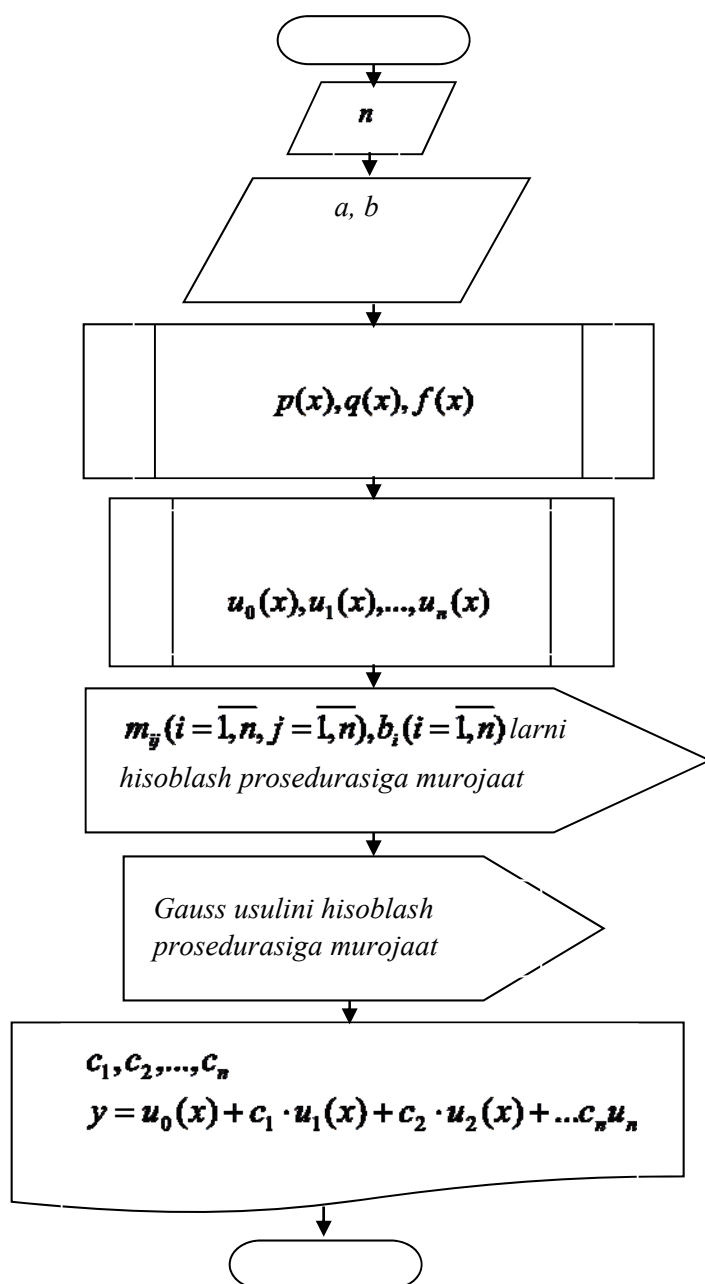
$$m_{22} = \int_0^1 L[u_2] u_2(x) dx = \int_0^1 (x^6 - x^5 - 6x^2 + 10x - 2) \cdot (x^3 - x^2) \cdot dx ;$$

$$b_1 = \int_0^1 (L[u_0] - f(x)) u_1(x) dx = \int_0^1 (12x^2 - 8x^3 + x^7 - x^2 + 2) \cdot (x^2 - x) dx;$$

$$b_2 = \int_0^1 (L[u_0] - f(x)) u_2(x) dx = \int_0^1 (12x^2 - 8x^3 + x^7 - x^2 + 2) \cdot (x^3 - x^2) dx;$$

Barcha koeffisientlarni integrallarni hisoblash orqali aniqlab olingach, (6.30) chiziqli algebraik tenglamalar sistemasini c_1 va c_2 noma'lumlarga nisbatan yechishni Gauss usuli bilan tashkil qilamiz. Hosil qilingan natijalarni, ya'ni c_1 va c_2 larning qiymatlarini $y = u_0(x) + c_1 \cdot u_1(x) + c_2 \cdot u_2(x)$ formulaga qo'yib, berilgan chegaraviy masalaning taqribiy-analitik yechimini hosil qilamiz.

Galyorkin usulining ishchi algoritmi uchun blok-sxema



Algoritmning dastur matni

```

# Program Galerkin;
import math
q=2
mas = [[0]*q]*q
mas1 = [0]*q
fx, lu0, lu1, lu2, a, b, z, x, h = 0,0,0,0,0,0,0,0,0
m = mas
c, m1 = mas1, mas1
i=0
def sqr(x):

```

```

    return x**2
def f(x, k):
    u0,u1,u2,u3=0,0,0,0
    lu0=sqr(x)-2
    u1=sqr(x)-x
    lu1=x**5-x**4-4*x+4
    lu2=x**6-x**5-6*sqr(x)+10*x-2
    fx = 12*x*x-8*x**3+x**7
    if k==1:
        f=lu1*u1
    elif k==2:
        f=lu2*u1
    elif k==3:
        f=lu1*u1*x
    elif k==4:
        f=lu2*u1*x
    elif k==5:
        f=(fx-lu0)*u1
    elif k==6:
        f=(fx-lu0)*u1*x
    return f
def integ(a,b,k):
    y,h1,i=0,0,0
    h1 = (b-a)/20
    y=(f(a,k)+f(b,k))/2
    print(f"{y:12.3f}")
    for i in range(19):
        y+=f(a-i*h1,k)
    y*=h1
    integ = y
    print(f"{y:12.3f}")
    return integ
def gauss(a, b, x, n):
    k,m,l,s=0,0,0,0
    for k in range(0, n-1):
        for m in range(k+1, n):
            for l in range(k+1, n):
                a[m][l]=a[m][l]-a[m][k]*a[k][l]/a[k][k]
                b[m]=b[m]-a[m][k]*b[k]/a[k][k]
    x[n-1]=b[n-1]/a[n-1][n-1]
    for k in range(n-1, 0, -1):
        s=0
        for i in range(k+1, n):
            s+=a[k][i]*x[i]

```

```

    x[k]=(b[k]-s)/a[k][k]
    return x
a,b=map(float, input("a,b = ").split())
m[0][0]=integ(a,b,1)
m[0][1]=integ(a,b,2)
m[1][0]=integ(a,b,3)
m[1][1]=integ(a,b,4)
m1[0]=integ(a,b,5)
m1[2]=integ(a,b,6)
c = gauss(m,m1,c,q)
for i in range(q):
    print(f"{c[i]}:12f.4")
for i in range(10):
    h=(b-a)/10
    x=a+i*2
    x=a+i*h
    z=x+c[0]*(x*x-x)+x[2]*(x*x*x-x*x)
    print(x, z, sqr(x)*sqr(x), abs(z-sqr(x)*sqr(x)))

```

Yuqoridagi misol uchun dastur ta`minotini ishlatib, olingan natijalar quyidagi jadvalda keltirilgan:

$c[1]= 0.7431$ $c[2]=1.9562$

x	taqribiy	aniq	xatolik
xq0.0	0.00000000	0.00000000	0.00000000
xq0.1	0.01551908	0.00010000	0.01541908
xq0.2	0.01851317	0.00160000	0.01691317
xq0.3	0.02071920	0.00810000	0.01261920
xq0.4	0.03387413	0.02560000	0.00827413
xq0.5	0.06971492	0.06250000	0.00721492
xq0.6	0.13997851	0.12960000	0.01037851
xq0.7	0.25640186	0.24010000	0.01630186
xq0.8	0.43072193	0.40960000	0.02112193
xq0.9	0.67467566	0.65610000	0.01857566
xq1.0	1.00000000	1.00000000	0.00000000

Natijalardan va xatolik miqdorini kam ekanligidan ishlab chiqilgan algoritmlardan amaliy masalalar yechishda foydalanish mumkin degan xulosa kelib chiqadi.



Nazorat savollari

1. Galyorkin usuli qanday usullar guruhiga kiradi? Nima uchun?
2. Galyorkin usulida bazis funksiyalar qanday tanlanadi?
3. Galyorkin usulida hosil bo'lgan tenglamalar sistemasi qaysi usulda yechiladi?
4. Galyorkin usulida aniqlikni oshirish imkoni bormi?

XULOSA

Ushbu bobda differensial tenglamaning asosiy sinflari, oddiy va xususiy hosilali differensial tenglamaning umumiy ta'rif keltirildi.

Oddiy differensial tenglamaning umumiy va xususiy yechimi tushunchasi bayon qilindi va yechish usullari guruhlar tahlil etildi.

Koshi masalasini yechishning Eyler va Runge-Kutta usullari uchun hisoblash algoritmlari va dastur ta'minotlar tavsiya qilindi.

Chegaraviy masalani yechishda qo'llaniladigan turli hil chekli ayirmali formulalar hosil qilinib, ular yordamida ikkinchi tartibli oddiy differensial tenglama sonli taqribiy usulda yechildi.

Chegaraviy masalani yechishda taqribiy-analitik usullar guruhiga kiruvchi Galyorkin usuli va unga mos ishchi algoritm-lar hamda dastur ta'minotlari keltirildi.

Barcha ishlab chiqilgan dastur ta'minotlari aniq masalar uchun qo'llanilib, olingan natijalar tahlil etildi. Pirovardida tavsiya qilingan hisoblash algoritmlarining ishonchliligi tasdiqlandi.



Bobga doir muammoli vaziyatlar!

- ✚ Matematik modellari oddiy differensial tenglamalar orqali ifodalanuvchi jarayonlarga misollar keltira olasizmi? Hosila bilan ishtirok etuvchi parametrlar amalda qanday qonuniyatlar orasida o'zgarishi mumkin?
- ✚ Differensial tenglamaning umumiy yechimini ifodalovchi funksiya bir nechta o'zgarmaslarga bog'liq bo'lsa ham amaliy jarayonning u yoki bu xususiyatini tavsiflashi mumkinmi?
- ✚ Eyler usuli foydalanish va o'zgarish uchun juda qulay bo'lganidan amalda keng qo'llaniladi degan mulohazalarga qo'shilasizmi?
- ✚ Chegaraviy masalalarga qo'shimcha shartlarning yetarli emas-ligini qanday oqibatlariga olib kelishi mumkinligini tushintira olasizmi?
- ✚ Chegaraviy masalani yechish uchun qaysi usullar guruhini qo'llagan maqsadga muvofiq deb o'ylaysiz?
- ✚ Chekli ayirmali usullarda qo'llaniladigan o'ng va chap chekli ayirmali formulalarning qaysi birida xatolik miqdori kamroq bo'lishi mumkin? Markaziy chekli ayirmali formulada xatolikning yanada kam bo'lishi ehtimoli mavjudmi? Fikringizni tushuntiring.
- ✚ "Haydash usuli"ning nomlanishi usulning mohiyatiga ko'ra qandaydir ma'noni anglatmasmi? Ya'ni bu usulda aynan nima "haydalishi" mumkin?
- ✚ Galyorkin usulida $n=3$ bo'lgan hol uchun tafovut funksiyasini va unga mos tenglamalar sistemasini hosil qila olasizmi? n ning katta bo'lgan qiymatlari orqali aniqlikni oshirish fikriga qo'shilasizmi?
- ✚ Galyorkin usulida izlanayotgan funksiya tarkibida ishtirok etuvchi koeffitsient funksiyalarning nima uchun aynan chiziqli bog'liqmas bo'lishi talab etiladi?

7-BOB. XUSUSIY HOSILALI DIFFERENSIAL TENGLAMALARNI TAQRIBIY YECHISH

Oliy matematika kursidan ma'lumki, agar differensial tenglamadagi noma'lum funksiya ikki yoki undan ortiq argumentlarga bog'liq bo'lsa, bunday differensial tenglamalarni xususiy hosilali differensial tenglamalar deb ataladi. Demak, bunday tenglamalarda funksiyaning erkli argumenti bo'yicha xususiy hosilalari qatnashadi. Juda ko'p amaliy jarayonlar xususiy hosilali differensial tenglamalar bilan ifodalangani bilan ularni umumiy holda yechish uchun aniq formula va qoidalar mavjud emas.

Shu bois, bunday tenglamalarni yechish algoritmlarini bilish, ularni yechishni tashkil etuvchi amaliy dastur ta'minotlarini yaratish masalasi davrimizning g'oyat muhim masalalaridan biri hisoblanadi. Ushbu bobda xususiy hosilali differensial tenglamalarning muayyan tiplarini yechish usullari, ularga mos ishchi algoritmlar va dastur ta'minotlari tavsiya qilinadi.

1-§. Xususiy hosilali differensial tenglamalar, ularning tiplari va yechish usullari



Tayanch so'z va atamalar

Xususiy hosilali differensial tenglama, xususiy hosilali differensial tenglamaning tipi, xususiy hosilali differensial tenglamani yechish usuli, aniq usullar, taqribiy- analitik usullar, sonli- taqribiy usullar.

Amalda xususiy hosilali differensial tenglamalar juda ko'p fizik jarayonlarni tahlil qilishda ishlatiladi. Masalan, turar joy binolari va korxonalar qurishdagi hisob ishlari, ko'p qavatli binolarning issiqlik rejimini saqlash maqsadida yechiladigan g'ovak to'siqlarning issiqlik o'tkazuvchanlik masalasi (bunda jism sirtiga o'tkaziladigan issiqlik ta'siri vaqt bo'yicha juda tez o'zgarishi va jism har xil materiallar aralashmasidan iborat bo'lishi mumkin), ingichka torlar, har xil materiallardan ishlangan tayoqlar va boshqa xildagi konstruksiyalarning ko'ndalang va bo'ylama tebranishlari jarayonlari, neft va gaz konlaridagi ishlab

chiqarishni tashkillashtirish va boshqarishni avtomatlashtirish maqsadida qaralayotgan qatlam parametrlarini aniqlik ko'rsatkichini yanada yaxshilash, quvurlardagi qovushqoq suyuqliklarning nostasionar harakati jarayonlari.

Bu jarayonlarning barchasi uchun yaratiladigan matematik modellar xususiy hosilali differensial tenglamalar orqali ifodalanadi.

Xususiy hosilali differensial tenglamalarni matematik-fizika tenglamalari deb ham ataladi. Oddiy differensial tenglamalar kabi xususiy hosilali differensial tenglamalar ham cheksiz ko'p yechimlarga ega. Ular umumiy yechimlar deyilib, xususiy yechimlar umumiy yechimlardan ma'lum shartlar asosida ajratiladi. Agar qo'shimcha shartlar soha chegarasida berilsa, bunday masalaga chegaraviy masala deyiladi. Agar chegaraviy shartlar berilmasdan faqat boshlang'ich shart berilsa, bunday masalaga xususiy hosilali differensial tenglama uchun Koshi masalasi deyiladi.

Bunda masala cheksiz sohada qaraladi. Masalada ham boshlang'ich, ham chegaraviy shartlar qatnashsa, bunday masalaga aralash masala deyiladi.

Xususiy hosilali differensial tenglamalarni ikki o'lchovli hol uchun quyidagicha yozish mumkin (qulaylik uchun faqat xususiy holni, ya'ni ikkinchi tartibli hosilalarga nisbatan chiziqli tenglamalarnigina qaraymiz):

$$au_{xx} + 2bu_{xy} + cu_{yy} + du_x + eu_y + fu = g \quad (7.1)$$

bunda x, y -erkli o'zgaruvchilar, $u(x, y)$ -qidirilayotgan noma'lum funksiya, indeksdagi x, y lar noma'lum funksiyaning x va y bo'yicha xususiy hosilalarini anglatadi. a, b, c, d, e, f, g -koeffisientlar umuman x, y va u ga bog'liq funksiyalar bo'lishi mumkin.

Agar ular o'zgarmas sonlardan iborat bo'lsa, (7.1) tenglamani o'zgarmas koeffisientli, x va y ga bog'liq funksiyalar bo'lsa o'zgaruvchi koeffisientli va nihoyat, x, y va u ga bog'liq funksiyalar bo'lsa, tenglama kvazichiziqli deyiladi. Bu funksiyalar berilgan ma'lum funksiyalar bo'lib, yopiq $\bar{G} = G + \Gamma$ sohada aniqlangandir. G soha x va y o'zgaruvchilarning o'zgarish sohasi bo'lib Γ kontur bilan chegaralangandir.

(7.1) ko‘rinishdagi matematik-fizika tenglamalarning tipi $D = b^2 - ac$ diskriminantning ishorasi bilan aniqlanadi. Agar $D > 0$ bo‘lsa, tenglama giperbolik tipga, $D = 0$ bo‘lsa, tenglama parabolik tipga, $D < 0$ bo‘lsa, tenglama elliptik tipga tegishli bo‘ladi. Tenglamaning tipini aniqlash juda muhim ahamiyatga ega, chunki bir xil tipdagi har xil tenglamalar juda ko‘p umumiy xususiyatlarga ega bo‘ladi. Xususiylar hosilali differensial tenglamalarni yechish usullari xuddi oddiy differensial tenglamalardagi kabi, bir necha guruhga bo‘linadi:

Aniq usullar;

Taqribiy-analitik usullar;

Sonli-taqribiy usullar;

Aniq usullar bilan asosan chiziqli xususiylar hosilali tenglamalar sodda ko‘rinishdagi chegaraviy va boshlang‘ich shartlar bilan berilganda yaxshi natijalar olish mumkin. Bu guruhga o‘zgaruvchilarni ajratish, Laplas almashtirishlari va boshqa usullar kiradi. Taqribiy-analitik usullar bilan umumiy ko‘rinishdagi tenglamalarni yechish imkoniyati deyarli yo‘q, faqat ayrim xususiylar hollardagina biror-bir natija chiqishi mumkin. Amalda esa foydalanishga qulayligi va dasturlashga osonligi uchun asosan sonli-taqribiy usullarni qo‘llaniladi.



Nazorat savollari

1. Qanday tenglamalar matematik-fizika tenglamalari deb ataladi?
2. Matematik-fizika tenglamalarini taqribiy hisoblash zaruriyati qайдan kelib chiqadi?
3. Xususiylar hosilali differensial tenglamani yechishning qanday usullarini bilasiz?
4. Xususiylar hosilali differensial tenglamani yechishda ko‘proq qaysi usuldan foydalaniladi?

2-§. Giperbolik tipdagi tenglamalarni yechish uchun to‘r usuli

Tayanch so‘z va atamalar



Tebranuvchi jarayonlar, giperbolik tipdagi tenglamalar, boshlang‘ich shart, Dirixle masalasi, Neyman masalasi, aralash masala, oshkor sxema, oshkormas sxema, algoritm blok-sxemasi, dastur ta‘minoti.

Yuqorida ta‘kidlab o‘tganimizdek, amalda uchraydigan barcha jarayonlar o‘zlarining asosiy xususiyatlarini ifodalovchi matematik modellarga egadirlar. Masalaning mohiyatiga qarab, bu modellarni ifodalovchi matematik tenglamalar turli ko‘rinishda, jumladan, murakkab jarayonlarning matematik modellari matematik-fizika tenglamalari orqali ifodalanadi.

Agar tebranuvchan xarakterdagi jarayonlar, aniqroq qilib aytadigan bo‘lsak, turli xil ingichka torlar, har xil materiallardan ishlangan tayoqlar va boshqa xildagi konstruksiyalarning ko‘ndalang va bo‘ylama tebranishlari jarayonlari o‘rganilayotgan bo‘lsa, bunday masalalarning matematik modellari giperbolik tipdagi tenglamalarga keltiriladi. Tebranishlar esa so‘nib boruvchi yoki aksincha bo‘lishi mumkin. Xususiyl holda giperbolik tipdagi tenglamalarni quyidagicha yozish mumkin (fazoviy koordinata bo‘yicha bir o‘lchov bilan chegaralanib):

$$\frac{\partial^2 u(x,t)}{\partial t^2} = c^2 \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t) \quad (7.2)$$

Bunda $u(x,t)$ -izlanuvchi funksiya, t -vaqt, x -chiziqli koordinata, c^2 -o‘zgarmas koeffisient. (7.2)-ko‘rinishdagi giperbolik tipdagi tenglamalar uchun odatda ikkita boshlang‘ich va ikkita chegaraviy shart beriladi. Qaralayotgan soha $x \in [a,b]$ va $t \in [0,T]$ lardan iborat bo‘lsa, qidirilayotgan noma‘lum $u(x,t)$ funksiya quyidagi boshlang‘ich shartlarni:

$$u(x,0) = f_1(x), \quad \frac{\partial u(x,0)}{\partial t} = f_2(x) \quad (7.3)$$

va quyidagicha chegaraviy shartlarni (soddalik uchun eng sodda chegaraviy shart, Dirixle masalasi qabul qilindi):

$$u(a,t) = \phi_1(t), \quad u(b,t) = \phi_2(t) \quad (7.4)$$

qanoatlantirishi kerak.

Umuman barcha tipdagi matematik-fizika tenglamalari uchun chegaraviy shartlar quyidagi ko‘rinishlarda qo‘yilishi mumkin:

1) **Dirixle masalasi:** $u(x, y)|_r = \gamma^{(0)}$

2) **Neyman masalasi:** $\frac{\partial u(x, y)}{\partial \vec{n}}|_r = \gamma^{(1)}$

3) **Aralash masala:** $\alpha u(x, y) + \beta \frac{\partial u(x, y)}{\partial \vec{n}}|_r = \gamma^{(2)}$

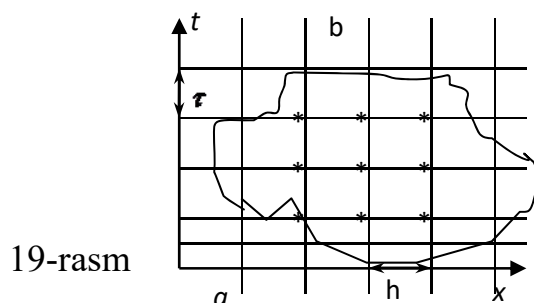
Bu yerda $u(x, y)$ -izlanayotgan funksiya; $\gamma^{(0)}, \gamma^{(1)}, \gamma^{(2)}$ -qiymatlari ma’lum funksiyalar yoki o‘zgarmaslar; Γ -yechim qidirilayotgan soha chegarasi; $\vec{n}$ -soha chegarasiga o‘tkazilgan normal birlik vektor; α, β -chegaraviy shart belgilari.

(7.2) ko‘rinishdagi xususiy hosilali differensial tenglamalarni yechish uchun sonli usullar ichida eng keng tarqalgan usul-cheqli ayirmalar usulidan foydalanamiz.

Tenglamada qatnashuvchi funksiya ikkita argumentga bog‘liq bo‘lgani uchun, uning aniqlanish sohasi tekislikda bo‘ladi. Bu sohani G -deb, uning chegarasini esa Γ -deb belgilaylik. Chekli ayirmalar usulida dastlab G -soha chiziqlar yordamida bo‘laklarga bo‘linadi. Bo‘linish nuqtalari tugun, ulardan tashkil topgan to‘plamga esa to‘r deb ataladi.

G -sohaning ichida yotgan nuqtalar ichki tugun nuqtalar, Γ -chegarada yotgan nuqtalarga chegaraviy nuqtalar deymiz (19-rasm).

To‘r sohani quyidagicha tashkil etamiz: $[a, b]$ kesmani $x_i = a + ih$ ($i = 0, 1, 2, \dots, n$) tugun nuqtalar yordamida, t -bo‘yicha esa $[0, T]$ oraliqni $t_j = j \cdot \tau$ bo‘laklarga bo‘lamiz va tekis to‘r hosil qilamiz. Bu yerda $h = \frac{b-a}{n}$, $\tau = \frac{T}{m}$ ga teng. Bunda vaqt bo‘yicha tanlangan qadam fazoviy koordinata x bo‘yicha tanlangan qadamdan kichik bo‘lishi shart. Aks holda, hosil qilingan sxemalar turg‘un bo‘lmaydi.



19-rasm

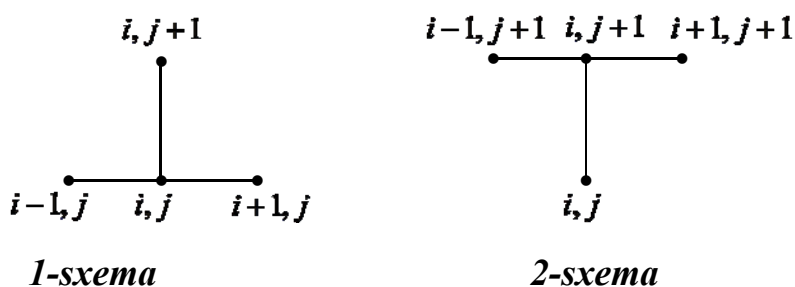
τ sohada yotgan (x_i, t_j) tugun nuqtalar uchun (7.2) tenglamani quyidagi ko'rinishda qayta yozamiz:

$$\frac{\partial^2 u(x_i, t_j)}{\partial t^2} = c^2 \frac{\partial^2 u(x_i, t_j)}{\partial x^2} + f(x_i, t_j) \quad (7.5)$$

To'r sohada to'r funksiyasi deb ataluvchi funksiyalarni qaraymiz. Ular τ sohada aniqlangan funksiyalar o'rnida qaraluvchi diskret funksiyalardan iborat bo'ladi, ya'ni oddiy differensial tenglamalardagi kabi xususiy hosilalar ham chekli ayirmalar bilan almashtiriladi.

Hosilalarni almashtirishdagi chekli ayirmalarda ishlatiluvchi tugun nuqtalar majmuasiga shablon deyiladi. Bir xil hosilalar uchun bir necha xil shablon asosida chekli ayirmalar tuzish mumkin. Shunday qilib, differensial tenglama berilgan boshlang'ich va chegaraviy shartlarda chekli ayirmali masalaga keltiriladi.

Biz hosil qilgan to'r sohadagi har bir tugun (x_i, t_j) nuqtalarda $u(x, t)$ yechimning qiymatlari $u(x_i, t_j)$ dan iborat bo'ladi. (7.5) tenglamani chekli ayirmalarga o'tkazish uchun (x_{i-1}, t_j) , (x_i, t_j) , (x_{i+1}, t_j) , (x_i, t_{j+1}) , (x_{i-1}, t_{j+1}) , (x_{i+1}, t_{j+1}) nuqtalardan tashkil topgan shablonlarni ishlatamiz. Bu shablonda (7.5) tenglamadagi xususiy hosilalarni quyidagi sxemalar yordamida chekli ayirmalar bilan almashtirish mumkin:



1-sxemada vaqt bo'yicha $j+1$ - qatlamning bitta nuqtadagi noma'lum yechimini j -qatlamdagi uchta tugun nuqtadagi ma'lum yechimlar orqali aniq, oshkor shaklda ifodalanadi. Shuning uchun, bunday sxemalarga oshkor sxemalar deyiladi.

Oshkor sxemalarda oldingi qatlamda yo'l qo'yilgan xatoliklar yig'indisi keyingi qatlamga ham o'tganligi uchun, bir necha qatlamdan so'ng xatoliklar to'planmasi hosil bo'ladi va ular olingan natijalarni butunlay yaroqsiz qilib qo'yishi mumkin. Shuning uchun, amalda oshkor sxemalardan faqat qisqa vaqt

oralig'ida yechiladigan masalalarni xal qilishdagina foydalangan ma'qul.

2-sxemada vaqt bo'yicha har bir yangi qatlamning uchta tugun nuqtadagi noma'lum yechimlari, o'zidan oldingi qatlamdagi bitta ma'lum yechim orqali ifodalanadi. Bu holda, har bir keyingi qatlamdagi yechimlarni odingi qatlamdagi yechimlar orqali bevosita, oshkor holda ifodalab bo'lmaydi. Bunday sxemalarga oshkormas sxemalar deyiladi.

Oshkormas sxemada biror qatlamda yo'l qo'yilgan hisoblash xatoliklari boshqa qatlamga deyarli uzatilmaydi. Shuning uchun, bunday sxemalar orqali hosil qilingan hisoblash formulalari birmuncha murakkab bo'lsa ham, lekin ulardagi xatolik miqdori kam bo'ladi. Demak, amalda oshkormas sxemalardan foydalangan

ma'qulroq. (7.5) tenglamadagi $\frac{\partial^2 u(x_i, t_j)}{\partial t^2}$; $\frac{\partial^2 u(x_i, t_j)}{\partial x^2}$ ifodalar o'rniga

$$\frac{\partial^2 u(x_i, t_j)}{\partial t^2} \approx \frac{u_i^{j+1} - 2u_i^j + u_i^{j-1}}{\tau^2} \quad \text{va} \quad \frac{\partial^2 u(x_i, t_j)}{\partial x^2} \approx \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{h^2}$$

chekli ayirmali formulalarni qo'yib, (7.5) tenglamalarga mos quyidagi chekli-ayirmali tenglamalarni hosil qilamiz:

$$\frac{u_i^{j+1} - 2u_i^j + u_i^{j-1}}{\tau^2} = c^2 \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{h^2} + f(x_i, t_j) \quad (7.6)$$

(7.6) tenglamani u_i^{j+1} ga nisbatan yechib,

$$u_i^{j+1} = 2u_i^j - u_i^{j-1} + \frac{\tau^2 c^2}{h^2} (u_{i+1}^j - 2u_i^j + u_{i-1}^j) + \tau^2 f_i^j \quad (7.7)$$

ishchi formulani hosil qilamiz. Bu yerda $j = \overline{1, m-1}$, $i = \overline{1, n-1}$.

(7.7) formuladan ko'rinib turibdiki, vaqtning $j+1$ -qatlamidagi yechimini topish uchun j va $j-1$ -qatlamlardagi yechimlardan foydalaniladi. Bu rekkurent formulaning ishlashi uchun dastlabki nolinch va birinchi qatlamlardagi yechim qiymatlarini berish kerak. Bu yechimlarni (7.3) boshlang'ich shart orqali hosil qilinadi. Nolinch qatlamda:

$$u_i^0 = f_1(x_i), \quad i = \overline{0, n} \quad (7.8)$$

Birinchi qatlamdagi yechim (7.3) dagi $f_1(x)$ va $f_2(x)$ lar orqali ifodalanadi:

$$\frac{\partial u(x_i, 0)}{\partial t} \approx \frac{u_i^1 - u_i^0}{\tau} \approx f_2(x_i)$$

va bulardan

$$u_i^1 \approx u_i^0 + \tau f_2(x_i) = f_1(x_i) + \tau f_2(x_i), \quad i = \overline{0, n} \quad (7.9)$$

ni hosil qilamiz.

Integrallash sohasining chegaralari $x=a$ va $x=b$ dagi yechimlar esa (7.4) chegaraviy sharti orqali aniqlanadi:

$$u_0^j = \phi_1^j \quad \text{va} \quad u_n^j = \phi_2^j, \quad j = \overline{0, m} \quad (7.10)$$

Shunday qilib, (7.7), (7.8), (7.9), (7.10) formulalarni birgalikda ishlatish orqali giperbolik tipli tenglamani yechim qidirilayotgan sohaning (x_i, t_j) tugun nuqtalaridagi sonli-taqribiy yechimlarini hosil qilinadi.

Hosil bo'lgan (7.7) formula oshkor sxema asosida olingan bo'lib, berilgan xususiy hosilali differensial tenglamaning taqribiy yechimini hisoblaydi. Yuqorida ta'kidlanganidek, oshkor sxemalarda hosil qilingan taqribiy yechim muayyan xatoliklar jamlanmasini o'zida saqlaydi. Yo'l qo'yilishi mumkin bo'lgan xatolikni birmuncha kamaytirish maqsadida oshkormas sxemali chekli-ayirmali almashtirishlardan foydalanamiz. Buning uchun (x_i, t_j) tugun nuqtalarda olingan fazoviy koordinatalar bo'yicha xususiy hosilalarni $\frac{\partial^2 u(x_i, t_j)}{\partial x^2} \approx \frac{u_{i+1}^{j+1} - 2u_i^{j+1} + u_{i-1}^{j+1}}{h^2}$ chekli-ayirmali formula orqali almashtirib, uni (7.5) tenglamaga qo'ysak, quyidagi ishchi tenglama hosil bo'ladi:

$$\frac{u_i^{j+1} - 2u_i^j + u_i^{j-1}}{\tau^2} = c^2 \frac{u_{i+1}^{j+1} - 2u_i^{j+1} + u_{i-1}^{j+1}}{h^2} + f(x_i, t_j) \quad (7.11)$$

(7.11) tenglamada o'xshash hadlarni ixchamlab,

$$\frac{\tau^2}{h^2} c^2 u_{i+1}^{j+1} - (2 \frac{c^2 \tau^2}{h^2} - 1) u_i^{j+1} + \frac{c^2 \tau^2}{h^2} u_{i-1}^{j+1} = -f_{ij} \tau^2 - 2u_i^j + u_i^{j-1} \quad (7.12)$$

va quyidagi belgilashlarni kiritib,

$$A_i = \frac{\tau^2}{h^2} c^2, B_i = \frac{2\tau^2 c^2}{h^2} - 1, C_i = \frac{c^2 \tau^2}{h^2} \quad \text{va} \quad F_i = -f_{ij} \tau^2 - 2u_i^j + u_i^{j-1};$$

vaqt faktorining har bir j -qiymati uchun ($j=1, 2, 3, \dots, m-1$) quyidagi uch diagonalli tenglamalar sistemasini hosil qilamiz:

$$A_i \cdot u_{i+1}^{j+1} - B_i \cdot u_i^{j+1} + C_i \cdot u_{i-1}^{j+1} = F_i \quad (7.13) \quad (i=1, 2, \dots, n-1)$$

Hosil bo'lgan (7.13) uch diagonalli tenglamalar sistemasi j ning har bir qiymatida $n-1$ ta tenglama va $n+1$ ta noma'lumlardan iborat. yetishmayotgan ikkita tenglamani (7.4) chegaraviy shartlardan olamiz. Natijada $n+1$ ta noma'lumli $n+1$ ta tenglamadan iborat bo'lgan uch diagonalli tenglamalar sistemasi hosil bo'ladi. Oldingi bobdagi chegaraviy masalalarni chekli-ayirmalar orqali yechishda ta'kidlab o'tilganidek, bunday ko'rinishdagi tenglamalar sistemasini haydash usuli bilan yechish uchun noma'lum yechimni

$$u_i^{j+1} = \alpha_{i+1} u_{i+1}^{j+1} + \beta_{i+1} \quad (7.14)$$

ko'rinishda qidiramiz. Bu yerdagi α_{i+1} va β_{i+1} noma'lum koeffisientlar

$$\alpha_{i+1} = \frac{A_i}{B_i} - C_i \cdot \alpha_i; \quad \beta_{i+1} = \frac{C_i - \beta_i - F_i}{B_i - C_i \cdot \alpha_i} \quad (7.15)$$

formulalar yordamida topiladi ($i = 1, 2, \dots, n-1$).

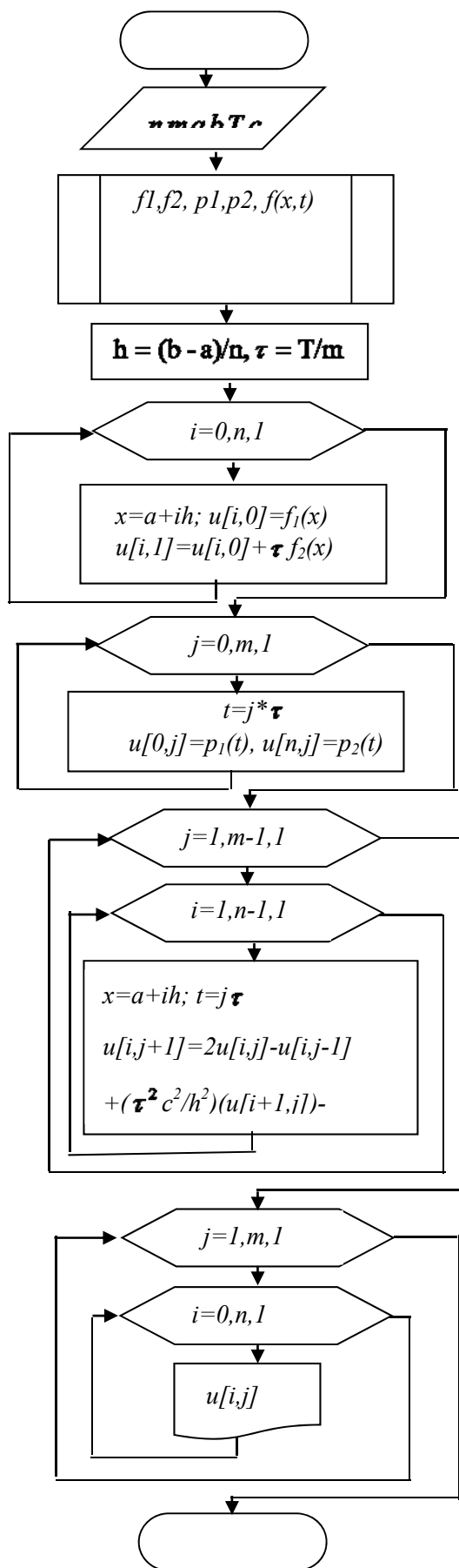
Berilgan chegaraviy shartlardan birinchisini, ya'ni $x = a$ nuqtadagi $u(a, t) = \phi_1(t)$ shartni va (7.14) formulani $i = 0$ nuqtada taqqoslab, $\alpha_1 = 0, \beta_1 = \phi_1(t)$ noma'lum koeffisientlarning boshlang'ich qiymatlari hosil qilinadi. So'ngra, $\alpha_{i+1}, \beta_{i+1}$ koeffisientlarning barchasi (7.15) formula bilan ketma-ket hisoblanadi.

Ikkinchi chegaraviy shartdan, ya'ni $u(b, t) = \phi_2(t)$ dan $u_n^j = \phi_2(t_j)$ tenglikni hosil qilamiz. So'ngra,

$$u_i^{j+1} = \alpha_{i+1} \cdot u_{i+1}^{j+1} + \beta_{i+1} \quad (i = n-1, \dots, 0) \quad (7.16)$$

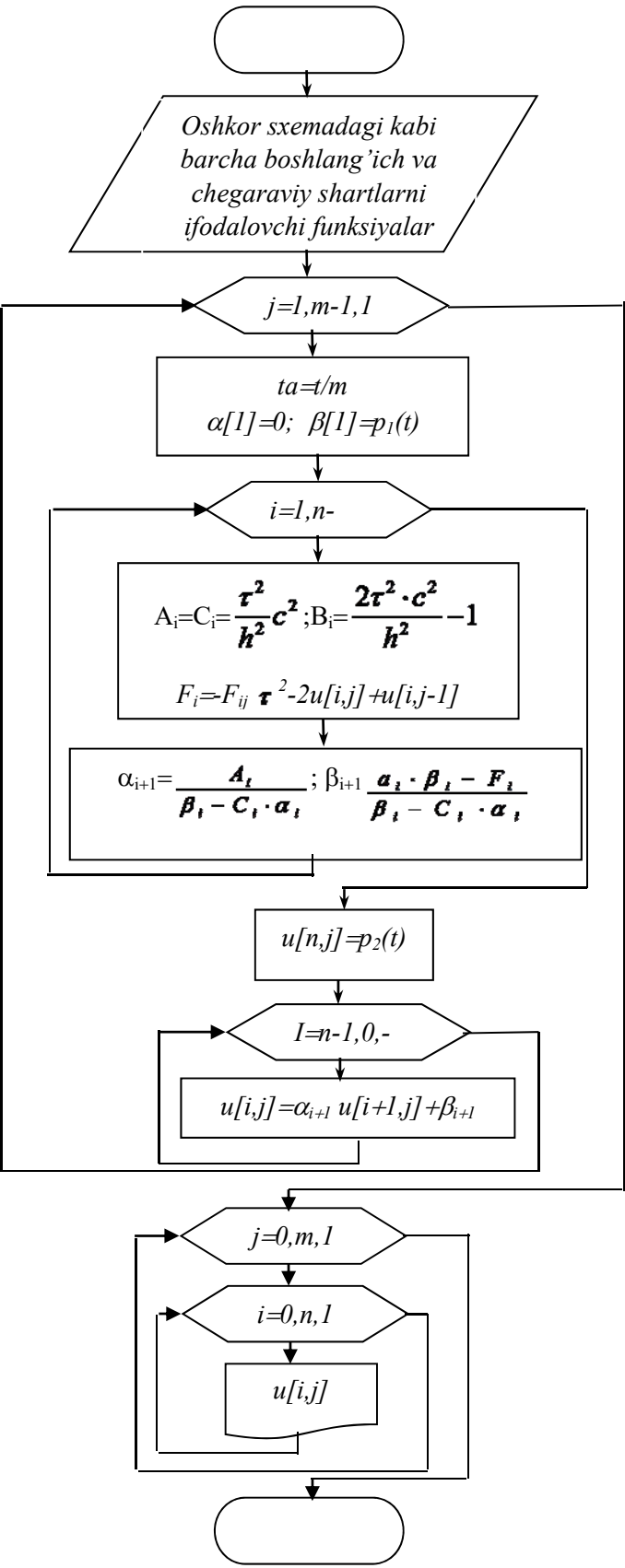
formula yordamida tenglamaning qolgan barcha qidirilayotgan yechimlari, ya'ni noma'lum u_i^j lar hisoblanadi. Yana shuni qayta eslatib o'tish lozimki, (7.13) ko'rinishdagi uch diagonalli tenglamalar sistemasi vaqt t ning har bir qiymati t_j uchun hosil qilinadi. (7.12) formulaning o'ng tomonidagi u_i^j va u_i^{j-1} larning boshlang'ich qiymatlari oshkor sxemali almashtirishlarda ishlatilgan (7.8) va (7.9) formulalar orqali hisoblanadi. Ishchi formulalarni hosil qilishda foydalanilgan chekli-ayirmalar-ning xatolik darajalari haqida oddiy differensial tenglamalarni yechishda to'liq ma'lumotlar keltirilgan.

Giperbolik tenglamani yechish oshkor sxema uchun algoritmnining blok-sxemasi va dasturi



Giperbolik tenglama uchun oshkormas sxemaning yechish dasturi

Giperbolik tenglamani yechishda oshkormas sxema uchun algoritmnining blok-sxemasi



Olingan natijalar tahlili.

Ishlab chiqilgan oshkor sxemali algoritm va dasturlarni ishga sozligini tekshirish uchun quyidagi giperbolik tipdagi tenglamani yechishni tashkil qilamiz:

$$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

Tenglamaning aniq yechimini $u(x,t) = e^{xt} + 3 \sin x - \cos t$ deb qabul qilsak, u holda $f(x,t)$ quyidagicha aniqlanadi:

$$f(x,t) = e^{xt}(t^2 - x^2) + \cos t - 3 \sin x$$

integrallash oralig'ini esa quyidagicha tanlab olamiz:

$$D: \{x \in [0,1]; \quad t \in [0,1]\}$$

U holda boshlang'ich shartlar: $u(x,0) = f_1(x) = 3 \sin x$, $\frac{\partial u(x,0)}{\partial t} = f_2(x) = x$

ko'rinishida, chegaraviy shartlar esa:

$$u(0,t) = P_1(t) = 1 - \cos t, \quad u(1,t) = P_2(t) = e^t + 3 \sin 1 - \cos t$$

ko'rinishida ifodalanadi. Hisob ishlari uchun $n=6$ va $m=10$ deb qabul qilindi.

Olingan natijalar quyidagi jadvalda ifoda qilingan.

Olingan natijalarni aniq yechim bilan taqqoslash shuni ko'rsatadiki, t ning ortib borishi bilan oshkor sxemani xatoligi ortib boradi.

Oshkor sxemali almashtirishlar uchun natijalar

Taqribiy yechim	Aniq yechim	Xatolik
$t=0.01$		
0.00000000	0.00005000	0.00005000
0.49935506	0.49940645	0.00005139
0.98491742	0.98497299	0.00005556
1.44327662	1.44333914	0.00006252
1.86177608	1.86184835	0.00007227
2.22886389	2.22894871	0.00008482
2.53441295	2.53451312	0.00010017

Taqribiy yechim	Aniq yechim	Xatolik
t= 0.05		
0.00124974	0.00124974	0.00000000
0.50705317	0.50730629	0.00025312
0.99936444	0.99964016	0.00027572
1.46453201	1.46484148	0.00030947
1.88989702	1.89025426	0.00035725
2.26391376	2.26432720	0.00041344
2.57693379	2.57693379	0.00000000
t= 0.10		
0.00499583	0.00499583	0.00000000
0.51900748	0.51949056	0.00048308
1.01992844	1.02047504	0.00054660
1.49393169	1.49454355	0.00061186
1.92834014	1.92904435	0.00070421
2.31165759	2.31243044	0.00077285
2.63457971	2.63457971	0.00000000

Oshkormas sxemali algoritm va dasturni ishga sozligini tekshirish uchun esa yana yuqoridagi kabi giperbolik tipdagi tenglamani yechishni tashkil qilamiz:

$$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

Tenglamaning aniq yechimini $u(x,t) = x^2 + t^2$ deb qabul qilsak, u holda $f(x,t)$ quyidagicha aniqlanadi: $f(x,t) = 0$

integrallash oralig'ini esa quyidagicha tanlab olamiz:

$$D : \{x \in [0,1]; \quad t \in [0,1]\}$$

U holda boshlang'ich shartlar: $u(x,0) = f_1(x) = x^2$, $\frac{\partial u(x,0)}{\partial t} = f_2(x) = 0$

ko'rinishida, chegaraviy shartlar esa: $u(0,t) = P_1(t) = t^2$, $u(1,t) = P_2(t) = 1 + t^2$

ko'rinishida ifodalanadi. Hisob ishlari uchun $n=10$ va $m=100$ deb qabul qilindi.

Olingan natijalar quyidagi jadvalda ifoda qilingan.

Taqribiy yechim	Aniq yechim	Xatolik
t=0.1		
0.00499583	0.00499583	0.00000000
0.31128044	0.31454625	0.00326581
0.61532565	0.62120517	0.00587952
0.91350050	0.92201099	0.00851049
1.20298495	1.21406164	0.01107668
1.48099529	1.49454355	0.01354826
1.74485925	1.76075980	0.01590055
1.99205193	2.01015708	0.01810515
2.22033581	2.24035118	0.02001537
2.42953079	2.44915085	0.01962006
2.63457971	2.63457971	0.00000000
t=0.5		
0.12241744	0.12241744	0.00000000
0.39720200	0.47318878	0.07598679
0.67706367	0.82359635	0.04653268
0.95544677	1.17081230	0.01536554
1.23140496	1.51207522	0.08067027
1.50750087	1.84471947	0.03721860
1.79466067	2.16620367	0.07154300
2.11217486	2.47413805	0.06196319
2.47208355	2.76631041	0.09422686
2.86794513	3.04071035	0.07276522
3.29555166	3.29555166	0.00000000

Taqribiy yechim	Aniq yechim	Xatolik
t=0.9		
0.45131014	0.45131014	0.00000000
0.62164789	0.85487669	0.03322880
0.83678892	1.26628053	0.02949160
1.10873329	1.68368606	0.07495277
1.44435035	2.10543448	0.06108413
1.83966320	2.53008499	0.09042179
2.29244547	2.95645638	0.06401091
2.80364532	3.38366886	0.0102355
3.37091108	3.81118604	0.04027496
3.99257441	4.23885687	0.04628245
4.66695757	4.66695757	0.00000000

Vaqtning o'zgarish qadamini kichikroq tanlash hisobiga taqribiy yechimning aniqligini yanada orttirish mumkin. Bundan tashqari, sinov sifatida tanlab olingan funksiyalar ham xatolik miqdoriga sezilarli ta'sir qilishi mumkin. Xatolik miqdorini unchalik katta emasligi ishlab chiqilgan algoritmlardan amaliy masalalarni yechishda foydalanish mumkinligini ko'rsatadi.

Nazorat savollari

1. Oshkor va oshkormas sxemalarning asosiy mohiyati nimadan iborat?
2. Giperbolik tipdagi tenglamalarni to'r usulida yechish algoritmini ayting?

3-§.Parabolik tipdagi tenglamalarni yechish uchun to‘r usuli



Tayanch so‘z va atamalar

Issiqlik o‘tkazuvchanlik masalasi, parabolik tipdagi tenglamalar, boshlang‘ich shart, chegaraviy shart, oshkor sxema, oshkormas sxema, algoritm blok-sxemasi, dasturi

Agar o‘rganilayotgan jarayonda vaqt bo‘yicha jarayonning kechish tezligi o‘zgarmas bo‘lsa, bu jarayonlarning matematik modeli parabolik tipdagi tenglamalar orqali ifodalanadi. Bunday jarayonlarga quvurlardagi qovushqoq suyuqliklarning nostasionar harakati jarayonlari, g‘ovak to‘siqlarning issiqlik o‘tkazuvchanlik masalalari, diffuziya jarayonlari va boshqalar kiradi.

Parabolik tipdagi tenglamalarni xususiy holda (fazoviy koordinata bo‘yicha bir o‘lchov bilan chegaralanib) quyidagicha yozish mumkin:

$$\frac{\partial u(x,t)}{\partial t} = c^2 \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t), \quad x \in [a,b] \quad t \in [0,T] \quad (7.17)$$

bu yerda $u(x,t)$ – izlanayotgan noma‘lum funksiya, $f(x,t)$ – masalaning fizik mohiyatidan kelib chiqib beriluvchi manbaa funksiya, c^2 – o‘zgarmas koeffisient. Bizga (7.17) tenglamani

$$u(x,0) = \phi(x) \quad (7.18)$$

boshlang‘ich shartni va $[a,b]$ oraliqning chetlarida

$$u(a,t) = \phi_1(t) \text{ va } u(b,t) = \phi_2(t) \quad (7.19)$$

chegaraviy shartlarni qanoatlantiruvchi yechimini topish masalasi qo‘yilgan bo‘lsin. Berilgan chegaraviy shartlar, biz yuqorida keltirgan chegaraviy shart turlariga ko‘ra, Dirixle masalasiga mos keladi. Xuddi giperbolik tipdagi kabi, parabolik tipdagi tenglamalarni ham to‘r usulida yechish mumkin. Buning uchun, dastlab, x va t o‘qlar bo‘yicha to‘r kiritamiz.

$$x_i = a + i \cdot h; \quad h = \frac{b-a}{n}; \quad t_j = j \cdot \tau; \quad \tau = \frac{T}{m};$$

bu yerda n -OX o‘qi bo‘yicha olingan tugun nuqtalar soni ;

h – x o‘zgaruvchi bo‘yicha kiritilgan to‘r qadami ;

T – qaralayotgan vaqt oralig‘i;

m – vaqt bo‘laklari soni;

τ – vaqt bo‘yicha kiritilgan to‘r qadami.

To‘r sohasidagi $\{x_i = a + i \cdot h, \quad h = \frac{b-a}{n}, \quad t_j = j \cdot \tau, \quad \tau = T / m\}$ har bir (x_i, t_j) nuqta

(7.17) tenglamani qanoatlantirgani uchun uni shu nuqtalarga nisbatan yozib olamiz.

$$\frac{\partial u(x_i, t_j)}{\partial t} = c^2 \frac{\partial^2 u(x_i, t_j)}{\partial x^2} + f(x_i, t_j) \quad (7.20)$$

(7.20) tenglamadagi $\frac{\partial u(x_i, t_j)}{\partial t}$ va $\frac{\partial^2 u(x_i, t_j)}{\partial x^2}$ xususiy hosilalar o‘rniga (qulaylik

uchun $u(x_i, t_j) = u_i^j$, $f(x_i, t_j) = f_{ij}$ deb belgilab olamiz) $\frac{\partial u_i^j}{\partial t} \approx \frac{u_i^{j+1} - u_i^j}{\tau}$,

$\frac{\partial^2 u_i^j}{\partial x^2} \approx \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{h^2}$ oshkor sxemali almashtirishlarga asoslangan chekli ayirmali

formulalarni qo‘yib, integrallash sohasining tugun nuqtalarida yozilgan quyidagi

$$\frac{u_i^{j+1} - u_i^j}{\tau} = c^2 \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{h^2} + f_{ij}$$

tenglamani hosil qilamiz. Uni u_i^{j+1} ga nisbatan yechib,

$$u_i^{j+1} = u_i^j + \frac{\tau \cdot c^2}{h^2} [u_{i+1}^j - u_i^j + u_{i-1}^j] + \tau \cdot f_{ij} \quad (7.21)$$

ko‘rinishidagi ishchi formulani hosil qilamiz.

Hosil qilingan (7.21) rekkurent formula bilan parabolik tipdagi tenglamaning (x_i, t_j) nuqtalardagi yechimlarini topish uchun unga chegaraviy va boshlang‘ich shartlarni ham qo‘shamiz:

$$u_i^0 = \phi_i, \quad i = \overline{0, n} \quad (7.22),$$

$$u_0^j = \phi_1^j, \quad u_n^j = \phi_2^j, \quad j = \overline{0, m} \quad (7.23)$$

(7.21) formuladan ko‘rinib turibdiki, vaqtning har bir $j+1$ - qatlamidagi noma‘lum yechimlar j -qatlam orqali topiladi va demak, u oshkor sxemali

almashtirish hisoblanadi. Oldingi paragrafda eslatib o'tganimizdek, bunday sxemali almashtirishlar natijasida barcha oldingi qatlamlarda yo'l qo'yilgan xatoliklar yig'ilib boradi.

Buning natijasida, vaqt faktorini oshib borishi bilan olinayotgan taqribiy yechimlarning ishonchlilik darajasi keskin kamayib boradi. Shuning uchun, topiladigan yechimning aniqliligini oshirish maqsadida oshkormas chekli ayirmali

sxemalarni ishlatish maqsadga muvofiqdir. Buning uchun (7.20) formuladagi $\frac{\partial^2 u_i^j}{\partial x_i^2}$

xususiy hosila o'rniga $\frac{\partial^2 u_i^j}{\partial x^2} \approx \frac{u_{i-1}^{j+1} - 2u_i^{j+1} + u_{i+1}^{j+1}}{h^2}$ chekli ayirmali formulani qo'yib,

$\frac{u_i^{j+1} - u_i^j}{\tau} = c^2 \cdot \frac{u_{i-1}^{j+1} - 2u_i^{j+1} + u_{i+1}^{j+1}}{h^2} + f_{ij}$ tenglamani hosil qilamiz.

O'xshash hadlarini ixchamlab, uni quyidagi ko'rinishda yozib olamiz:

$$\frac{c^2 \cdot \tau}{h^2} \cdot u_{i-1}^{j+1} - (2 \frac{c^2 \cdot \tau}{h^2} + 1) \cdot u_i^{j+1} + \frac{c^2 \cdot \tau}{h^2} \cdot u_{i+1}^{j+1} = -\tau \cdot f_{ij} - u_i^j \quad (7.24)$$

So'ngra (7.24) tenglamada

$$A_i = \frac{c^2 \cdot \tau}{h^2}; \quad B_i = \frac{2 \cdot c^2 \cdot \tau}{h^2} - 1; \quad C_i = \frac{c^2 \cdot \tau}{h^2}; \quad F_i = -\tau \cdot f_{ij} - u_i^j;$$

kabi belgilashlar kiritib, quyidagi

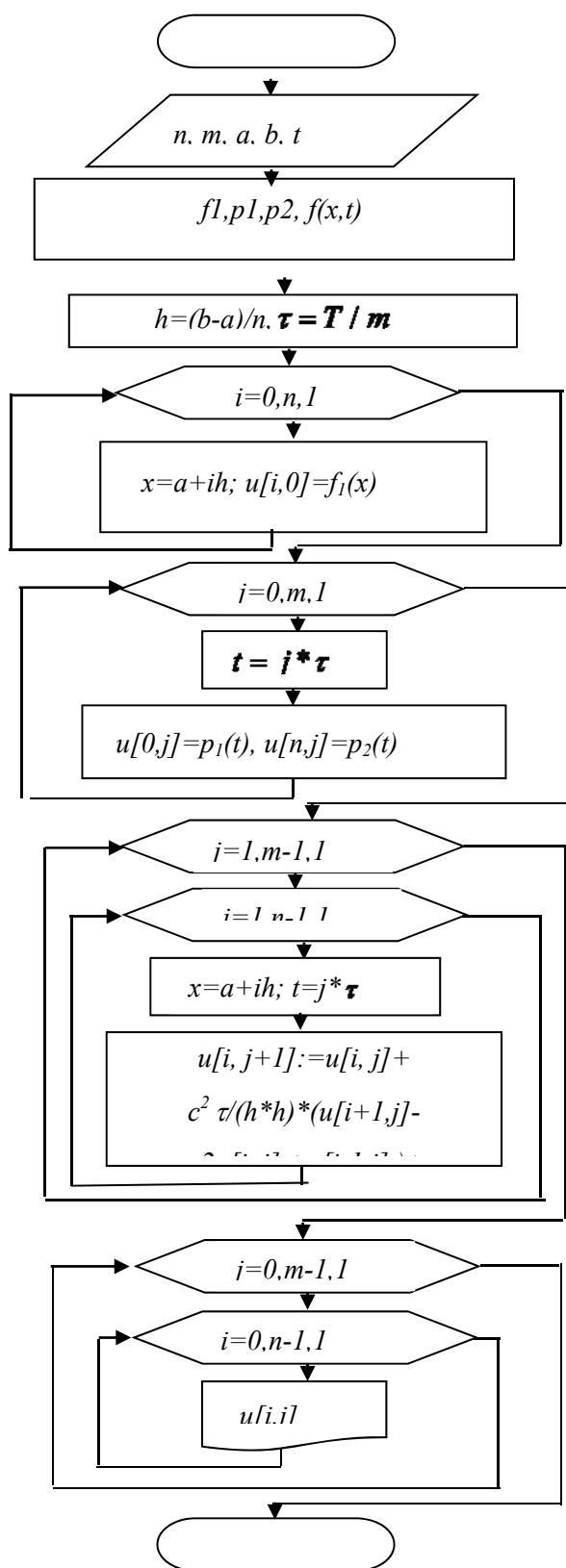
$$A_i \cdot u_{i-1}^{j+1} - B_i \cdot u_i^{j+1} + C_i \cdot u_{i+1}^{j+1} = F_i \quad (7.25)$$

uch diagonalli tenglamalar sistemasini hosil qilamiz.

Hosil bo'lgan (7.25) tenglamalar sistemasi j ning har bir qiymatida $n-1$ ta tenglama va $n+1$ ta noma'lumlardan iborat. yetishmayotgan ikkita tenglamani chegaraviy shartlardan olamiz.

Natijada $n+1$ ta noma'lumli $n+1$ ta tenglamadan iborat uch diagonalli tenglamalar sistemasi hosil bo'ladi. Bunday sistemani "haydash" usuli bilan yechish maqsadga muvofiq bo'lib, bu jarayon giperbolik tenglamalar uchun to'la ko'rsatildi. Yuqoridagi (7.24) sistemani ham shu tarzda yechib, barcha qidirilayotgan yechimlar aniqlanadi. Bu sxemada ham boshlang'ich va chegaraviy shartlar (7.22) va (7.23) kabi hisoblanib jarayonda ishtirok etadi.

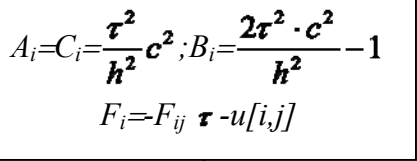
Parabolik tip tenglamalarni yechish algoritmiga mos blok-sxema va dasturi quyida berilgan.



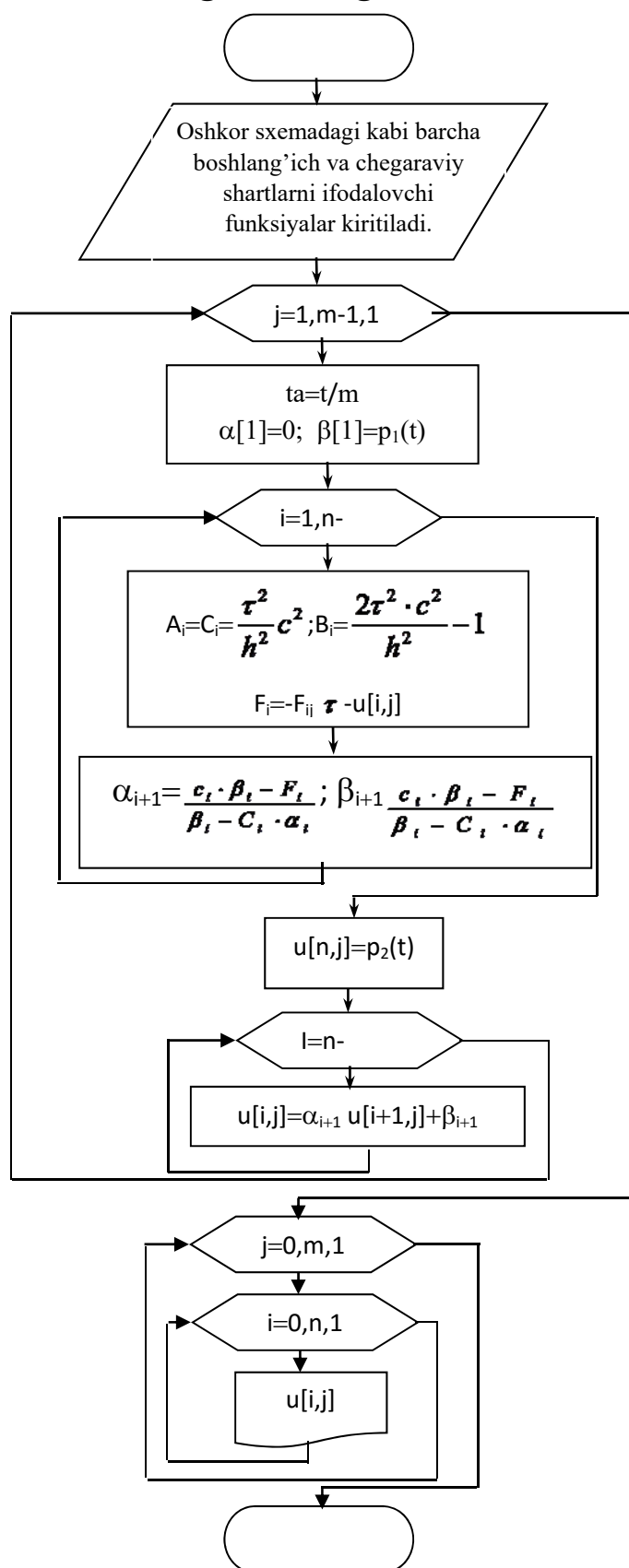
```

# Program Para_osh;
n=4; m=10;
a,b,h,ta,t,x=0,0,0,0,0,0
i,j=0,0
u = [[0]*m]*n
def sqr(x):
    return x*x
def f1(x):
    f1 = sqr(x)
    return f1
def p1(t):
    p1 = sqr(t)
    return p1
def p2(t):
    p2 = 1+sqr(t)
    return p2
def f(x,t):
    f = 2*t-2
    return f
a=0; b=1; t=1; h=(b-a)/n; ta=t/m;
for i in range(n):
    x=a+i*h
    u[0][j] = p1(t)
    u[n-1][j] = p2(t)
print("taqriban aniq xatolik")
for j in range(m-1):
    for i in range(n-1):
        x=a+i*h
        u[i][j] + (ta/(h*h))*(u[i+1][j]-
        2*u[i][j]+u[i-1][j])+ta*f(x,t)
    for j in range(m):
        t = j*ta
        print("t= ", t)
    for i in range(n):
        x=a+i*h
        print(u[i][j], sqr(x)+sqr(t), abs(u[i][j]-
        (sqr(x)+sqr(t))))
  
```

algoritmining blok-sxemasi



Parabolik tipdagi tenglamalarni oshkormas sxemalar yordamida yechish algoritmining blok-sxemasi



Parabolik tipdagi tenglamalarni oshkormas sxemalar algoritmiga mos dastur matni

```
# Program par_oshkormas;
import math
n=10; m=100;
kk,a,b,h,ta,t,x=0,0,0,0,0,0,0
i,j=0,0
ai,bi,ci,fi=0,0,0,0
u = [[0]*n]*m
al,be=[0]*n
def fl(x):
    fl = 1+x*x
    return fl
def pl(x):
    p1 = math.cos(x*2)
    return p1
def p2(x):
    p2 = math.cos(2*x)+math.exp(x)
    return p2
def f(x, t):
    f = -2*math.sin(2*t)+x*x
    return f
a=0; b=1; t=1; h=n=4; m=10;
a,b,h,ta,t,x=0,0,0,0,0,0
i,j=0,0
u = [[0]*m]*n
def sqr(x):
    return x*x
def fl(x):
    fl = sqr(x)
    return fl
def pl(t):
    p1 = sqr(t)
    return p1
def p2(t):
    p2 = 1+sqr(t)
    return p2
def f(x,t):
    f = 2*t-2
    return f
a=0; b=1; t=1; h=(b-a)/n; ta=t/m;
for i in range(n):
    x=a+i*h
    u[0][j] = p1(t)
    u[n-1][j] = p2(t)
    print("taqriban aniq xatolik")
    for j in range(m-1):
        for i in range(n-1):
            x=a+i*h
            u[i][j]+(ta/(h*h))*(u[i+1][j]-
            2*u[i][j]+u[i-1][j])+ta*f(x,t)
        for j in range(m):
            t = j*ta
            print("t= ", t)
            for i in range(n):
                x=a+i*h
                print(u[i][j], sqr(x)+sqr(t), abs(u[i][j]-
                (sqr(x)+sqr(t))))
```

Olingan natijalar tahlili.

Oshkor va oshkormas sxemalarga asoslanib ishlab chiqilgan algoritmlarni va dasturlarni ishga sozlash uchun quyidagi parabolik tipdagi tenglamani yechishni tashkil qilamiz:

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial u^2(x,t)}{\partial x^2} + f(x,t)$$

Oshkor sxemali almashtirishlar uchun tenglamaning aniq yechimi deb $u(x,t) = x^2 + t^2$ ni qabul qilsak, u holda tenglamadagi $f(x,t)$ quyidagicha aniqlanadi:

$$f(x,t) = u_t - u_{xx} = 2t - 2$$

integrallash oralig'ini esa quyidagicha deb qabul qilaylik:

$$D: \{x \in [0,1]; \quad t \in [0,1]\}$$

u holda boshlang'ich shart $u(x,0) = 1 + x^2$ chegaraviy shartlar esa $u(0,t) = t^2$, $u(1,t) = 1 + t^2$ ko'rinishda ifodalanadi. Hisob ishlari uchun $n=6$ va $m=100$ deb qabul qilingan. Olingan natijalar quyidagi jadvalda ifoda qilingan.

Taqribiy yechim	Aniq yechim	Xatolik
t=0.10		
0.01000000	0.01000000	0.00000000
0.07190289	0.07250000	0.00059711
0.25923136	0.26000000	0.00076864
0.57190289	0.57250000	0.00059711
1.01000000	1.01000000	0.00000000
t=0.50		
0.25000000	0.25000000	0.00000000
0.31156914	0.31250000	0.00093086
0.49875939	0.50000000	0.00124061
0.81156914	0.81250000	0.00093086
1.25000000	1.25000000	0.00000000
t=1.00		
1.00000000	1.00000000	0.00000000
1.06156255	1.06250000	0.00093745
1.24875007	1.25000000	0.00124993
1.56156255	1.56250000	0.00093745
2.00000000	2.00000000	0.00000000

Olingan natijalarni aniq yechim bilan taqqoslash ishlab chiqilgan algoritm va dasturning to'g'riligini ta'kidlab turibdi. Oshkor sxemali algoritm asosida olingan natijalar xatoligi kutilganidek vaqt ortishi bilan ko'payib bormoqda.

Oshkormas sxemali almashtirishlar uchun tenglamaning aniq yechimi deb $u(x,t)=\cos 2t+x^2et$ funksiyani qabul qilsak, u holda tenglamadagi $f(x,t)$ quyidagicha aniqlanadi:

$$f(x,t)=ut-uxx=-2sin2t+x^2et-2et$$

integrallash oralig'ini esa quyidagi deb qabul qilaylik:

$$D:\{x\in[0,1];t\in[0,1]\}$$

u holda boshlang'ich shart $u(x,0)=1+x^2$, chegaraviy shartlar esa $u(0,t)=\cos 2t$, $u(1,t)=\cos 2t+e^t$ ko'rinishda ifodalanadi. Hisob ishlari uchun $n=10$ va $m=100$ deb qabul qilingan. Olingan natijalar quyidagi jadvalda faqat ayrim qatlamlar uchun aks ettirilgan.

Taqribiy yechim	Aniq yechim	Xatolik
t=0.10		
0.980067	0.980067	0.000000
0.994909	0.991118	0.003790
1.028046	1.024273	0.003772
1.083244	1.079532	0.003712
1.160488	1.156894	0.003594
1.259757	1.256359	0.003398
1.381025	1.377928	0.003097
1.524255	1.521600	0.002655
1.689404	1.687376	0.002028
1.876418	1.875255	0.001163
2.085237	2.085237	0.000000
t=0.50		
0.540302	0.540302	0.000000
0.572980	0.556790	0.016191
0.621692	0.606251	0.015441
0.703151	0.688687	0.014463

0.817344	0.804098	0.013246
0.964256	0.952483	0.011774
1.143873	1.133842	0.010031
1.356176	1.348176	0.008001
1.601147	1.595484	0.005664
1.878769	1.875767	0.003002
2.189024	2.189024	0.000000
t=0.90		
0.227202	0.227202	0.000000
0.183072	0.202606	0.019534
0.109768	0.128818	0.019050
0.012261	0.005838	0.018098
0.183037	0.166334	0.016702
0.402581	0.387699	0.014882
0.670910	0.658255	0.012655
0.988042	0.978003	0.010038
1.353990	1.346944	0.007046
1.768771	1.765076	0.003694
2.232401	2.232401	0.000000

Olingan natijalarni aniq yechim bilan taqqoslash ishlab chiqilgan algoritm va dasturning to‘g‘riligini ta’kidlab turibdi. Bu holda ham oshkormas sxemali algoritm dasturi kutilgan natijalarni berdi. Xatolikning unchalik katta emasligi undan amaliy masalalar yechishda foydalanish mumkinligini ko‘rsatadi.



Nazorat savollari

1. Qanday jarayonlarning matematik modeli parabolik tipdagi tenglamalar orqali ifodalanadi?
2. Parabolik tipdagi tenglamalar uchun qanday qo‘shimcha shartlar beriladi?
3. Parabolik tipdagi tenglamani oshkor va oshkormas sxemalar orqali yechish qanday tashkil etiladi?

4-§. Elliptik tipdagi tenglamalarni yechish uchun to‘r usuli



Tayanch so‘z va atamalar

Stasionar jarayonlar, elliptik tipdagi tenglamalar, Puasson tenglamasi, Zeydel usuli, algoritm blok-sxemasi, dastur ta‘minoti.

Ma‘lumki, qaralayotgan masalada vaqt faktori kuchsiz rol o‘ynasa, ya‘ni jarayonning matematik modelida vaqtni ifodalovchi parametrlar qatnashmasa, bunday jarayonlarni stasionar jarayonlar deb ataladi. Stasionar jarayonlarga qurilish mexanikasini zo‘riqish va egilish masalalarini kiritish mumkin. Stasionar jarayonlarning matematik modellari elliptik tipdagi quyidagi tenglamalar orqali ifodalanishi mumkin (ikki o‘lchovli hol uchun):

$$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = f(x, y) \quad (7.26)$$

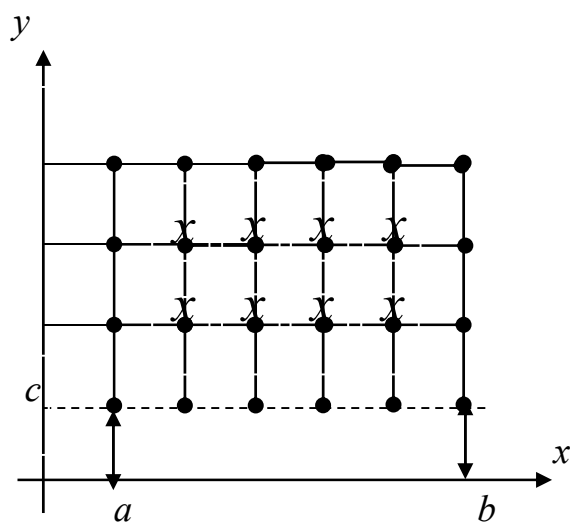
(7.26) tenglamani Puasson tenglamasi deb ataladi. Agar tenglamaning chap tomonida turgan miqdor $f(x, y)$ nolga teng bo‘lsa hosil bo‘lgan tenglamani

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad (7.27)$$

Laplas tenglamasi deb ataladi.

Ikki o‘lchovli (7.26) tenglamani yechimini ikki o‘lchamli G sohada aniqlangan deb faraz qilamiz. Soha chegarasi G yopiq chiziq bilan chegaralangan deb hisoblaymiz. Soddalik uchun integrallash oralig‘ini to‘g‘ri to‘rtburchak deb olamiz. Yuqorida keltirilgan Puasson tenglamasi uchun Dirixle masalasi berilgan to‘rtburchak soha uchun quyidagicha bo‘ladi:

$$\begin{aligned} u(x, y)|_{x=a} &= f_1(y) & u(x, y)|_{y=c} &= f_3(x) \\ u(x, y)|_{x=b} &= f_2(y) & u(x, y)|_{y=d} &= f_4(x) \end{aligned} \quad (7.28)$$



20-rasm

Bu yerda $f_1(x)$, $f_2(x)$, $f_3(x)$, $f_4(x)$ lar integrallash sohasining chegaralarida berilgan funksiyalar. Puasson tenglamasi uchun Dirixle masalasini to‘r usulida yechish uchun dastlab qaralayotgan OX va OY o‘qlar bo‘yicha $x_i=a+ih$, $y_j=c+j\tau$ tugun nuqtalar yordamida to‘r kiritamiz. 20-rasmda chegaraviy nuqtalar «•» belgisi bilan, ichki nuqtalar esa «x» belgisi bilan belgilangan.

Kiritilgan to‘rda OX va OY o‘qlari bo‘yicha to‘r qadamlari quyidagicha aniqlanadi:

$$h = \frac{b-a}{n}, \quad \tau = \frac{d-c}{m}$$

bu yerda n va m lar mos ravishda OX va OY o‘qlaridagi tugun nuqtalar soni.

Umuman olganda hisoblash algoritmini soddalashtirish maqsadida $h=l$ deb olish ham mumkin.

Kiritilgan tugun nuqta (x_i, y_j) larda Puasson tenglamasi (7.26) ni qayta yozamiz:

$$u_{xx}(x_i, y_j) + u_{yy}(x_i, y_j) = f(x_i, y_j) \quad (7.29)$$

(7.29) dagi ikkinchi tartibli xususiy hosilalar o‘rniga oldingi paragraflarda kiritilgan chekli-ayirmali formulalarni qo‘yib quyidagi ikki indeksli chekli ayirmali tenglamalarni :

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}^j}{h^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{l^2} = f(x_i, y_j) \quad (7.30)$$

chiziqli tenglamalar sistemasini hosil qilamiz. Sodda uchun (fizik ma'nosiga ko'ra) $h=l$ deb olib (7.30) tenglamani

$$u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j} = h^2 f_{i,j}, \quad i = \overline{1, n-1}, \quad j = \overline{1, m-1} \quad (7.31)$$

ko'rinishidagi chiziqli algebraik tenglamalar sistemasi ko'rinishida yozamiz.

Qidirilayotgan to'r funksiyaning G chegarasidagi qiymatlarini (7.28) formulada keltirilgan shart orqali aniqlanadi:

$$u_{0,j} = f_{1,j}, \quad u_{n,j} = f_{2,j}, \quad u_{i,0} = f_{3,i}, \quad u_{i,m} = f_{4,i} \quad i = \overline{0, n}, \quad j = \overline{0, m} \quad (7.32)$$

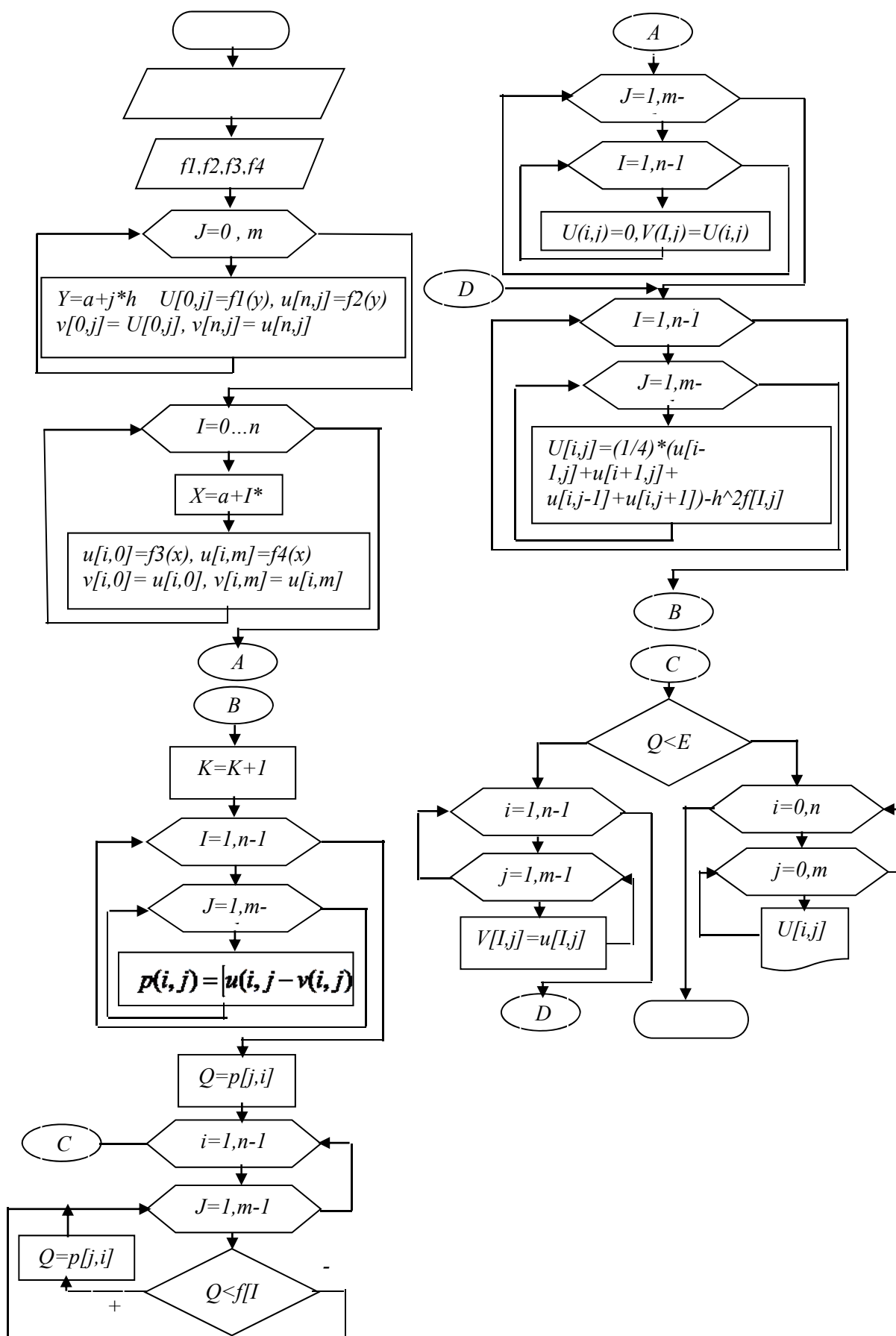
(7.31) tenglamalar sistemasini yechish uchun bizga oldingi boblardan ma'lum bo'lgan iteratsion usullardan foydalaniladi:

$$u_{i,j}^{(k+1)} = (u_{i+1,j}^{(k)} + u_{i-1,j}^{(k)} + u_{i,j+1}^{(k)} + u_{i,j-1}^{(k)} - h^2 f_{i,j}) / 4, \quad k = 0, 1, \dots, s$$

Elliptik tipdagi tenglamalarni to'r usulida yechish algoritmiga mos dastur

```
#program elliptik;
n=10; m=10; e=0.001;
p,u,v=[[0]*n]*m
k,i,j=0,0,0
a,b,c,d,q,h,x,y=0,0,0,0,0,0,0
def sqr(x):
    return x**2
def f1(y):
    return sqr(x)
def f2(y):
    f2 = 1+sqr(y)
    return f2
def f3(x):
    f3 = sqr(x)
    return f3
def f4(x):
    f4 = 1+sqr(x)
    return f4
def f(x):
    f = 0*x+0*y+4
    return f
h=0.1; a=0; b=1; c=0; d=1;
for j in range(m):
    y = c+j*h
    u[0][j] = f1(y)
    u[n-1][j] = f2(y)
    v[0][j] = u[0][j]
    v[n-1][j] = u[n-1][j]
for i in range(n):
    x = a+i*h
    u[i][0] = f3(x)
    u[i][m-1] = f4(x)
    v[i][0] = u[i][0]
    v[i][m-1] = u[i][m-1]
    for j in range(m-1):
        u[i][j] = 0
        v[i][j] = u[i][j]
    br = True
    while br:
        for i in range(n-1):
            x = a+i*h
            for j in range(m-1):
                y = c+j*h
                u[i][j] = (1/4)*(u[i-1,j]+v[i+1][j]+v[i][j-1]-
                h*h/4*f(x,y))
                p[i][j] = abs(u[i][j]-v[i][j])
            k+=1
        q=p[0][0]
        for i in range(n-1):
            for j in range(m-1):
                if q<p[i][j]:
                    q=p[i][j]
        for i in range(n):
            for j in range(m):
                v[i][j] = u[i][j]
        if q<e:
            br = False
    print("k= ", k)
    for i in range(n):
        if (i==1) or (i==5) or (i==10):
            for j in range(m):
                print("u[i][j] = ", u[i][j], " ",
                sqr(j*h)+sqr(j*h))
```

Elliptik tipdagi tenglamani yechish algoritmining blok-sxemasi



Olingan natijalar tahlili.

Ishlab chiqilgan algoritmni va dasturni ishga sozlash uchun quyidagi elliptik tipdagi tenglamani yechishni tashkil qilamiz:

$$\frac{\partial^2 u(x,y)}{\partial x^2} + \frac{\partial^2 u(x,y)}{\partial y^2} = f(x,y)$$

Tenglamaning aniq yechimi deb $u(x,y)=x^2+y^2$ ni qabul qilsak, u holda tenglamadagi $f(x,y)$ quyidagicha aniqlanadi:

$$f(x,y)=u_{yy}+u_{xx}=2+2=4$$

integrallash oralig'ini esa quyidagicha deb qabul qilaylik:

$$D: \{x \in [0,1]; y \in [0,1]\}$$

u holda chegaraviy shartlar $u(0,y)=y^2$, $u(n,y)=1+y^2$, $u(x,0)=x^2$, $u(x,n)=1+x^2$

ko'rinishda ifodalanadi. Hisob ishlari uchun $n=10$ deb qabul qilingan.

y=0.1	X	Taqribiy yechim	Aniq yechim	Aniqlik
	0.0	0.01000000	0.01000000	0.00000000
	0.1	0.01874920	0.02000000	0.00125080
	0.2	0.04773644	0.05000000	0.00226356
	0.3	0.09703570	0.10000000	0.00296430
	0.4	0.16668435	0.17000000	0.00331565
	0.5	0.25668298	0.26000000	0.00331702
	0.6	0.36699865	0.37000000	0.00300135
	0.7	0.49757117	0.50000000	0.00242883
	0.8	0.64832138	0.65000000	0.00167862
	0.9	0.81916060	0.82000000	0.00083940
	0.10	1.01000000	1.01000000	0.00000000
y=0.5				

	0.0	0.25000000	0.25000000	0.00000000
	0.1	0.25668298	0.26000000	0.00331702
	0.2	0.28399824	0.29000000	0.00600176
	0.3	0.33214165	0.34000000	0.00785835
	0.4	0.40121182	0.41000000	0.00878818
	0.5	0.49120965	0.50000000	0.00879035
	0.6	0.60204738	0.61000000	0.00795262
	0.7	0.73356516	0.74000000	0.00643484
	0.8	0.88555319	0.89000000	0.00444681
	0.9	1.05777650	1.06000000	0.00222350
	0.10	1.25000000	1.25000000	0.00000000

Olingan natijalarni aniq yechim bilan taqqoslash ishlab chiqilgan algoritm va dasturning to‘g‘riligini ta’kidlab turibdi. Xatolikning unchalik katta emasligi undan amaliy masalalar yechishda foydalanish mumkinligini ko‘rsatadi.

 Nazorat savollari

1. Eliptik tipdagi tenglamani umumiy holda ifodalay olasizmi?
2. Elliptik tipdagi tenglama uchun qanday qo‘shimcha shartlar beriladi?
3. Elliptik tipdagi tenglamani to‘r usulida yechishning mohiyatini tushuntiring.
4. Eliptik tipdagi tenglamani yechishda Zeydel usuli uchun boshlang‘ich qiymatlar qanday aniqlanadi?

XULOSA

Ushbu bobda xususiy hosilali differensial tenglamalar, ularning amaliy tadbiqlari, differensial tenglamani tiplarga ajratish, turli xil tipdagi differensial tenglamalar uchun beriladigan boshlang'ich va chegaraviy shartlar haqida zarur ma'lumotlar berildi.

Dirixle masalasi, Neyman masalasi, aralash masala, oshkor va oshkormas sxemalarning umumiy tavsifi keltirildi.

Giperbolik va parabolik tipdagi tenglamalarni oshkor va oshkormas sxemalar yordamida yechish uchun hisoblash algoritmlari va dastur ta'minotlari ishlab chiqildi.

Elliptik tipdagi tenglamani to'r usulida yechish algoritmi, uning dastur ta'minoti tavsiya qilindi.

Har bir tipdagi tenglamani yechish algoritmiga mos dastur ta'minotlari ishlatib ko'rildi va olingan natijalar tahlil etildi.



Bobga doir muammoli vaziyatlar!

- ✚ Nima uchun xususiy hosilali differensial tenglamalarni tiplarga ajratib o'rganiladi? Ularning barchasi uchun umumiy bo'lgan yechish usullarini ishlab chiqish mumkin emasmi?
- ✚ Matematik modellari xususiy hosilali differensial tenglamalar orqali ifodalanuvchi jarayonlarga misollar keltira olasizmi? Hosila bilan ishtirok etuvchi parametrlar amalda qanday qonuniyatlar asosida o'zgarishi mumkin?
- ✚ Xususiy hosilali differensial tenglamalarni yechishda olingan sonli-taqribiy yechimlarning aniqligini oshirish bo'yicha tavsiyalar bera olasizmi?
- ✚ Xususiy hosilali differensial tenglamalarni yechishda oshkor sxemali almashtirishlarni bajarish natijasida xatoliklar yig'indisi hosil qilinadi, u holda nima uchun oshkor sxemalar amalda qo'llanadi?

- ✚ Nima uchun elliptik tipdagi tenglamani yechishda aynan Zeydel usulidan foydalaniladi? Zeydel usulining iteratsiya usuliga ko'ra yechimga tez yaqinlashishiga nima sabab bo'lishi mumkin deb o'ylaysiz?
- ✚ Eliptik tipdagi tenlamalarda vaqt faktorining kuchli ta'sir etmasligi no'malum yechimlarni aniqlash jarayoniga ta'sir etishi mumkinmi?
- ✚ Oshkor va oshkormas shablonlarda differensialni approksimasiya qilish uchun ikkala holda ham chekli-ayirmali almashtirishlar qo'llanadi va ikki xil ishchi formula hosil qilinadi. Bunga sabab nimada?

Xulosa qilib aytganda, mustahkam egallangan bilim va malakalar natijasida talabalarda kelgusida amaliy masalalarni yechishda hisoblash usullarini qo'llash bo'yicha muayyan tajribalar hosil bo'lishi hamda ular ilmiy izlanishlarga va tadqiqotlarga yo'naltirishi mumkin. Samarali o'tkazilgan dars mashg'uloti esa undan ko'zlangan maqsadlarga to'laqonli erishilishini va talabalarda mustahkam bilimlar hosil bo'lishini kafolatlaydi.

Darslik fanlardan amaliy mashg'ulotlarni innovasion usullar yordamida loyihalashda hamda o'tkazishda muhim ahamiyatga egadir. Bundan tashqari fanni o'zlashtirish natijasida talabalar nafaqat tanlangan usullarga mos algoritmlar ishlab chiqish, dastur ta'minotlarini yaratish, balki tayyor matematik dasturiy tizimlardan foydalanib qo'yilgan masalalarni hal qilish imkoniyatiga ega bo'ladilar.

FOYDALANILGAN ADABIYOTLAR RO'YXATI

1. Abduqodirov A.A., Fozilov F.N., Umurzoqov T.N. Hisoblash matematikasi va programmalash. Toshkent, «O'qituvchi», 1989.
2. Badalov F.B, Shodmonov G'. Sh. Riyoziy modellar va muhandislik masalalarini sonli yechish usullari. Toshkent, «Fan», 2000.
3. Xolmatov T.X., Toyloqov N.Sh. Amaliy matematika va kompyu-terning dasturiy ta'minoti. Toshkent, «Mehnat», 2000.
4. Siddiqov A. Sonli usullar va programmalash. Toshkent, «O'zbekiston», 2001.
5. Боглаев Ю.П. Вычислительная математика и программирование. Москва, «Высшая школа», 1990.
6. Ho'jayorov B.X. Qurilish masalalarini sonli yechish usullari. Toshkent, «O'zbekiston», 1995.
7. Бахвалов Н.С. Жидков Н.Г., Кобелков Г.М. Численные методы. Москва, «Наука», 1987.
8. Самарский А.А., Гулин А.В. Численные методы. Москва, «Наука», 1989.
9. Boyzoqov A., Qayumov Sh. Hisoblash matematikasi asoslari. Toshkent, TDIU, 2000.
10. Isroilov M.I. Hisoblash metodlari. 1-qism. Toshkent, «O'zbekiston», 2003.
11. Qobulov V.K. Funktsional analiz va hisoblash matematikasi. Toshkent, «O'qituvchi», 1976.
12. Демидович Б.П., Марон И. А. Основы вычислительной математики. Москва, «Наука», 1970.
13. Калиткин Н.Н. Численные методы. Москва, «Наука», 1979.
14. Марчук Г. И. Методы вычислительной математики. Москва, «Наука», 1989.
15. Кабулов В.К., Файзуллаев А.Ф., Назиров Ш.А. Ал-Хорезми, алгоритм и алгоритмизация. Ташкент, «Фан», 2006.
16. Хюз Д., Митом Д. Структурный подход к программированию. Москва, «Мир», 1980.

17. Karimov P., Irisqulov S.S., Isaboev A. Dasturlash. Toshkent, «O‘zbekiston», 2003.
18. Абрамов В.Г., Трифанов Н.П., Трифанов Г.Н. Введение в язык Паскал. Москва, «Наука»,1989.
19. Поршнева С.В., Беленкова И.В. Численные методы на базе Mathcad. СПб, 2005.-464 с.
20. Ракитин В.И.Руководство по ВМ и приложения Mathcad. М.:ФМ,2005.- 264с.
21. Имомов А. Hisoblash usullari va Mathcad.Amaliy ishlarni bajarish na`munalari. Namangan, NamDU, 2013.-88 б.
22. Охорзин В.А. Прикладная математика в системе Mathcad. СПб, Лан,2008- 352 с.

MUNDARIJA

1-BOB. HISOBLASH TAJRIBASI VA XATOLIKLAR.....	6
1-§. Masalani kompyuterda yechish bosqichlari	7
2-§. Xatoliklar va ularni baholash.....	12
2-BOB. CHIZIQSIZ TENGLAMA VA TENGLAMALAR SISTEMASINI TAQRIBIY YECHISH.....	20
1-§. Umumiy tushunchalar	20
2-§. Oraliqni teng ikkiga bo‘lish (biseksiya) usuli.....	25
3-§. Urinmalar (Nyuton) usuli	28
4-§. Vatarlar usuli	32
5-§. Iteratsiya usuli	36
6-§. Chiziqsiz tenglamalar sistemasini yechishning oddiy iteratsiya usuli.....	44
7-§. Chiziqsiz tenglamalar sistemasini yechishning Nyuton usuli.....	51
3-BOB. CHIZIQLI ALGEBRAIK TENGLAMALAR SISTEMASINI YECHISH	59
1-§. Chizikli algebraik tenglamalar sistemasini yechishning umumiy qoidalari ...	59
2-§. Gauss usuli	62
3-§. Yuqori tartibli determinantlarni hisoblash	67
4-§. Teskari matrisani Gauss usuli bilan hisoblash.....	71
5-§. Iteratsiya usuli	74
6-§. Zeydel usuli.....	80
4-BOB. FUNKSIYALARNI INTERPOLYATSIYALASH.....	85
1-§. Interpolyatsiya masalasi	85
2-§. Nyuton interpolyasion formulasi.....	88
3-§ Lagranj interpolyasion formulasi	93
4-§. Eng kichik kvadratlar usuli	98

5-BOB. INTEGRALLARNI TAQIRIBIY HISOBLASH	104
1-§. Aniq integrallarni taqribiy hisoblash haqida umumiy tushunchalar.....	104
2-§. Tug'ri turtburchaklar formulasi.....	106
3-§. Trapetsiya usuli	108
4 -§. Simpson (parabolalar) formulasi	110
6-BOB. ODDIY DIFFERENSIAL TENGLAMALARNI SONLI YECHISH USULLARI.....	114
1-§. Differensial tenglamalar va ularni yechish usullari haqida umumiy ma'lumotlar.....	114
2-§. Koshi masalasini yechishning Eyler usuli.....	116
3-§. Koshi masalasini yechishning Runge-Kutta usuli	120
4-§. Chegaraviy masala va uni yechish usullari.....	123
5-§. Chekli ayirmalar usuli	126
6-§. Galyorkin usuli.....	136
7-BOB. XUSUSIY HOSILALI DIFFERENSIAL TENGLAMALARNI TAQIRIBIY YECHISH	148
1-§. Xususiy hosilali differensial tenglamalar, ularning tiplari va yechish usullari.....	148
2-§. Giperbolik tipdagi tenglamalarni yechish uchun to'r usuli	151
3-§.Parabolik tipdagi tenglamalarni yechish uchun to'r usuli	165
4-§. Elliptik tipdagi tenglamalarni yechish uchun to'r usuli	175
FOYDALANILGAN ADABIYOTLAR RO'YXATI	183

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

M.Olimov T.Juraev S.Abdusalilov

SONLI USULLAR VA ALGORITMLAR

*606102000-Axborot tizimlari va texnologiyalari ta'lim
yo'nalishi talabalari uchun*

Muharrir:

Dilorom ERGASHEVA

Texnik muharrir:

Abdulaziz TOSHPO'LATOV

2022-yil 10-noyabrda terishga berildi.

2022-yil 22-noyabrda bosishga ruxsat etildi.

Bichimi 60x84.1/16. Hajmi 11.5 bosma taboq.

Times New Roman garniturasida.

Buyurtma raqami – 7. Adadi 100 nusxa.

«MASHRAB» nashriyoti

Namangan shahri, Navoiy ko'chasi, 36-uy, 3-qavat

Nashriyot tasdiqnoma raqami 6135

2020 yil 16 dekabrda berilgan.

Murojaat uchun telefon: (+99899) 638-11-26

“Toshbuloq oqshomi” bosmaxonasida chop etildi.

Manzil: Namangan shahri, I. Karimov ko'chasi, 10-uy.

Murojaat uchun telefon: (+99891) 362-93-00