

МРНТИ 28.17.19
УДК 004.94

<https://doi.org/10.51889/2959-5894.2023.81.1.006>

Е. Махмут^{1*}, Е.Ф. Кенжебек¹, Т.С. Иманкулов¹

¹Әл-Фараби атындағы Қазақ Ұлттық Университеті, Алматы қ., Қазақстан
*e-mail:erlanmahmut@gmail.com

МРІ+OPENMP ГИБРИДТІ ПАРАЛЛЕЛЬДІ АЛГОРИТМІН ПАЙДАЛАНЫП МҰНАЙ ЫҒЫСТЫРУ МӘСЕЛЕСІН ШЕШУ

Аңдатпа

Соңғы уақытта заманауи ақпараттық технологиялар өнеркәсіптің көптеген салаларында қолданылады және параллельді есептеулерді қолдану сандық модельдеудің өнімділігін арттыруға мүмкіндік береді. Бұл зерттеуде жоғары өнімді есептеу технологияларын қолдана отырып, мұнайды ығыстырудың сандық моделі әзірленді. Атап айтқанда, Бакли-Леверетт моделі арқылы мұнайдың ығысу процесі сипатталды. Жоғары өнімді есептеу технологиялары ретінде OpenMP, MPI және гибриді (OpenMP+MPI) параллельді есептеу моделі қарастырылды. Бұл зерттеудің негізгі мақсаты, әр түрлі технологиялар бойынша параллельді алгоритмдердің есептеу уақытына және үдеуіне салыстырмалы талдау жасау болып табылады. Параллельді алгоритмдердің есептеу моделдері құрылды және нәтижелері салыстырылды.

Түйін сөздер: EOR, HPC, OpenMP, MPI, үлестірмелі жүйелер.

Аннотация

Е. Махмут¹, Е.Ф. Кенжебек¹, Т.С. Иманкулов¹

¹ Казахстанский Национальный Университет имени Ал-Фараби, г. Алматы, Казахстан

РЕШЕНИЕ ЗАДАЧИ ВЫТЕСНЕНИЯ НЕФТИ С ПОМОЩЬЮ ГИБРИДНОГО ПАРАЛЛЕЛЬНОГО АЛГОРИТМА МРІ+OPENMP

В последнее время современные информационные технологии используются во многих областях промышленности, а использование параллельных вычислений позволяет повысить производительность численного моделирования. В данном исследовании была разработана численная модель вытеснения нефти с использованием высокопроизводительных вычислительных технологий. В частности, процесс вытеснения нефти описывался с помощью модели Бакли-Леверетта. В качестве технологий высокопроизводительных вычислений рассматривались OpenMP, MPI и гибридные (OpenMP+MPI) модели параллельных вычислений. Основная цель данного исследования — провести сравнительный анализ времени расчета и ускорения параллельных алгоритмов для разных технологий. Были созданы вычислительные модели параллельных алгоритмов и проведено сравнение результатов.

Ключевые слова: EOR, HPC, OpenMP, MPI, распределенные системы.

Abstract

HYBRID PARALLEL ALGORITHM USING MPI AND OPENMP TO SOLVE OIL DISPLACEMENT PROBLEM

Makhmut E. ^{1*}, Kenzhebek Y. ¹, Imankulov T.S. ¹,

¹Al-Farabi Kazakh National University, Almaty, Kazakhstan

Recently, modern information technologies are used in many areas of industry, and the use of parallel computing allows to increase the productivity of numerical modeling. In this study, a numerical model of oil displacement was developed using high performance computing technologies. In particular, the process of oil displacement was described using the Buckley-Leverett model. OpenMP, MPI and hybrid (OpenMP+MPI) parallel computing models were considered as high-performance computing technologies. The main goal of this study is to make a comparative analysis of the calculation time and acceleration of parallel algorithms for different technologies. Computational models of parallel algorithms were created and the results were compared.

Keywords: EOR, HPC, OpenMP, MPI, distributed systems.

Кіріспе

Мұнай өндіру инженериясында мұнай қабатына суды айдау әдісі маңызды әдістердің біріне айналды. Бұл әдіс арқылы мұнай өндірісінің 40% -60% қамтамасыз етуге болады [1]. Екі фазалы сұйықтықтардың араласпай ағу механизмі көптеген мақалаларда жарияланған. Бір өлшемді, араласпайтын, екі фазалы ағынның ерекше жағдайын 1942 жылы Бакли мен Леверетт [2] теориялық тұрғыдан зерттеді. Сумен ығыстыру процесін Бакли-Леверетт әдісі арқылы математикалық түрде көруге болады. Бакли-Леверетт әдісі бір өлшемді немесе квази-бір өлшемді коллектордағы араласпайтын ығысу процесін сипаттайды және бірнеше қабаттарға бөлінеді. Сандық тәсілдерді пайдалана отырып, өріс бірнеше нүктелерге дискреттеледі. Бұл нүктелер есептеу құнын арттырады, сондықтан есептеу уақытын азайту үшін жоғары өнімді есептеу қажет болады.

Соңғы жылдары жоғары өнімді есептеуіш технологиялар мұнай өндіру өнеркәсібінде күрделі есептерді есептеудің негізгі тенденциясына айналуға бастады. Осылайша, мұнай және газ кен орындарында жоғары өнімді есептеуіш технологияларды (OpenMP, MPI және CUDA) қолдану күн санап өсіп келеді. Жоғары өнімді модельдердің мүмкіндігі есептеу өнімділігінің жоғарлауына байланысты арта түсті. Келесі [3] жұмыс авторлары параллель программалаудың есептеу уақытын ұсынды. Автор өз жұмысында жылдамдық пен тиімділік мәндері параллельді бағдарламалау қатынастарымен ерекшеленетінін атап өтті.

OpenMP параллельді бағдарламалаудың ең кең тараған үлгілерінің бірі болып табылады. Ол ортақ жады пайдаланатын бағдарламалау үлгісі деп те аталады. Сондықтан тапсырмалар ортақ мекенжай кеңістігін бөліседі және бұл тапсырмалар арасындағы байланыс жасырын болады. Белгілі бір дәрежеде бұл ағындар деректерді көшіреді және оны жеке дерек ретінде қарастырады, сондықтан олар жаққа қымбат қолжетімділікті болдырмайды. Бірдей сандағы нүктелерге әртүрлі сандағы ағындардың жылдамдығын [4] жұмыс авторлары ұсынды. Олар OpenMP әдісін пайдаланып, әртүрлі ағындар санының орындалу уақытын есептеді және 16 ағынмен 3,43 максималды жылдамдық нәтижесін алған. Осыған ұқсас [5] жұмыста әртүрлі өткізгіштіктердің үш түрлі жиынын алған, сондай ақ 160 ағын үшін ең жоғары дегенде 7,9 ($k_x = k_y = 20,0 \times 10^{-15} \text{ м}^2$ және $k_z = 8,0 \times 10^{-15} \text{ м}^2$) және 40 ағын үшін ең аз дегенде 4,9 ($k_x = k_y = 5,0 \times 10^{-15} \text{ м}^2$ және $k_z = 2,0 \times 10^{-15} \text{ м}^2$) жылдамдыққа қол жеткізген. Келесі [6] зерттеу авторлары есептеу уақытының құнын төмендету үшін OpenMP параллельді бағдарламалау техникасын ұсынды. Оның жұмысында параллельді есептеу уақыты тізбектей есептеуге қарағанда шамамен 2,7 есеге жетті, ал ұяшықтар санының өсуіне байланысты параллельді есептеудің жылдамдығы артқан.

OpenMP-де ағындар тек бір түйінде (процессорда) жұмыс істей алады, өйткені олар ортақ жады пайдаланады. Барлық ағындар бірдей жады мекенжайларын ортақ пайдаланады және көп түйінді жүйенің параллель процесін жүзеге асыра алмайды. Көп түйінді жүйелерде параллельді есептеулерді жобалау үшін хабарламаны жіберу техникасын қолдану қажет болады.

Хабарламаны жіберуді бағдарламалау үлгісінде әрбір түйіннің өзінің жеке кеңістігі болады және бұл түйіндер арасындағы байланыс анық жүзеге асырылады. Байланыс операциясында екі жақтың тегтері (tag) бірдей болуы керек және әрбір жіберу және қабылдау операциясы бір-біріне сәйкес келуі керек. Хабарламаны жіберу бағдарламасының ең көп тараған үлгісі MPI болып табылады. Кеуекті ортадағы полимерлі сумен ығыстыру процесі есебінің сандық моделі үшін [7] жұмыс авторлары параллельді компьютерлерде мұнай өндіру мәселелерін шешуді жылдамдату үшін MPI пайдаланады.

OpenMP және MPI артықшылықтарын көрсету үшін гибриді параллель технологиялар деп аталатын параллельді есептеу әдістерінің комбинациясы бар. Ол бірнеше параллельді модельді біріктіру арқылы оңтайлы өнімділікке жеткені туралы мәліметтер әдебиетте зерттелген [8]. Сондықтан MPI+OpenMP гибриді моделі параллельді есептеулер үшін өте қолайлы. Гибриді модель ортақ және үлестірмелі жады арасында біріктірілген, осы мәселеде [9] жұмыс авторлары өз еңбегінде MPI+OpenMP гибриді моделін ұсынған.

Есептің қойылымы

Су мен мұнай кеуекті орта қабатында сығылмайтын сұйықтық ретінде қаралады. Қарастырылып отырған модель екі фазалы (су, мұнай) ағын. Су және мұнай үшін массаны сақтау заңдарын ескере отырып, мұнай ығыстыру мәселесінің математикалық моделін төмендегідей сипаттаймыз.

Су фазасының теңдеуі:

$$m \frac{\partial s_w}{\partial t} + \text{div}(\vec{v}_w) = 0; \quad (1)$$

Мұнай фазасының теңдеуі:

$$m \frac{\partial S_o}{\partial t} + \operatorname{div}(\vec{v}_o) = 0; \quad (2)$$

Полимер концентрациясының теңдеуі:

$$\frac{\partial}{\partial t} (m S_w C_p + (1 - m) A_d) - \operatorname{div}(\vec{v}_w C_p) = 0; \quad (3)$$

$$S_o + S_w = 1; \quad (4)$$

мұндағы m – кеуектілік, S_o and S_w мұнай мен судың қанықтылығы, \vec{v}_w and \vec{v}_o су мен мұнай фазаларының жылдамдығы, A_d резервуармен адсорбцияланған полимер, C_p су фазасындағы полимер концентрациясы. Сонымен қатар жылдамдықтар Дарси заңымен өрнектеледі:

$$\vec{v}_i = -k \frac{f_i(S)}{\mu_i} \nabla p; \quad i=o,w \quad (5)$$

мұндағы k – абсолютті өткізгіштік, μ_i ($i=o,w$) – мұнай мен судың тұтқырлығы, $f_i(S)$ келесі теңдеумен өрнектелетін салыстырмалы фазалы өткізгіштік:

$$f_w(S_w) = S_w^2, \quad f_o(S_o) = (1 - S_o)^2, \quad (6)$$

Полимердің адсорбция теңдеуі A_d белгілі бір дәрежеде, C_p полимер концентрациясына байланысты. Бұл байланыс полимер концентрациясының функциясы ретінде сипатталады C_p :

$$A_d = f(C_p); \quad (7)$$

(1) және (2) теңдеуінен шығатын қысым теңдеуі келесідей:

$$\operatorname{div}(\vec{v}_w) + \operatorname{div}(\vec{v}_o) = 0; \quad (8)$$

Бастапқы шарттар: $S|_{t=0} = S_0$; $P|_{t=0} = P_0$; $C_p|_{t=0} = C_{p0}$;

Шекаралық шарттар:

$$S|_{x=0} = S_{inj}; \quad P|_{x=0} = P_{inj}; \quad P|_{x=1} = P_{prod}; \quad C_p|_{x=0} = C_{pinj};$$

$$\frac{\partial s}{\partial x}|_{x=1} = 0; \quad \frac{\partial p}{\partial x}|_{x=1} = 0;$$

Бакли-Леверетт моделі [2] жоғарыдағы (1) - (8) мұнай ығыстыру мәселесін шешу үшін қолданылды. Бакли-Леверетт моделінде келесі болжамдар қарастырылды:

- ағын сызықты, горизонтал және тұрақты қалыңдықта;
- ағын тұрақты температураны сақтайды, сығылмайды және Дарси заңына бағынады;
- су мен мұнай араласпайды;
- гравитация және капиллярлық қысым әсерлері шамалы;
- кеуектілік тұрақты деп есептеледі.

Есепті шешудің сандық модель

Мұнайдың ығысуының жоғарыда келтірілген (1) - (8) математикалық моделі сызықты емес, бұл мәселені аналитикалық жолмен шешу өте қыйын. Сондықтан, бұл мақалада осы мәселені сандық түрде шешу үшін Якоби әдісі қолданылды.

Қысым үшін:

$$P_i^{t+1} = \frac{M_{i+\frac{1}{2}} P_{i+1}^t + M_{i-\frac{1}{2}} P_{i-1}^t}{M_{i+\frac{1}{2}} + M_{i-\frac{1}{2}}}; \quad (9)$$

мұнда,

$$M_{i+\frac{1}{2}} = \frac{M_i + M_{i+1}}{2}; \quad M_{i-\frac{1}{2}} = \frac{M_i + M_{i-1}}{2}; \quad M_i = \left[-k \frac{f_0(s)}{\mu_o} \right] + \left[-k \frac{f_w(s)}{\mu_w} \right];$$

Қанықтылық үшін:

$$S_i^{t+1} = S_i^t + \frac{\Delta t}{m\Delta x^2} \left[K_{i+\frac{1}{2}}(P_{i+1}^t - P_i^t) - K_{i-\frac{1}{2}}(P_i^t - P_{i-1}^t) \right]; \quad (10)$$

Мұнда,

$$K_{i+\frac{1}{2}} = \frac{K_i + K_{i+1}}{2}; \quad K_{i-\frac{1}{2}} = \frac{K_i + K_{i-1}}{2}; \quad K_i = -k \frac{f_w(S)}{\mu_w}.$$

Полимерді концентрациясы үшін:

$$U_i = \left(-k(f(s)_i^{t+1}) \right) (P_{i+1}^{t+1} - P_i^{t+1}) / (\mu_w \Delta x) \quad (11)$$

$$W_i^{t+1} = W_i^t - (\Delta t / (2\Delta x)) (U_i (Cp_{i+1}^t - Cp_i^t) - (U_i (Cp_i^t - Cp_{i-1}^t))) \quad (12)$$

$$Cp_i^{t+1} = W_i^t / (mS_i^{t+1} + (1 - m)\tau) \quad (13)$$

мұндағы τ – адсорбциялық функцияның коэффициенті.

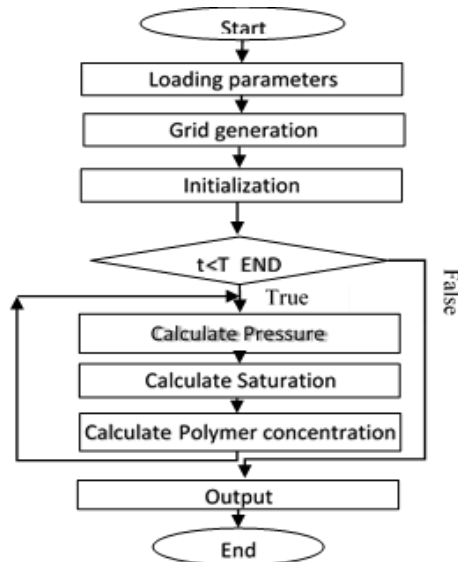
Есептің сызықты бағдарламалық алгоритмі төменде 1-суретте 4 қадаммен сипатталған.

1) Параметрлерді енгізу. Бұл бастапқы қадамда су мен мұнай үшін тұтқырлық, өткізгіштік, су қанықтылығы үшін бастапқы мән, мұнай қанықтылығы және полимер концентрациясы, кеуектілік сияқты мұнайды алудың параметрлері енгізіледі.

2) Инициализация. Есептеудің бастапқы шартына сәйкес есептеу торларындағы нүктелердің бастапқы мәндері инициализацияланады.

3) Есептеу. Бұл қадамда қысым, қанықтылық және полимер концентрациясы уақыт қадамы аяқталғанша есептелінеді.

4) Есептеу нәтижесі файлға жазылады және бағдарлама аяқталады.



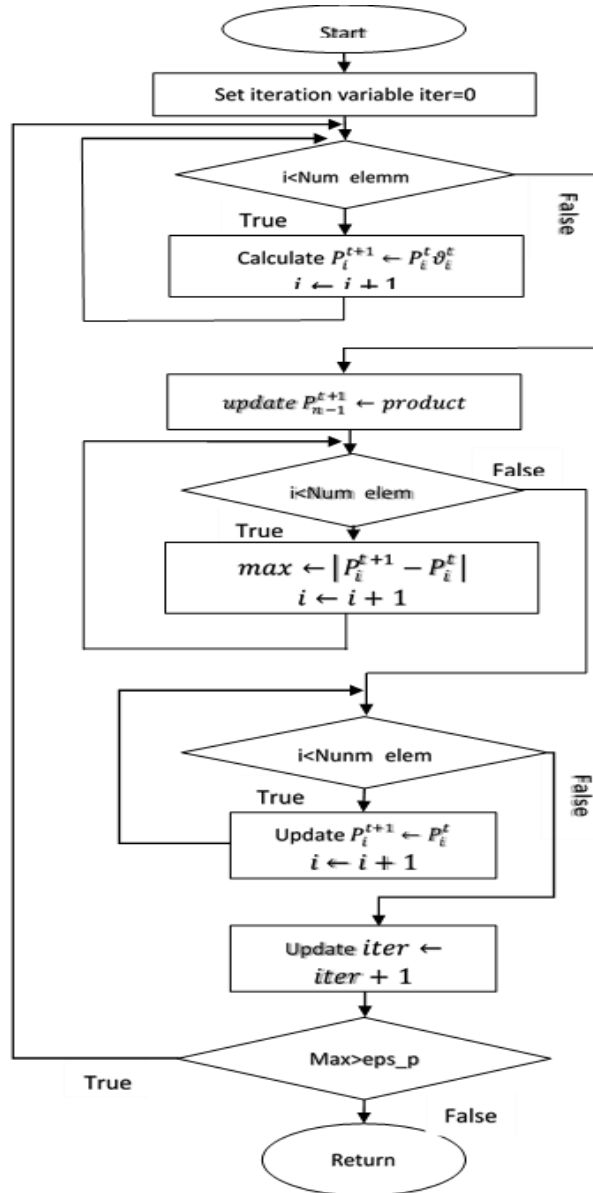
Сурет 1. Тізбектей шешу алгоритмі

Қысымды есептеудің тізбектелген алгоритмі (9) теңдеуге сәйкес 2-суретте сипатталған.

1) n нүкте үшін M_i мәні есептелінеді.

2) n нүкте үшін $M_{i+\frac{1}{2}}$, $M_{i-\frac{1}{2}}$ мәндері M_i -ді пайдаланып есептелінеді.

- 3) n нүкте үшін P_i қысымның мәні $M_{i+\frac{1}{2}}, M_{i-\frac{1}{2}}$ пайдаланып есептелінеді.
- 4) P_i -дың ескі және жаңа мәндерінің айырмасының ең үлкені Max табылады, сондай-ақ, қысымның мәні жаңартылады.
- 5) Есептеу дәлдігі eps пен Max салыстырылады. Max , eps -ден кіші болғанға дейін есептеу циклі жалғастырылады, кері жағдайда есептеу циклі аяқталады.

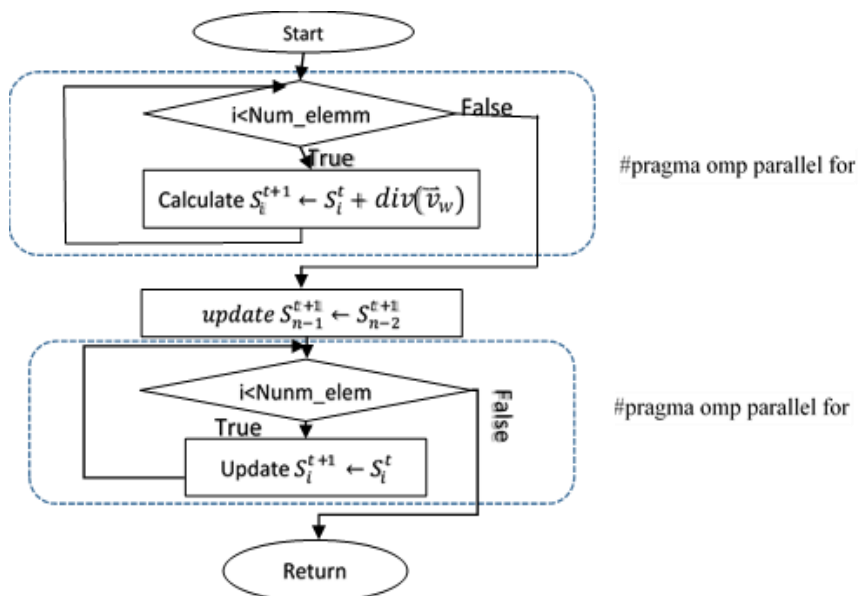


Сурет 2. Қысымды есептеу алгоритмі

MPI, OpenMP және гибриді параллелизация әдістері

Теңдеуді тізбектей шешу үшін Якоби әдісі қолданылды. CPU-де Якоби әдісін параллель жүзеге асыру үшін OpenMP және MPI пайдаланылды. Бұл жұмыста OpenMP параллелизация процесі үшін негізінен for-loop операторына назар аударылды, өйткені OpenMP жақсы қалыптасқан құрылым болып табылады. Ол параллель циклдар, циклдар кестесі сияқты өте кең таралған жоғары өнімді есептеу үлгілеріне қолдау көрсетеді. #pragma omp, #pragma omp parallel for reduction, #pragma omp parallel және #pragma omp critical қолданылды.

Мысал ретінде қанықтылық есептейтін бөлікті қарастырайық. 3-суретте қанықтылықты есептеу процесі қалай параллельденетіні көрсетілген. Үзік сызық сызылған бөлігінде OpenMP пайдаланылды.



Сурет 3. OpenMP арқылы параллелизация

MPI параллелизация процесі үшін екі негізгі фактор қарастырылады. Біріншіден, түбір процессордан басқа процессорларға қанша элемент үлестірілетіні маңызды. Екіншіден, оның қай орыннан жөнелтілетінін ескеру керек.

Біздің теңдеуімізде жаңа мәнді есептеуге алдыңғы және келесі орынның мәні әсер етеді. Мысалы, қысымның жаңа мәнін есептеу келесідей, $P_i^{t+1} = M1P_{i-1}^t + M2P_{i+1}^t$.

Барлық элементтерді басқа процессорларға (түбірді қоса) жіберу үшін $sendcount=(number_of_dispatched_elements/number_of_processor)$ функциясын қолдансақ, келесі жағдайлар орын алуы мүмкін.

Біріншісі, P_i^{t+1} және P_{i-1}^t ұқсас процессорда орналасқан, бірақ P_{i+1}^t басқа процессорға жіберіледі; екіншісі, P_i^{t+1} және P_{i+1}^t ұқсас процессорда орналасқан, бірақ P_{i-1}^t басқа процессорға жіберіледі.

Бұл жағдайда біз P_i^{t+1} есептеу үшін P_i^{t+1} және P_{i-1}^t алдыңғы немесе келесі сәйкес нүктелердің мәнін басқа процессорлардан алуға тура келеді, демек бұл процесс біздің өнімділікке тікелей әсер етеді.

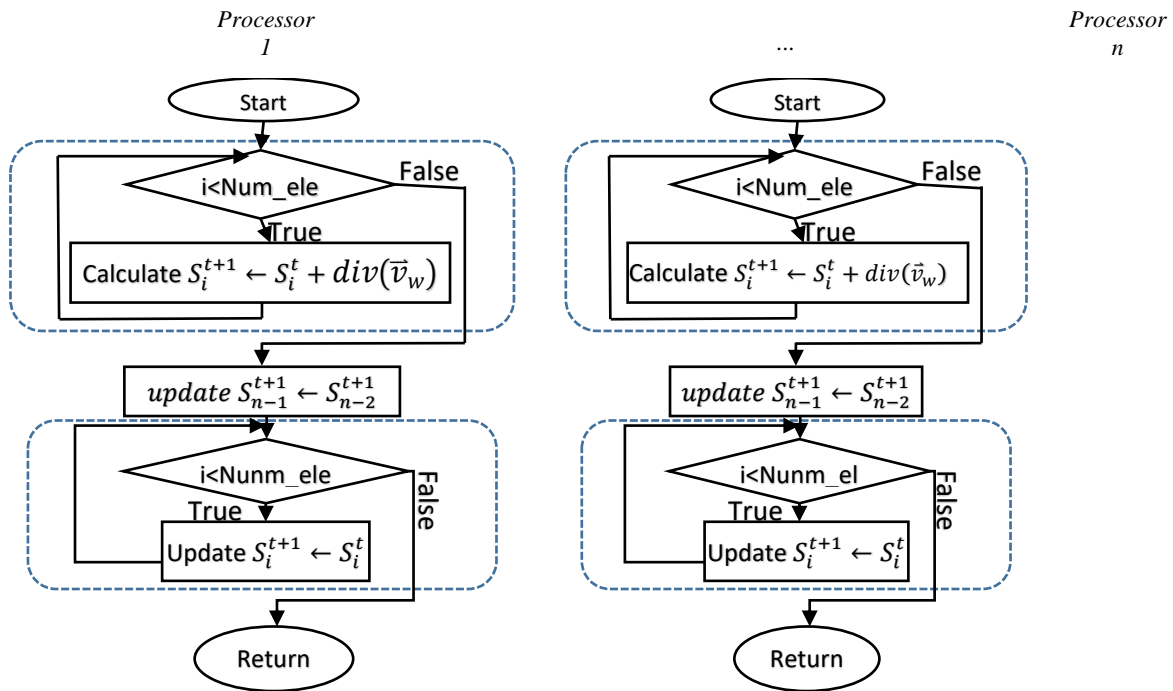
Элементті бірінші және соңғы процессорларға жіберу үшін $sendcount=(number_of_dispatched_elements/number_of_processors)+1$ функциясын қолдансақ, ал элементті басқа процессорларға жіберу үшін $sendcount=(number_of_dispatched_elements/number_of_processors)+2$ функциясын қолданамыз, сол себепті P_i^{t+1} есептейтін кезде ешқандай жіберу және қабылдау процесі болмайды, себебі P_i^{t+1} , бір процессорда.

Осы мақсатқа жету үшін бұл жұмыста $MPI_Scatterv()$ және $MPI_Gatherv()$ әдістері қолданылды.

OpenMP және MPI параллелизациясының гибриді үшін жоғарыда аталған OpenMP және MPI параллелизация процесіне біріктіріп қарастырдық.

Бұл модельде MPI әрбір процессорға деректерді жіберуге жауап береді, ал OpenMP әрбір for-loop мәлімдемесіне жауап береді.

4-суретте OpenMP және MPI моделінің гибриді параллелизация процесі көрсетілген. Үзік сызық бөлігінде OpenMP пайдаланылды.

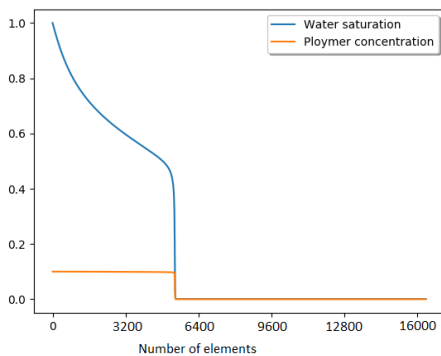


Сурет 4. OpenMP және MPI арқылы гибриді параллелизация

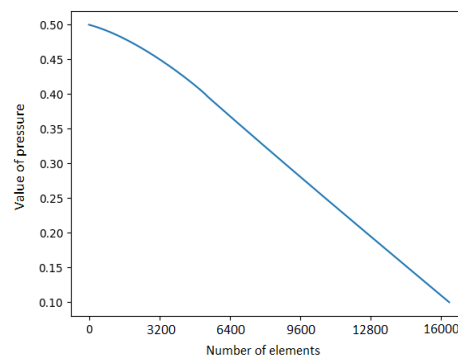
Параллельді есептеу нәтижелері

Жоғарыда қарастырылған мұнай ығыстыру мәселесінің параллельді алгоритмдерінің параллельді бағдарламалары жүзеге асырылды. Параллельді есептеудің нәтижесі алынды және талдау жасалынды.

5(a)- суретде қарастырылған есептің су қанықтылығы мен полимер концентрациясының таралуы, ал 5(b)- суретінде қысым өзгерісі көрсетілген.



Сурет 5(a). Полимер және су қанықтылығының таралуы



Сурет 5(b). Қысымның таралуы

1-кестеде эксперименттерде қолданылған Бакли-Леверетт моделінің кіріс параметрлері көрсетілген.

2-кестеде модельдеудің тізбектей орындау уақытының нәтижелері көрсетілген (қысым үшін). Нүктелер санының көбеюіне байланысты орындау уақыты да артады.

3-кестеде әртүрлі ағындар саны бойынша openMP параллель орындаудың уақыты мен үдеуінің нәтижесі көрсетілген. Бұл орындау уақытын айтарлықтай жылдамдатты. OpenMP-дегі ең маңызды нүктелердің бірі - ағындар саны, ағындар санының өсуі үдеудің жоғарылауын білдірмейді. Кестеде нүктелер саны көбейген сайын 8 ағынның орындалу уақыты 2 және 4 ағындар санынан жылдамырақ екені байқалады.

4-кестеде процессорлардың әртүрлі санына байланысты MPI-дың параллель орындалу уақыты мен үдеуінің нәтижелері көрсетілген. Көріп отырғаныңыздай, процессор санының артуына байланысты, орындау уақыты басқаларға қарағанда жылдамырақ болды. Мысалы, екі процессор, 2048*2048

нүктелерінде, оның орындалу уақыты небәрі 2080,190s, ал сегіз процессор бірдей нүктелер санында 1482,456 секунд көрсетті, және біріншісіне қарағанда 1,7 есе жылдам болды. Бірақ бұл процессорлар санының өсуімен біз ең жақсы өнімділікке қол жеткіземіз дегенді білдірмейді [10]. Матрица өлшемі 48000x48000 болғанда, 1, 8, 64, 128, 256, 512 процессорлар арасында ең жақсы нәтижені 64 процессор көрсетті.

Кесте 1. Модель параметрлері

Параметрлер	Мән
K (абсолютті өткізгіштік)	0,001
судың тұтқырлығы	0,09
майдың тұтқырлығы	0.3
Кеуектілік	0.2
P_{inj}	0,5
P_{init}	0.3
P_{prod}	0.1
S_{inj}	1.0
S_{init}	0,001
Cp_{inj}	0.1
Cp_{init}	0,0
τ (Адсорбция коэффициенті)	0,0001

Кесте 2. Тізбектей орындау уақыты (қысым үшін)

Нүктелер саны	Орындау уақыты
2^{14}	6.501
2^{16}	27.985
2^{18}	120.383
2^{20}	536.222
2^{22}	2221.43

Кесте 3. OpenMP параллельді орындау уақыты (қысым үшін)

Нүктелер саны	Ағындар саны					
	2		4		8	
	Орындау уақыты	үдеуі	Орындау уақыты	үдеуі	Орындау уақыты	үдеуі
2^{14}	3,754	1.73	2.158	3.01	2.283	2.85
2^{16}	15.453	1.81	8.626	3.24	8.858	3.16
2^{18}	65.213	1,85	40.34	2.98	39.138	3.08
2^{20}	318.084	1.67	219.669	2.44	218.401	2.46
2^{22}	1304.57	1,70	905.26	2.45	903.229	2.46

Кесте 4. MPI параллельді орындау уақыты (қысым үшін)

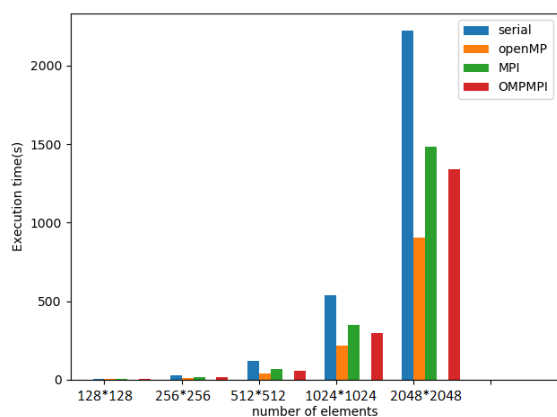
Нүктелер саны	Процессор саны					
	2		4		8	
	Орындау уақыты	үдеуі	Орындау уақыты	үдеуі	Орындау уақыты	үдеуі
2^{14}	8.296	0,78	4.699	1.38	4.951	1.31
2^{16}	33.132	0,85	17,639	1.59	17.543	1,60
2^{18}	184.476	0,65	60.072	2.00	68.368	1,90
2^{20}	588.710	0,91	348.358	1.54	349.006	1.54
2^{22}	2480.190	0,90	1491.049	1.49	1482.456	1,50

5 - кестеде ағындар мен процессорлардың әртүрлі санына байланысты openMP + MPI гибриді параллель орындалу уақыты мен үдеуінің нәтижелері көрсетілген. Орындау уақыты MPI модельмен салыстырғанда бір шама жоғары нәтижені көрсетті.

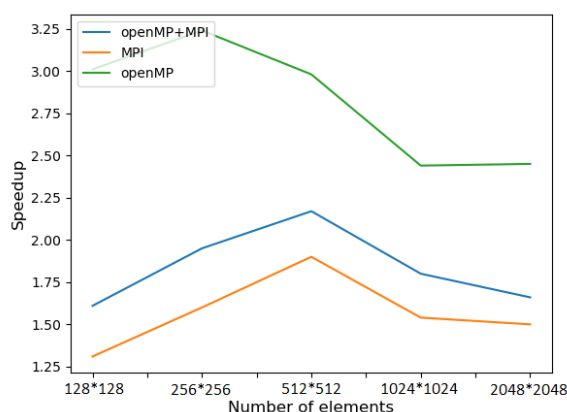
Кесте 5. OpenMP+MPI гибриді параллель орындау уақыты (қысым үшін)

Нүктелер саны	Процессор/ағындар саны			
	2/4		4/2	
	Орындау уақыты	үдеуі	Орындау уақыты	үдеуі
2^{14}	4.269	1.52	4.050	1.61
2^{16}	14.250	1.96	14.344	1.95
2^{18}	55.248	2.18	55.501	2.17
2^{20}	298.690	1,80	297.752	1,80
2^{22}	1350.159	1.65	1340.510	1.66

6(a) суретте әртүрлі параллельді есептеу модельдерінің орындалу уақытының салыстырылуы көрсетілген. 6(b) суретте әртүрлі параллельді есептеу модельдерінің үдеу мәндері салыстырылып көрсетілген.



Сурет 6(a). Параллель модельдердің есептеу уақыты



Сурет 6(b). Параллель модельдің үдеулері

Нүктелердің санының артуына байланысты бастапқы тізбекті үлгінің орындалу уақыты басқа параллельді модельдермен салыстырғанда тез өседі және әртүрлі параллельді есептеу модельдерінің үдеуі артады.

6-кестеде әрбір параллельді модельдеу технологияларының орындалу уақытының нәтижелерін салыстыру көрсетілген. Бір түйінде жұмыс жасағанда, OpenMP, MPI-ға қарағанда жақсы өнім корсеткенімен, бірақ көп түйіндер арасындағы есептеулерде әлсіздік танытады. Ал көп түйіндер арасындағы есептеулерде MPI өз ролын көрнектендіре түседі. Демек, OpenMP мен MPI-дың осы ерекшеліктерін лайықты біріктіре алсақ, параллельді есептеудың үдеуін сәйкесті арыттыруға әбден болады. Көріп отырғаныңыздай, OpenMP+MPI-дың ұқсамаған нүктелердегі нәтижесі MPI мен салыстырғанда әлде қайда жоғары.

Кесте 6. Параллель орындау уақытының нәтижелерін салыстыру (қысым үшін)

Нүктелер саны	Тізбектей	OpenMP	MPI	OpenMP+MPI
2^{14}	6.501	2.283	4.951	4.050
2^{16}	27.985	8.858	17.543	14.344
2^{18}	120.383	39.138	68.368	55.501
2^{20}	536.222	218.401	349.006	297.752
2^{22}	2221.43	903.229	1482.456	1340.510

Қорытынды

Бұл жұмыста сандық модель үшін Якоби әдісі қолданылды. Бір өлшемді Бакли-Леверетт математикалық моделі құрылды. Мұнайды ығыстыру есептерінің өнімділігін арттыру үшін тізбектелген және openMP, MPI, және openMP+MPI параллелді алгоритмдері құрылды. Үш түрлі стандарт бойынша параллелді бағдарламаларға 2^{14} - 2^{22} нүктелер саны бойынша тестілеу жүргізілді. Тізбектелген бағдарламаның әртүрлі нүктелер санында есептеу уақыттары алынды. Сонымен қатар, openMP, MPI және openMP+MPI (гибридті) параллелді бағдарламалары үшін әртүрлі нүктелер санында тестілеу жүргізіліп, есептеу уақыттары мен үдеулері алынды. Бұл нәтижелер 3-кестеде, 4-кестеде және 5-кестеде көрсетілген. Алынған нәтижелерге сәйкес, OpenMP, MPI және олардың гибриді формаларының есептеу уақыты тізбектелген бағдарламамен салыстырғанда арта түскені анықталды (6-кесте). Есептеу өнімділігі тек нүктелердің санына ғана байланысты емес, әрбір ағындардың, процессорлардың немесе ядролардың оңтайлы санына да байланысты.

Бұл зерттеуде OpenMP, MPI және гибриді модельдер қарастырылған қысым теңдеуін есептеуге кететін уақытын (CPU уақыты) айтарлықтай қысқартылғаны көрсетілді. Параллелді гибриді OpenMP+MPI моделінің есептеу уақыты MPI параллелді есептеумен салыстырғанда айтарлықтай айырмашылығы болмағанымен, нүктелер санының артуына байланысты күшейетіні анықталды.

Алғыстар

Бұл зерттеуді Қазақстан Республикасының ғылым және жоғары білім министрлігінің Ғылым комитеті қаржыландырды (Грант №AP09260564).

Пайдаланылған әдебиеттер тізімі:

- 1 Willhite G. P. (1986) *Waterflooding*
- 2 Buckley S.E. and Leverett M.C. (1942) *Mechanism of Fluid Displacements in Sands, Transactions of the AIME, 146, 107-116. <https://doi.org/10.2118/942107-G>*
- 3 Imankulov T., Lebedev D., Aidarov K. and Tura O. r (2014) *Design of HPC system for analysis the gel-polymer flooding of oil fields. Contemporary Engineering Sciences, vol. 7, p. 1531–1545. <http://dx.doi.org/10.12988/ces.2014.410187>*
- 4 Chhabra A., Thulasiraman P. Islam M. T. and Thulasiram R. K. (2004) *An OpenMP Implementation of FTCS Method for Reduced Black-Sholes Equation, in HPCS.*
- 5 Werneck L. F., Medeiros de Freitas M., Guaraldi da Silva Jr H., de Souza G. and Amaral Souto H. P. (2016) *An OpenMP parallel implementation for numerical simulation of gas reservoirs using Intel Xeon Phi coprocessor, in Proceedings of the XXXVII Iberian Latin-American Congress on Computational Methods in Engineering. ABMEC. DOI: <https://doi.org/10.26512/ripe.v2i21.21697>*
- 6 Ryanto G. P. H. and others (2017) *An OpenMP parallel godunov scheme for 1D two phase oil displacement problem, in 2017 5th International Conference on Information and Communication Technology (ICoICT). DOI:10.1109/ICoICT.2017.8074664*
- 7 Zhong H., Liu H., Cui T. Chen Z., Shen L., Yang B., He R. and Guo X. (2020) *Numerical simulations of polymer flooding process in porous media on distributed-memory parallel computers, Journal of Computational Physics, vol. 400, p. 108995. DOI:10.1016/j.jcp.2019.108995*
- 8 Alghamdi A. M., Eassa F. E., Khamakhem M. A., Al-Ghamdi A. S. A.-M., Alfakeeh A. S., Alshahrani A. S. and Alarood A. A. (2020) *Parallel hybrid testing techniques for the dual-programming models-based programs, Symmetry, vol. 12, p. 1555. DOI: <https://doi.org/10.3390/sym12091555>*
- 9 Jin H., Jespersen D., Mehrotra P., Biswas R., Huang L. and Chapman B. (2011) *High performance computing using MPI and OpenMP on multi-core parallel systems, Parallel Computing, vol. 37, 562–575. DOI: <https://doi.org/10.1016/j.parco.2011.02.002>*
- 10 Akhmed Zaki D. Zh., Lebedev D., Imankulov T. S., Kenzhebek Y., and d Kassymbek N. (2020) *High-performance and Intelligent Computational Models for Oil Production Problems, Journal of Computational Technologies, No 3.*