

ӘОК 004.42

ДЕРЕКТЕРДІ ПАРАЛЛЕЛЬ ЖӘНЕ ТАРМАҚТАЛҒАН ӨНДЕУ ТЕХНОЛОГИЯЛАРЫ

Манақбай Алдияр Бөкейханұлы

aldiyar_mb@mail.ru

Л.Н.Гумилев атындағы ЕҰУ, Ақпараттық технологиялар факультеті

Параллельді бағдарламалау термині бірнеше процессорлық құрылғылардан тұратын компьютерлік жүйелерде есептеулерді ұйымдастыруға байланысты кең ауқымды білдіреді. Мұндай жүйелерге көп ядролық процессорлар, ортақ жады бар көп процессорлық машиналар, бөлінген жады немесе гибридік архитектурасы бар жоғары тиімді есептеуіш кластерлер жатады.

Соңғы кездері параллель есептеулерге көп көңіл бөлінуде. Бұл негізінен екі факторға байланысты. Бірінші фактор ғылыми және технологиялық прогрестің арқасында пайда болды, нәтижесінде математикалық модельдеу әдістерін қолдануды талап ететін білімнің жаңа бағыттары пайда болды. Модельдері де күрделене түседі. Нәтижесінде, ресурстарды қарқынды есептеулерге деген қажеттілік тұрақты түрде өсіп келеді, кейбір жағдайларда параллельді немесе бөлінген есептеу әдістерін қолдана отырып, жоғары өнімді жабдық негізінде ғана орындала алады.

Бүгінгі күнде параллельді бағдарламалаудың қолданысқа ие болған технологиялары ол MPI және OpenMP болып табылады.

MPI технологиясы тиімді жүзеге асыруға болатын процессорлар арасында байланыс аппаратурасына үлкен талап қоймаса да, OpenMP технологиясына қарағанда қолданылу жағынан қиынырақ, сондықтан бағдарламаушыға қойылатын талап жоғары болады.

Осының негізінде OpenMP (SMP-серверлер) тиімді түрде іске асуы мүмкін болатын аппаратура өте қымбат және жақсы емес. MPI (мамандандырылған есептеу кластерлерінің мамандандырылған желілерін) енгізу үшін жарамды аппаратура әлдеқайда арзан және шектелмейді, алайда әлдеқайда төмен тиімділікке ие.

Соңғы жылдары есептеуіш кластерлердің мамандандырылған желілері қарқынды дамып, салыстырмалы түрде арзан және кең ауқымды болып келеді. SMP серверлері мен мамандандырылған желілер арасындағы өзара әрекеттесудің тиімділігі мен сапасы арасындағы айырмашылық әлі де үлкен, дегенмен, ол бірнеше жыл бұрын мамандандырылған желілерді жетілдірумен байланысты әлдеқайда аз болды. Қазіргі мамандандырылған желілер тіпті 5-10 жыл бұрын болмаған көп нәрсені біледі [1].

Осыған орай, аралық бағдарламалау технологиясы туралы мәселе орынды болып көрінеді, бұл аппаратураның екі классы арасындағы алшақтықты толықтай жабу сәтін күтпестен, бағдарламашыға осы жетілдірулерді сезінуге мүмкіндік береді.

Бұл технология 10 жыл бұрын Cray арқылы дамыды, ал бастапқы уақытта басқа фирмалардағы компьютерлерде ешқашан пайдаланылмаған. Көптеген жылдар бойы практикалық тестілеуден сәтті өткен кластерлік суперкомпьютерлерге аралық параллельді бағдарламалау технологиясындағы Cray оқиғалары кеңейуде.

Технология shmем (ортақ жад) деп аталады және MPI сияқты, функциялардың кітапханасы (кіші бағдарламалар) болып табылады.

Бұл технология MPI-ден статикалық (бағдарламаны іске қосу кезінде) процестің туындауын, сондай-ақ осы процесс үшін «өзінің» және «бөтен» жады арасындағы айқын айырмашылықты мұра етті.

OpenMP-тен - процесті синхрондаудың негізгі режимі ретінде кедергі синхрондауды және барлық процестерге қолжетімді болатын қандай да бір бөлінетін жады кеңістігін алды.

Shmem - параллельді бағдарламалау кітапханаларының жиыны, олар кедергілері аз, бөлінген жадысы бар суперкомпьютерлер үшін RDMA параллельді өңдеудің бір жақты интерфейстерін қамтамасыз етеді.

Shmem-дегі жад кеңістігі төмендегідей орналасады.

Егер бағдарламада A массиві жарияланса, онда әрбір процестің өз массиві болады (MPI сияқты). Дегенмен, әрбір процесс «қояу» (put ()) немесе «қабылдау» (get ()) функциялары көмегімен кез келген басқа процестің A массивіне тікелей қол жеткізе алады.

MPI-ден айырмашылығы, деректердің «меншік иесінен» «жауап берушілік» туралы «бөтен» деректерге қол жеткізу талап етілмейді [2].

Shmem-де MPI-мен (және OpenMP сияқты) салыстырғанда, бағдарламамен талап етілетін синхрондау деректер алмасуынан бөлек ұйымдастырылуы керек. Ол үшін кедергілер қолданылады. MPI-де кедергі синхронизациясы қарастырылады, алайда ол қосымша сипатқа ие, барлық бағдарламаларда пайдаланылмайды. Shmem –де OpenMP сияқты, бұл синхрондау әдісі негізгі болып табылады.

Shmem-нің маңызды ерекшелігі - қысқа хабарламаларды алмасудың жоғары тиімділігі (әсіресе жазу кезінде). Қосымшалардың кейбір түрлерінде, мысалы, құрылымдалмаған торлар бойынша есептеулерде, процестер деректермен алмасады, ол деректер жадыда ірі бөліктермен емес, күрделі жолмен жадқа бөлінген көптеген шағын бөліктерде орналасады. Мұндай жағдайларда MPI-ді пайдалану кезінде, тиімділік үшін жіберу барысында деректерді ірі пакеттерге орауға, ал қабылдау кезінде ашуға тура келеді. Ал Shmem-ді пайдаланған кезде мұндай қажеттілік болмайды: деректердің кішкене бөліктерін (жеке нөмірлерге дейін) бөлек «put ()» операциялары арқылы бірден қажетті орынға жіберуге болады. Осыдан тиімділік іс жүзінде зардап шекпейді, ал бағдарлама, әдетте, айтарлықтай оңайлатылады [3].

Shmem кітапханасын үш тәсілмен пайдалануға болады:

1. Жеке байланыс кітапханасы ретінде. MBS-экспресс, суперкомпьютерінде, мысалы, бүгінгі күні shmem-express кітапханасы – байланыстың жалғыз негізгі құралы. Осы машинаға арналған MPI іске асыру әзірше жоқ, және жүзеге асқан жағдайда ол shmem бойынша құрылады.

2. MPI-мен бірлеседі. Бұл әдіс кезінде Cray және SGI суперкомпьютерлерінің пайдаланушыларына ұсынылды. Бағдарлама MPI пайдалана отырып жазылған, бірақ MPI-де нашар жазылған бөліктері, мысалы, құрылымдық емес торлармен жұмыс істеу, shmem-ге жүгінеді. MPI-ге «енгізілген» shmem-нің жиынын іске асыру қазіргі уақытта ИПМ РАН кластерлерінде қол жетімді.

3. OpenMP-мен бірлеседі. Бүгінде дерлік барлық есептеу кластерлерінің барлығы дерлік көпядролық түйіндер негізінде құрылады және олардың әрқайсысы SMP сервері болып табылады. Ал кластер торабындағы процессорлық ядролардың саны аз (2-4) болғандықтан, бағдарламашылар, әдетте, торап шегіндегі барлық ядроларға ортақ жадты елемеді, әр түрлі ядролардағы процестердің өзара әрекеттесуін, әр ядро өзге ядромен тек тор арқылы байланысқан секілді, ұйымдастырды. Қазіргі уақытта құрамында 8, 16, 32 және одан көп процессорлық ядролары бар кластер түйіндері қалыпты болып табылады. Бұл түйін ішіндегі процессорлық ядролардың бірлескен жұмысын ұйымдастыру үшін OpenMP пайдаланылатын, ал түйіндердің өзара бір-бірімен әрекеттестігін ұйымдастыру үшін басқа да технология қолданылатын гибриді параллелді бағдарламаларға қызығушылықты күшейтеді. Осы уақытқа дейін бұл «басқа технология» ретінде тек MPI технологиясы немесе оның туындылары ғана болуы мүмкін деп есептелді. Көріп отырғанымыздай, бұл жолы shmem технологиясы жағымдырақ: ол заманауи торлардың мүмкіндіктеріне көбінесе сәйкес келуімен қатар, MPI-мен салыстырғанда, OpenMP бағдарламалау стиліне жақын. Сонымен қатар, shmem MPI-ға қарағанда айтарлықтай қарапайым. Мұның бәрі shmem-ді OpenMP-дің жалғыз SMP серверінен SMM серверлерінің кластеріне көшу кезіндегі кеңейтілімдерінің табиғи жиынтығы ретінде қарастыруға мүмкіндік береді.

MPI төрт негізгі тұжырымдарға негізделген:

- Хабарлама алмасудың операциясының түрі;
- Хабарламада жіберілген деректер түрі;
- Коммуникатор түсінігі(процестер тобы);
- Виртуалды топология түсінігі.

OpenMP-дің негізгі элементтері:

- Ағындарды жасау үшін құрылымдары (parallel директивасы);
- Ағымдар арасындағы жұмыстарды бөлу құрылымдары (DO / for және section директивалары);

- Деректермен жұмысты басқару құрылымдары (shared және private ұғымдары);
- Ағындарды синхрондау құрылымдары (critical, atomic және barrier директивалары);
- Орындау уақытын қолдау кітапханасының рәсімдері (мысалы, omp_get_thread_num);
- Қоршаған орта айнымалылары (мысалы, OMP_NUM_THREADS).

Ортақ жад жүйелеріне MPI және OpenMP технологияларын салыстыру

2-процессорлық сервер үшін OpenMP және MPI нұсқаларымен матрицалық көбейтуге арналған бағдарламаның кезекті нұсқасының орындалу уақытын салыстыру (Кесте 1).

Кесте 1

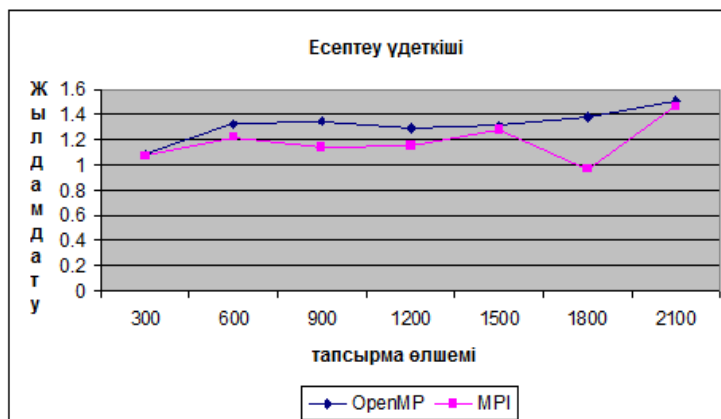
Матрицалық көбейтуге арналған бағдарламаның кезекті нұсқасының орындалу уақытын салыстыру.

Матрица реті: (N)	Уақыт $T_{\text{жүйе}}$ (жүйелі алгоритм)	OpenMP		MPI	
		Уақыт $T_{\text{пар}}$	Жылдамдату S	Уақыт T	Жылдамдату S
300	0.42	0.39	1.08	0.39	1.07
600	4.69	3.55	1.32	3.85	1.22
900	16.20	12.05	1.34	14.17	1.14
1200	38.67	30.00	1.29	33.72	1.15
1500	76.56	58.20	1.32	60.19	1.27
1800	150.08	108.42	1.38	154.73	0.97
2100	258.09	171.75	1.50	177.03	1.46

Параллель есептеулер арқылы матрицалық көбейтуді жеделдету (Кесте 2).

Кесте 2

Матрицалық көбейтуді жеделдету



4-процессорный сервер для OpenMP и MPI с вариантами матричного умножения с вариантами (Кесте 3).

Кесте 3

Матрицалық көбейтуге арналған бағдарламаның дәйекті нұсқасының орындалу уақытын салыстыру

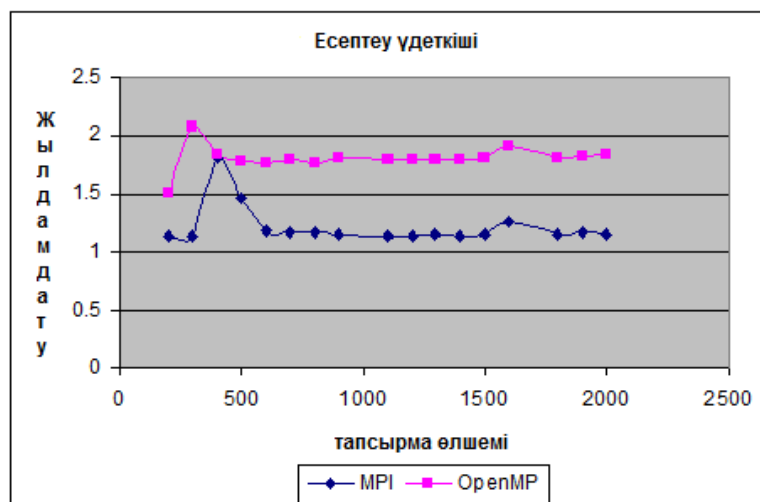
Матрица реті: (N)	Уақыт $T_{\text{жүйе}}$ (жүйелі алгоритм)	OpenMP		MPI	
		Уақыт $T_{\text{пар}}$	Жылдамдату S	Уақыт T	Жылдамдату S
300	0.36	0.17	2.09	0.32	1.14
600	6.66	3.78	1.76	5.61	1.19
900	22.92	12.70	1.80	20.01	1.15

1200	54.53	30.30	1.80	48.19	1.13
1500	107.91	59.67	1.81	93.27	1.16
1800	188.61	103.81	1.82	164.45	1.15
2100	262.09	142.73	1.84	226.45	1.16

Параллель есептеулер арқылы матрицалық көбейтуді жеделдету (Кесте 4).

Кесте 4

Параллель есептеулер арқылы матрицалық көбейтуді жеделдету



Қолданылған әдебиеттер тізімі

1. Barbara, Chapman Using OpenMP – Portable Shared Memory Parallel Programming / Barbara Chapman. - Москва: Мир, 2007. - 384 с.
2. Shokina, Resch, Computational Science and High Performance Computing IV, 2009. – 378 с.
3. Sterling T., High Performance Computing: Modern Systems and Practices, 2017. – 718 с.