

## PARALLEL ISHLOV BERISH VA PARALLEL PROTSESSORLAR ARXITEKTURASI SOLISHTIRMA TAHLILI

*Saurova Kamola Erkinovna*

*Muxammad al-Xorazmiy nomidagi TATU*

*Urganch filiali o'qituvchisi*

[kamolaerkinovna@gmail.com](mailto:kamolaerkinovna@gmail.com)

*Razzakova Gulora Razzakberdi qizi*

*Muxammad al-Xorazmiy nomidagi TATU*

*Urganch filiali o'qituvchisi*

[RazzakovaG0597@gmail.com](mailto:RazzakovaG0597@gmail.com)

**Annotatsiya:** Zamonaviy tizim operatsiyalari uchun zarur bo'lgan parallel ishlov berish bir vaqtning o'zida ishlaydigan bir nechta CPU orqali ma'lumotlarni qayta ishlash vazifalarining bir nechta oqimini qo'llab-quvvatlaydi. Parallel ishlov berish - bu umumiy vazifaning alohida qismlarini bajarish uchun ishlaydigan ikki yoki undan ortiq protsessorlarni (CPU) hisoblash usuli. Bir nechta protsessorlar orasida vazifaning turli qismlarini ajratish dasturni ishga tushirish vaqtini qisqartirishga yordam beradi. Bir nechta protsessorga ega har qanday tizim parallel ishlov berishni, shuningdek, bugungi kunda kompyuterlarda keng tarqalgan ko'p yadroli protsessorlarni amalga oshirishi mumkin. Ushbu maqolada parallel ishlov berish qanday ishlashi va uni real hayotda qo'llash misollari tushuntiriladi.

**Kalit so'zlar:** Parallel ishlov berish, ko'p yadroli protsessorlar, massiv parallel protsessorlar (MPP), Klasterlar.

**Abstract:** The parallel processing required for modern system operations supports multiple streams of data processing tasks through multiple CPUs running concurrently. Parallel processing is a method of computing two or more processing units (CPUs) working to complete separate parts of a common task. Splitting different parts of a task

between multiple processors helps reduce program execution time. Any system with multiple processors can implement parallel processing, as well as the multi-core processors that are common in computers today. This article explains how parallel processing works and examples of its real-life applications.

**Keywords:** Parallel processing, multi-core processors, massively parallel processors (MPP), Clusters.

**Аннотация:** Параллельная обработка, необходимая для современных системных операций, поддерживает несколько потоков задач обработки данных с помощью нескольких процессоров, работающих одновременно. Параллельная обработка - это метод вычислений двух или более процессоров (ЦП), работающих над выполнением отдельных частей общей задачи. Разделение различных частей задачи между несколькими процессорами помогает сократить время выполнения программы. Любая система с несколькими процессорами может реализовать параллельную обработку, а также многоядерные процессоры, которые сегодня распространены в компьютерах. В этой статье объясняется, как работает параллельная обработка, и приводятся примеры ее реального применения.

**Ключевые слова:** параллельная обработка, многоядерные процессоры, процессоры с массовым параллелизмом (MPP), кластеры.

## **KIRISH**

Parallel ishlov berish - bu bir vaqtning o'zida ishlaydigan ko'plab markaziy ishlov berish birliklari (CPU) orqali bir nechta hisoblar oqimi yoki ma'lumotlarni qayta ishlash vazifalari birgalikda amalga oshirilganda hisoblash texnikasi. Parallel ishlov berish bitta faoliyatning turli komponentlarini boshqarish uchun bir vaqtning o'zida ikki yoki undan ortiq protsessor yoki protsessordan foydalanadi. Tizimlar vazifaning ko'p qismlarini bir nechta protsessorlarga bo'lish orqali dasturni bajarish vaqtini qisqartirishi mumkin. Zamonaviy kompyuterlarda tez-tez uchraydigan ko'p yadroli protsessorlar va bir nechta protsessorli har qanday tizim parallel ishlov berish qobiliyatiga ega.[1]

Tezlikni oshirish, quvvat sarfini kamaytirish va bir nechta faoliyatni yanada samarali boshqarish uchun ko'p yadroli protsessorlar ikki yoki undan ortiq protsessorli integral mikrosxemalar (IC) chiplaridir. Aksariyat kompyuterlar ikkitadan to'rttagacha yadroga ega bo'lishi mumkin, boshqalari esa o'n ikkitagacha bo'lishi mumkin. Murakkab operatsiyalar va hisob-kitoblar ko'pincha parallel ishlov berishda bajariladi. Eng fundamental darajada registrlardan foydalanish usuli parallel va ketma-ket operatsiyalarni ajratib turadi. Shift registrlari ketma-ket ishlaydi, har bir bitni bir vaqtning o'zida qayta ishlaydi, parallel yuklash registrlari esa so'zning har bir bitini bir vaqtning o'zida qayta ishlaydi. Bir vaqtning o'zida bir xil yoki turli xil faoliyatni amalga oshiradigan turli funksional birliklardan foydalangan holda parallel ishlov berishni murakkablikning yuqori darajasida boshqarish mumkin. Odatda har bir protsessor normal ishlaydi va ko'rsatmalarga muvofiq parallel ravishda operatsiyalarni bajaradi va kompyuter xotirasidan ma'lumotlarni oladi. Protsessorlar, shuningdek, bir-biri bilan aloqa qilish uchun dasturiy ta'minotga tayanadi, shuning uchun ular ma'lumotlar qiymatlari o'zgarishi bo'yicha sinxronlashtirish oladilar. Agar barcha protsessorlar bir-biri bilan sinxronlashtirilsa, vazifa oxirida dasturiy ta'minot barcha ma'lumotlar qismlarini bir-biriga moslashtiradi.

Parallel hisoblashlarga qiziqish 1950-yillarning oxirida boshlangan va superkompyuterlar rivojlanishi 1960-1970-yillarda paydo bo'la boshlagan. Ushbu multiprotsessorlar umumiy xotira maydonidan foydalangan va bitta ma'lumotlar to'plamida parallel operatsiyalarni amalga oshirgan. 1980-yillarning o'rtalarida Caltech Concurrent Computation loyihasi 64 ta Intel 8086/8087 protsessorlaridan foydalangan holda ilmiy ilovalar uchun superkompyuterni qurganda, parallel hisoblashning yangi turi joriy etildi.[2]

Parallel hisoblash deganda bir vaqtning o'zida bir nechta protsessor yoki dasturni bajarish jarayoni tushuniladi. Umuman olganda, bu hisoblash arxitekturasi bir turi bo'lib, unda katta muammolar bir vaqtning o'zida qayta ishlanishi mumkin bo'lgan mustaqil, kichikroq, odatda o'xshash qismlarga bo'linadi. Bu umumiy xotira orqali

bog'langan bir nechta protsessorlar tomonidan amalga oshiriladi, bu yakunlangandan keyin natijalarni birlashtiradi. Bu katta hisob-kitoblarni bajarishda yordam beradi, chunki u katta muammoni bir nechta protsessorlar o'rtasida taqsimlaydi.

Parallel hisoblash, shuningdek, tizimlarning mavjud hisoblash quvvatini oshirish orqali ilovalarni tezroq qayta ishlash va vazifalarni hal qilishga yordam beradi. Parallel hisoblash tamoyillari ko'pchilik superkompyuterlar tomonidan qo'llaniladi. Katta ishlov berish quvvati yoki hisoblashni talab qiladigan operatsion senariylar, odatda, parallel ishlov berish u yerda qo'llaniladi.[3]

Odatda, bu infratuzilma turli protsessorlar server tokchasiga o'rnatilgan joyda joylashgan; dastur serveri hisoblash so'rovlarini kichik bo'laklarga taqsimlaydi, so'ngra so'rovlar har bir serverda bir vaqtning o'zida qayta ishlanadi. Eng qadimgi kompyuter dasturlari ketma-ket hisoblash uchun yozilgan, chunki ular bir vaqtning o'zida bitta ko'rsatmani bajarishga qodir, ammo parallel hisoblash bir vaqtning o'zida bir nechta protsessor dastur yoki hisob-kitoblarni bajarganida farq qiladi.

Parallel hisoblash turlari. Ochiq manbali va xususiy parallel hisoblashning odatda uchta turi mavjud, ular quyida muhokama qilinadi:

1. Bit darajasidagi parallelizm: parallel hisoblash shakli, unda har bir vazifa protsessor so'z hajmiga bog'liq. Katta o'lchamli ma'lumotlarda vazifani bajarish nuqtai nazaridan, protsessor bajarishi kerak bo'lgan ko'rsatmalar sonini kamaytiradi. Operatsiyani bir qator ko'rsatmalarga bo'lish kerak. Misol uchun, 8-bitli protsessor mavjud va siz 16-bitli raqamlar bilan operatsiya qilishni xohlaysiz. Birinchidan, u 8 ta pastki tartibli bitni, keyin esa 8 ta yuqori tartibli bitni ishlashi kerak. Shuning uchun operatsiyani bajarish uchun ikkita ko'rsatma kerak. Amaliyot 16 bitli protsessor tomonidan bitta buyruq bilan amalga oshirilishi mumkin.

2. Yo'riqnoma darajasidagi parallelizm: bitta protsessor soat siklida protsessor ko'rsatmalar darajasidagi parallelizmida bir vaqtning o'zida qancha ko'rsatmalar bajarilishini hal qiladi. Har bir soat sikli bosqichi uchun ko'rsatmalar darajasidagi parallelizmdagi protsessor bitta ko'rsatmadan kam bo'lgan manzilga murojaat qilish

qobiliyatiga ega bo'lishi mumkin. Yo'riqnoma darajasidagi parallelizmdagi dasturiy yondashuv statik parallelizmda ishlaydi, bunda kompyuter bir vaqtning o'zida qaysi ko'rsatmalarni bajarishni hal qiladi.

3. Vazifa parallelligi: vazifalar parallelizmi - bu parallelizmning shakli bo'lib, unda vazifalar kichik vazifalarga bo'linadi. Keyin, har bir kichik vazifa bajarish uchun ajratiladi. Va pastki vazifalarning bajarilishi protsessorlar tomonidan bir vaqtda amalga oshiriladi.[4]

Umuman olganda, parallel ishlov berish vazifani kamida ikkita mikroprotsessor o'rtasida taqsimlashni anglatadi. Bu g'oya juda oddiy: kompyuter olimi murakkab muammoni uning tarkibiy qismlariga ajratish uchun vazifa uchun yaratilgan maxsus dasturiy ta'minotdan foydalanadi. Keyin, ular har bir qism uchun maxsus protsessorni belgilaydilar. Butun hisoblash muammosini hal qilish uchun har bir protsessor o'z qismini yakunlaydi. Dastur murakkab dastlabki muammoni hal qilish uchun ma'lumotlarni qayta yig'adi. Qayta ishlash parallel ravishda amalga oshirilganda, katta ish mavjud ishlov berish birliklarining soni, hajmi va turiga yaxshiroq mos keladigan bir nechta kichikroq ishlarga bo'linadi. Vazifa bo'lingandan so'ng, har bir protsessor boshqalar bilan gaplashmasdan o'z qismida ishlay boshlaydi. Buning o'rniga, ular bir-biri bilan aloqada bo'lish va vazifalari qanday ketayotganini bilish uchun dasturiy ta'minotdan foydalanadilar.

Parallel ishlov berish tizimi vazifalarni tezroq bajarish uchun bir vaqtning o'zida ma'lumotlarni qayta ishlashga qodir. Masalan, joriy ko'rsatma protsessorning arifmetik-mantiqiy birligi (ALU) tomonidan qayta ishlanganligi sababli tizim xotiradan keyingi ko'rsatmani olishi mumkin. Parallel ishlov berishning asosiy maqsadi kompyuterining qayta ishlash quvvatini oshirish va o'tkazish qobiliyatini yoki ma'lum bir vaqt ichida bajarishi mumkin bo'lgan ish hajmini oshirishdir. Bir vaqtning o'zida o'xshash harakatlarni amalga oshirish orqali parallel ishlov berish tizimini yaratish uchun ko'plab funktsional birliklardan foydalanish mumkin.[5]

Kompyuter olimi odatda murakkab vazifani kichikroq qismlarga ajratish va har bir qismni protsessorga belgilash uchun dasturiy vositadan foydalanadi. Keyin har bir protsessor o'z qismini hal qiladi va javobni o'qish yoki operatsiyani bajarish uchun ma'lumotlar dasturiy vosita tomonidan qayta yig'iladi. Har bir protsessor odatda kompyuter xotirasidan ma'lumotlarni o'qish paytida ko'rsatilgandek ishlaydi va parallel vazifalarni bajaradi. Protsessorlar, shuningdek, ma'lumotlar qiymatlaridagi o'zgarishlarni bog'lash va kuzatib borish uchun dasturiy ta'minotdan foydalanadilar. Vazifadan so'ng, dasturiy ta'minot barcha protsessorlar sinxronlashtirilsa, barcha ma'lumotlar qismlarini bir-biriga moslashtiradi. Agar kompyuterlar klaster hosil qilish uchun tarmoqqa ulangan bo'lsa, parallel hisoblash uchun bir nechta protsessorlardan foydalanish mumkin.

Parallel ishlov berishning bir nechta turlari mavjud, ulardan ikkitasi eng ko'p ishlatiladigan SIMD va MIMD. SIMD yoki bitta buyruqli bir nechta ma'lumotlar - bu parallel ishlov berish shakli bo'lib, unda kompyuterda ikki yoki undan ortiq protsessorlar bir xil buyruqlar to'plamiga amal qiladi, har bir protsessor turli ma'lumotlarni qayta ishlaydi. SIMD odatda bir xil belgilangan mezonlarga asoslangan katta ma'lumotlar to'plamlarini tahlil qilish uchun ishlatiladi. MIMD yoki bir nechta buyruqli ma'lumotlar parallel ishlov berishning yana bir keng tarqalgan shakli bo'lib, har bir kompyuterda ikkita yoki undan ortiq o'z protsessorlari mavjud va ma'lumotlarni alohida ma'lumotlar oqimlaridan oladi.

1. Yagona ko'rsatma, yagona ma'lumot (SISD). Yagona ko'rsatma, Yagona ma'lumot deb nomlangan hisoblash turida bitta protsessor bir vaqtning o'zida bitta algoritmi yagona ma'lumot manbai sifatida boshqarish uchun javobgardir. Boshqarish bloki, protsessor birligi va xotira blokiga ega bo'lgan kompyuter tashkiloti SISD bilan ifodalanadi. U hozirgi seriyali kompyuterga o'xshaydi. Ko'rsatmalar SISD tomonidan ketma-ket amalga oshiriladi, ular konfiguratsiyasiga qarab parallel ishlov berishga qodir yoki bo'lmasligi mumkin. Ketma-ket bajarilgan ko'rsatmalar butun ijro bosqichlarida o'tishi mumkin. SISD kompyuterida bir nechta funktsional blok bo'lishi mumkin. Shu

bilan birga, bitta boshqaruv bloki barcha funktsional birliklar uchun javobgardir. Bunday tizimlar quvurlarni qayta ishlashga yoki parallel ishlov berishga erishish uchun ko'plab funktsional birliklardan foydalanishga imkon beradi.

2. Bir nechta ko'rsatmalar, yagona ma'lumotlar (MISD). Multiple Instruction, Single Data - buyruqlar majmuasidan foydalanadigan kompyuterlarda bir nechta protsessorlar standart hisoblanadi. Bir nechta algoritmlardan foydalanganda, barcha protsessorlar bir xil kirish ma'lumotlarini almashadilar. MISD kompyuterlari bir vaqtning o'zida bir xil ma'lumotlar to'plamida ko'plab operatsiyalarni bajarishi mumkin. Kutilganidek, operatsiyalar soni mavjud protsessorlar soniga ta'sir qiladi. MISD tuzilmasi ko'plab qayta ishlash bloklaridan iborat bo'lib, ularning har biri o'z ko'rsatmalari va taqqoslanadigan ma'lumotlar oqimi orqali ishlaydi. Bitta protsessorning chiqishi keyingi protsessor uchun kirishga aylanadi. Ushbu tashkilotning debyuti kam e'tiborga sazovor bo'ldi va arxitekturada ishlatilmadi.[6]

### **Parallel ishlov berish algoritmlarida arxitekturaning roli**

Arxitektura parallel algoritmlarni amalga oshirish uchun amaliy asos bo'lib xizmat qiladi. Arxitektura tizim dizayniga ishora qilib, uning turli tarkibiy qismlarining konfiguratsiyasi va o'zaro bog'liqligini ta'kidlaydi. Arxitektura protsessorlar qanday tashkil etilganligini, ular qanday aloqa qilishini va ma'lumotlar va xotirani qanday almashishini belgilaydi. Arxitektura turli shakllarda bo'ladi, ularning har biri o'ziga xos afzallik va murakkabliklarga ega. Quyida eng keng tarqalgan turlari ko'rib chiqiladi:

**Ko'p protsessorlar:** Bu umumiy xotiraga ega bo'lgan bir nechta protsessor yoki yadroli tizimlar. Ular nozik parallelizm uchun (vazifalar soni bo'yicha nisbatan kichik bo'lgan joylarda), ayniqsa umumiy ma'lumotlar tuzilmasi manipulyatsiyasi ishtirok etganda juda samarali.

**Ko'p kompyuterlar:** Bular tarmoq orqali o'zaro bog'langan alohida kompyuterlar yoki tugunlar to'plamidir. Har bir tugun o'z protsessoriga va xotirasiga ega bo'lib, ular xabarlarni uzatish orqali bog'lanadi. Ular qo'pol taneli parallelizmni boshqarishda samarali (vazifalar soni bo'yicha vazifalar nisbatan katta).

Arxitekturani tanlash parallel algoritm talablariga asoslanadi. Ba'zi algoritmlar ko'p protsessorlar uchun ko'proq mos keladi, boshqalari esa ko'p kompyuterlarda eng yaxshi ishlashi mumkin. Shuningdek, qarorga hisoblashning tabiati, ish yuki, masshtablilik va xatolarga chidamlilik zarurati ta'sir qiladi. [7]

### **Samarali parallel ishlov berish uchun algoritmlarni qo'llash**

Algoritmlarni parallel muhitlarga qo'llash jarayoni mos parallel arxitekturalarda parallel algoritmlarni loyihalash, amalga oshirish va optimallashtirishni o'z ichiga oladi. Bu parallelizmdan to'liq foydalanish uchun arxitektura xususiyatlarini, potentsial tuzoqlarni va algoritmik strategiyalarni chuqur tushunishni talab qiladi.

#### **Algoritmni qo'llashda bir necha bosqichlar bajariladi:**

Vazifalarni parchalash: Muammo parallel ravishda bajarilishi mumkin bo'lgan bir qator vazifalarga bo'linadi. Bu vazifalarning nozikligi - ular qo'pol yoki nozik taneli bo'ladimi - algoritm ishlashiga jiddiy ta'sir qiladi.

Ma'lumotlarni taqsimlash: ishlov beriladigan ma'lumotlar protsessorlar o'rtasida taqsimlanadi. Ushbu taqsimlash muvozanatli yuk va minimal protsessorlararo aloqani ta'minlashi kerak.

Aglomeratsiya: Kichik vazifalar kattaroq vazifalarni shakllantirish uchun birlashtirilib, har bir vazifani alohida bajarish uchun sarflanadigan xarajatlarni kamaytiradi.

Xaritalash: Vazifalar protsessorlarga joylashtiriladi yoki tayinlanadi. Optimal xaritalash aloqa xarajatlarini kamaytiradi va protsessorlar o'rtasidagi yukni muvozanatlashtiradi.

Parallel algoritmni loyihalashda qiyinchiliklar mavjud emas, chunki sinxronizatsiya, vazifalar o'rtasidagi aloqa va ma'lumotlarga bog'liqlik bilan bog'liq qo'shimcha xarajatlarni hisobga olish kerak. Bu omillar to'siqlarni oldini olish va resurslardan samarali foydalanishni ta'minlash uchun ehtiyotkorlik bilan boshqarilishi kerak.

### **FOYDALANILGAN ADABIYOTLAR RO'YHATI**



1. <https://oefen.uz/uz/documents/slaydlar/umumiy/big-data-ni-saqlash-va-uni-qayta-ishlashda-parallel-va-taqsimlangan-hisoblash-yondashuvlari>

Murojaat vaqti:14.05.2024 21:30

2. <http://www.genderi.org/parallel-ishlov-berish.html>

Murojaat vaqti:17.05.2024 09:33

3. <https://www.spiceworks.com/tech/iot/articles/what-is-parallel-processing/>

Murojaat vaqti:19.05.2024 10:10

4. [https://www.spiceworks.com/tech/iot/articles/what-is-parallel-processing/#\\_001](https://www.spiceworks.com/tech/iot/articles/what-is-parallel-processing/#_001)

Murojaat vaqti:21.05.2024 17:45

5. <https://bookdown.org/rdpeng/rprogdatascience/parallel-computation.html>

Murojaat vaqti:22.05.2024 08:32

6. <https://bookdown.org/rdpeng/rprogdatascience/parallel-computation.html>

Murojaat vaqti:23.05.2024 16:15

7. <https://www.studysmarter.co.uk/>

Murojaat vaqti:22.05.2024 14:08