Компьютерная Математика

Инструментальные средства информационных технологий

УДК 004.67

А.В. ГРЕЧКО, А.О. МЕЛАЩЕНКО

ЗАСТОСУВАННЯ ГРІД-ТЕХНОЛОГІЙ ТА ГРАФІЧНИХ ПРИСКОРЮВАЧІВ ДЛЯ ДОСЛІДЖЕННЯ ПРОЦЕСІВ В МІНЕРАЛАХ

Мета роботи – адаптування і впровадження програм та обчислювальних комплексів для обчислень в умовах грід-інфраструктури при вирішенні задач наномінералогії та радіаційної мінералогії, пов'язаних з властивостями нанорозмірних мінеральних структур та з впливом радіаційного опромінення на руйнування кристалічної решітки мінералів. Для виконання роботи використана віртуальна організація (ВО) «GEOPARD», створена для розвитку застосування ІТ. Для вдосконалення пакетів прикладного програмного забезпечення (ППП3), що використовуються у дослідженнях відділення наук про Землю, застосовано графічні прискорювачі GPGPU.

Вступ. Сьогодні однією з основних складових науково-технічної роботи є необхідність залучення значних обчислювальних ресурсів (мільярди операцій у секунду) для проведення моделювання фізичних процесів. Надати відповідні ресурси без застосування суперкомп'ютерних технологій неможливо.

Отримання розглянутих у роботі результатів дослідження впливу особливостей структури, типу хімічного зв'язку та ізоморфізму на фізичні характеристики U-, Th-вміщуючих мінералів, швидкості аморфізації їх структур під дією автовипромінювання потребувало застосування теоретично розрахункових методів, таких як моделювання методом молекулярної динаміки (МД) та квантово-хімічних досліджень з використанням теорії функціоналу густини. Проведення МД моделювань з використанням відповідного програмного забезпечення вимагало розглянути 4 структури реальних кристалів, які налічують від 5 до 10 млн. атомів та виконати моделювання за різними потенціалами (2–3) та за різними температурними режимами (як правило п'яти). Оскільки кожний запуск складався з трьох етапів, то в результаті необхідно було здійснити близько 150 незалежних запусків. Для виконання кожного запуску необхідно було витратити в середньому 1000 процесоро-годин. Враховуючи значні обчислювальні ресурси, які необхідні для виконання досліджень, лише використання грід-технологій могло дозволити виконати заплановані дослідження у повному обсязі в рамках виконання проекту.

Результати досліджень дозволять обґрунтовано вибирати найбільш оптимальні матриці для утилізації високоактивних радіоактивних відходів. Передбачається, що такі матриці будуть зберігати кристалічну структуру на протязі довгого строку (десятки тисяч років), та що винос радіонуклідів у навколишнє середовище на протязі цього строку буде незначним і не становитиме загрози для екології регіону. Також будуть визначені умови формування карбонатфторапатитів з певними типами ізоморфних заміщень і нанокластерами в структурі, що має важливе значення при дослідженнях закономірностей формування родовищ, які містять фосфатні і силікатні мінерали, і створенні синтетичних аналогів апатитових структур.

Постановка задачі. Для виконання досліджень впливу особливостей структури, типу хімічного зв'язку та ізоморфних заміщень на фізичні характеристики мінералів та швидкість аморфізації їх структури під дією автовипромінювання застосувати, налаштувати та сконфігурувати у грід-середовищі необхідне програмне забезпечення (ПЗ), що реалізує теоретичні розрахункові методи, такі як моделювання методами молекулярної динаміки (МД) і напівемпіричних атомних потенціалів, квантово-хімічні дослідження методом Хартрі – Фока та квантово-хімічні дослідження з використанням теорії функціоналу густини. До складу цього ПЗ входить:

– GULP – пакет, що дозволяє проводити теоретичні розрахунки можливих структур і фізичних властивостей матеріалів та мінералів;

– DL_POLY – паралельний програмний пакет для дослідження процесів, що відбуваються у структурі мінералів методом молекулярної динаміки;

– Quantum Espresso – інтегрований пакет з відкритим кодом для розрахунків електронної структури матеріалів з перших принципів;

– Abinit – інтегрований пакет для розрахунків повної енергії, густини заряду та електронної структури матеріалів з перших принципів.

Для виконання роботи використати віртуальну організацію (ВО) «GEOPARD» [1], створену Інститутом кібернетики імені В.М. Глушкова НАН України, Інститутом геохімії, мінералогії та рудоутворення ім. М.П. Семененка НАН України та Інститутом геофізики ім. С.І. Субботіна НАН України для розвитку застосування інформаційних технологій у сфері наук про Землю.

Застосування DL_POLY для моделювання структур. На першому етапі налагодження застосовувалося скомпільоване паралельне програмне забезпечення DL_POLY_3 [2] з використанням компілятора Intel Fortran Compiler 11 та бібліотек паралельного програмування MPICH. Також здійснено спробу застосувати нову версію DL_POLY_4, скомпільовану з використанням NVIDIA CUDA (з підтримкою графічних прискорювачів), але при моделюванні різних структур встановлено невідповідність обчислених значень еталонним результатам. Враховуючи цей факт, при подальшому моделюванні застосовано версія DL_POLY_3 без NVIDIA CUDA.

На наступному кроці структуровано вхідні дані та вихідні результати. Для моделювання фахівцями Інституту геохімії подано дві різні молекулярні структури, кожна з яких моделювалася при трьох різних енергіях у 4 режимах, отже всього отримано 24 комбінації.

Для організації моделювання, окрім конфігурування грідів під ПЗ, прийнято наступну структуру. Каталог програми для запуску містить 2 директорії, що відповідають кожній структурі. Кожна з вказаних директорій містить підкаталоги для кожної енергії з трьох (5, 10 і 15), в яких розташовані по 4 підкаталоги для кожного з 4 режимів і два скриптові файли report.sh (лістинг 1) та view.sh (лістинг 2). Останні відповідають за агрегацію інформації щодо проведених розрахунків.

Лістинг 1. Код файлу report.sh

```
#/bin/sh
find . -name STATIS -exec `pwd`/view.sh {} \;
```

Лістинг 2. Код файлу status.sh

```
#!/bin/sh
echo $1 `tail -n 15 $1 | grep '\<[0-9]\{5,\}\>'`
```

У свою чергу кожен підкаталог, що відповідає одному з чотирьох режимів, містить окремі каталоги для кожної структури і скриптовий файл run.sh (лістинг 3), а у каталозі для кожної структури розташовані скриптовий файл start.sh (лістинг 4) і системні файли DL_POLY.

Лістинг 3. Код файлу run.sh

```
#!/bin/sh
P=`pwd`
STR=`basename $P`
for D in `ls -d */`
do
cd $D
sbatch --mail-type=ALL --mail-user=javatask@ukr.net -J
"ti_""$STR""_""$D" --ntasks-per-node=8 -N 2 -p scit3
$PWD"/start.sh"
cd ..
done
```

Лістинг 4. Код файлу start.sh

```
#!/bin/sh
export
LD_LIBRARY_PATH=/home/users/alan/devel/mpich/lib:/home/users/al
an/intel/lib/intel64:$LD_LIBRARY_PATH
/home/users/alan/devel/mpich/bin/mpirun -env LD_LIBRARY_PATH
/home/users/alan/devel/mpich/lib:/home/users/alan/intel/lib/int
el64:$LD_LIBRARY_PATH $PWD/DLPOLY
```

Встановлення та налаштування пакету GULP для виконання комп'ютерного моделювання. GULP – програма для виконання багатьох типів симуляцій матеріалів, використовуючи граничні умови 0-D (молекули і кластери), 1-D (полімери), 2-D (поверхні, пласти і міжзернові границі), 3-D (періодичні породи). Фокус коду знаходиться у аналітичному рішенні через використання динаміки решіток, де можливо, а не молекулярної динаміки. Різноманітність силових полів може використовуватись у GULP, охоплюючи оболонкову модель для матеріалів, що містять іони, молекулярної механіки для органічних систем,

вбудовану модель атому для металів і реактивний потенціал REBO для вуглеводнів. Аналітичні похідні включені до щонайменше другого порядку для більшості силових полів і до третього порядку для багатьох з них [3].

Вхідні дані надходять до адміністратора у вигляді файлів *.gin. Програмний пакет GULP за своїм характером не паралельний, тобто для його використання у гріді необхідно запускати багато його екземплярів і обробляти одночасно різні структури. На вхід на момент написання звіту прийшло 197 .gin-файлів, кожний з яких обровлявся в середньому 8 годин, а отже всього витрати часу мали б скласти близько 1579 годин або 66 діб при розрахунку на персональному комп'ютері без врахування часу на запуски та нештатні ситуації. Використання грід-технологій значно прискорює цей процес, оскільки робочий комп'ютер фахівця Інституту геохімії обробив би всі ці файли за 328 діб (у 5 раз повільніше) через свої низькі технологічні характеристики без врахування витрат на роботу оператора, запуск і спостерігання. Отже без використання грід-технологій неможливо виконати повний цикл досліджень в рамках календарного року.

Для розпаралелювання задачі користувач гріду направляє структури на окремий процесор гріду, на якому запускається екземпляр пакету GULP, на виході буде отримано файл з розширенням .gout, який направляється грідом назад до користувача.

Схема організації обчислень з використанням пакета GULP показана на рис. 1.

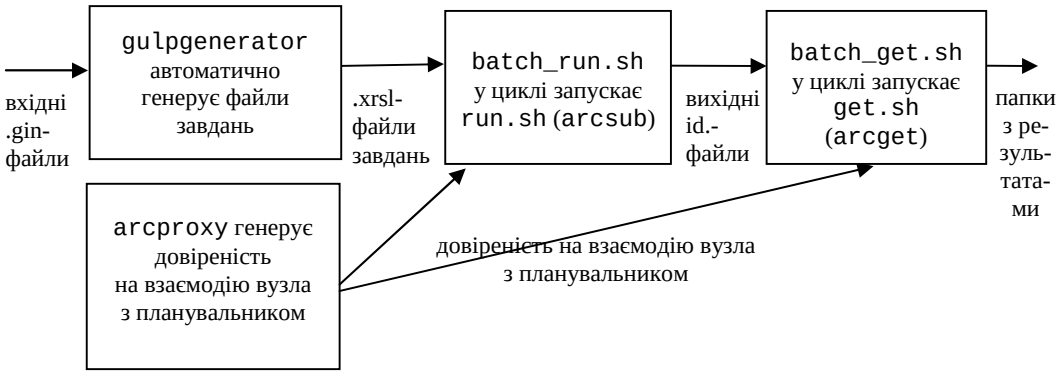

РИС. 1. Схема організації обчислень з використанням пакета GULP

Вихідні коди скриптів запуску та отримання результатів з застосуванням грід наведено у лістингах 5 – 9.

Лістінг 5. Вихідні коди gulpgenerator

```
package gulpgenerator;
import java.io.*;
/**
 *
 * @author alan
 */
```

```
   public class GulpGenerator {
       /**
        * @param args the command line arguments
        */
       public   static   void   main(String[]   args)   throws
IOException {
           String str = "&(executable=\"gulp\") \n" +
                        "(arguments=\"#\") \n" +
                         "(inputfiles= \n"+
                       "(\"#.gin\" \"#.gin\") \n" +
                        "(\"gulp\" \"gulp\") \n" +
                       ") \n" +
                       "(outputfiles= \n"+
                       "(\"#.gout\" \"#.gout\") \n"+
                        ") \n"+
                       "(cpuTime=\"4 hour\") \n"+
                       "(jobname=geo_gulp) \n"+
                       "(stdout=stdout.txt) \n"+
                       "(stderr=stderr.txt)";
           for (int i = 1; i <= 108; i++){
           File name = new File (String.valueOf(i)+".xrsl");
                   FileWriter fw = new FileWriter(name);
                   fw.write(str.replace("#",
String.valueOf(i)));
                   fw.close();
                   }    }}
```

Лістинг 6. Вихідні коди batch_run.sh

```
#!/bin/sh
for f in *.xrsl
do
echo "launcing $f"
`pwd`/run.sh $f
done
```

Лістинг 7. Вихідні коди run.sh

```
#!/bin/sh
arcsub -c uagrid.org.ua -f $1 -o $1.id
```

Лістинг 8. Вихідні коди batch_get.sh

```
#!/bin/sh
for f in *.id
do
echo "getting $f"
`pwd`/get.sh $f
done
```

Лістинг 9. Вихідні коди get.sh

```
#!/bin/sh
arcget -i $1 -s FINISHED
arcget -i $1 -s FAILED
```

На першому етапі користувач за допомогою програмного забезпечення gulpgenerator для кожного вхідного .gin-файла автоматично генерує файл завдання з тією ж назвою, але з розширенням xrsl, а також встановлює обмеження на час виконання розрахунків (параметр cpuTime) для уникнення зациклення при виконанні обчислень. Окрім параметрів виконання, вказується адреса електронної пошти, на яку надходять повідомлення про початок виконання програми та про факт її успішного або невдалого завершення. Приклад вмісту .xrsl-файлу завдання наведено у лістингу 10.

Лістинг 10. Приклад xrsl.-файлу завдання, автоматично згенерованого для вхідного файлу 1.gin

```
&(executable="gulp")
(arguments="1")
(inputfiles=
("1.gin" "")
("gulp" "")
)
(outputfiles=
("1.gout" "")
)
(cpuTime="8 hour")
(jobname=geo_gulp_job_6_1)
(notify="be javatask@ukr.net"
)(stdout=stdout.txt)
(stderr=stderr.txt)
```

Програма arcproxy генерує довіреність на взаємодію вузла з планувальником.

Спеціально розроблений для організації розрахунків скрипт batch_run.sh у циклі для кожного xrsl.-файлу завдання, що міститься у папці, запускає run.sh.

Скріпт run.sh за попередньо згенерованою довіреністю на взаємодію з планувальником запускає arcsub, яка ставить відповідне завдання у чергу завдань на виконання. Результати обчислень виводяться у файл з розширенням .id.

Після виконання розрахунків запускається скрипт batch_get.sh, який у циклі для кожного .id-файлу результатів, що міститься у папці, запускає get.sh.

Скріпт get.sh за попередньо згенерованою довіреністю на взаємодію з планувальником запускає arcget, яка генерує папки з результатами обчислень у відповідності з тим, були вони успішно виконані чи ні.

Результати роботи скриптів показані на рис. 2.

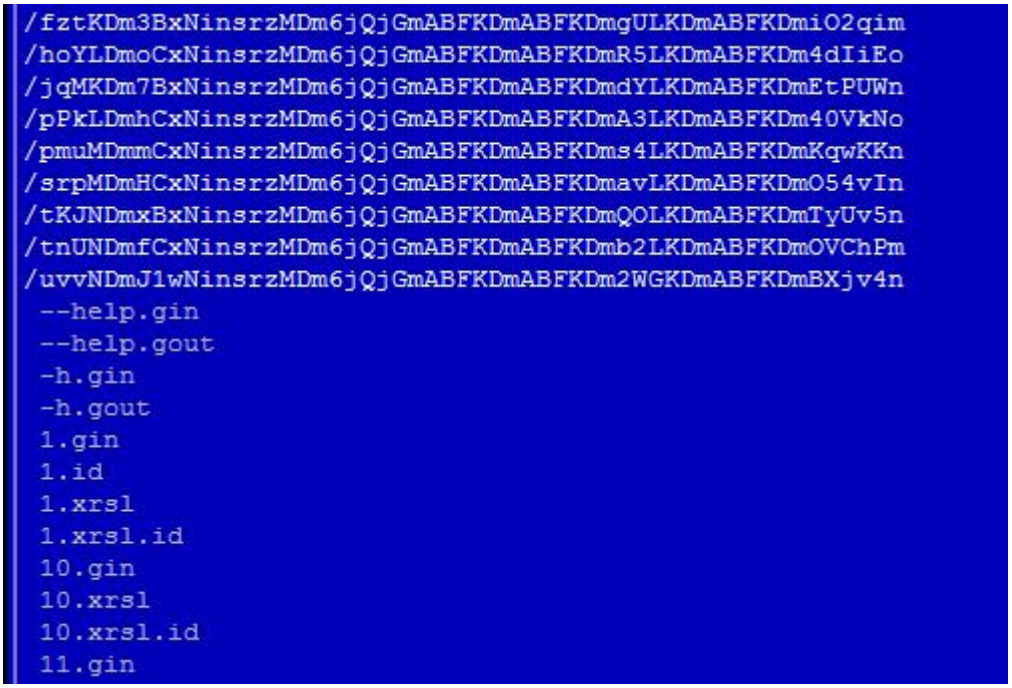

РИС. 2. Результати роботи скриптів

Встановлення та налаштування інтегрованого пакета Quantum Espresso з застосуванням GPGPU. Quantum Espresso – це інтегрований пакет з відкритим кодом для обчислень електронної структури і моделювання матеріалів у нано-масштабі. Його засновано на теорії функціоналу густини, плоских хвилях і псевдопотенціалах.

Вимоги до програмного та апаратного забезпечення. Рекомендується використовувати:

- компілятор Intel (версії вище 11);
- GNU GCC / GFORTRAN (версії вище 4.1, але нижче 4.6);
- карти NVIDIA (обчислювальною здатністю > = 2.0);
- CUDA 4.x або CUDA 5.0;
- MKL (версії > = 10.x) або ACML (> = 4.4.x, > = 5.x);
- Python > = 2.4.x;
- Quantum ESPRESSO 5.0.2 [4, 5].

Мінімальні вимоги:

- GNU GCC / GFORTRAN (версії вище 4.1 , але нижче 4.6);
- карти NVIDIA (продуктивність > = 1.3);
- CUDA 4.x або CUDA 5.0;
- Python > = 2.4.x;
- Quantum ESPRESSO 5.0.2.

Код протестований на 64-бітних системах. Для компіляції GPU версії Quantum Espresso використано останні версії Quantum Espresso та його GPU розширення, які було отримано з сервера вихідних кодів за адресою http://qeforge.qe-forge.org/svn/q-e/trunk/espresso.

Компіляція. Для виконання компіляції:

```
$ cd GPU
$ ./configure --disable-parallel --enable-openmp \
  --enable-cuda --with-gpu-arch=35 \
  --with-cuda-dir=/opt/cuda5.0   --enable-magma   --enable-
phigemm
$ cd ..
$ make -f Makefile.gpu all-gpu
```

Виконувані файли "*-gpu.x" знаходитимуться у папці bin/ (наприклад, "pw-gpu.x").

Додаткові опції для QE-GPU:

- enable-cuda : включає підтримку GPU;

-with-cuda-dir = <path > : вказує шлях до папки, в якій встановлена CUDA (обов'язковий);

-with-gpu-arch = < arch >: вказує обчислювальну здатність GPU. QE-GPU підтримує cc_13, cc_20, cc_21, cc_30, cc_35 (значення за замовчуванням – 20);

-enable-phigemm : включає прискорення BLAS Level 3 GPU за допомогою phiGEMM [6] (значення за замовчуванням – yes);

-enable-magma : включає прискорення GPU LAPACK на основі MAGMA 1.3.0 (значення за замовчуванням – yes);

-enable-fast-cuda: ця опція включає спеціальну оптимізацію. Використовується ТІЛЬКИ, якщо можна безпечно запустити тест у нормальних умовах (значення за замовчуванням – no);

-enable-pinned-mem : ця опція включає виділення закріпленої фізичної пам'яті (pinned memory) для швидкої H2D-D2H передачі даних (значення за замовчуванням – no).

--enable-profiling : включає профіль phiGEMM виклик за виклик (значення за замовчуванням – no);

-enable-debug-cuda : включає в стандартний вивід детальні повідомлення про налагодження;

-enable-openacc : включає директиви OpenACC (тільки PGI).

Для визначення найкращої комбінації прапорців для конкретної системи пропонується використовувати наступні 4 конфігурації, виконавши моделювання на один-два кроки перш, ніж виконувати усе моделювання повністю:

1) --enable-phigemm --enable-magma;

2) --enable-phigemm --enable-magma --enable-fast-cuda;

3) --enable-phigemm --enable-magma --enable-pinned-mem;

4) --enable-phigemm --enable-magma --enable-pinned-mem --enable-fast-cuda.

Найкраща конфігурація програмного забезпечення для паралельних обчислень на невеликому кластері, що складається з декількох гібридних вузлів:

- CUDA 5.0;
- Intel Compiler (існує безкоштовна версія під академічною ліцензією);
- Intel MKL (будь-яка версія вище 10);
- Open MPI.

Найкраща апаратна конфігурація:

- карти NVIDIA з обчислювальною здатністю 2.x (C20xx, M20xx) або 3.5 (K20/K20x):
- достатньою кількостю пам'яті на карті є 3 Гбайта;
- для систем з менш ніж 50 атомів, як правило, достатньо 1,5 Гбайт.

Запуск. При застосуванні GPU обов'язково слід вказати кілька додаткових змінних середовища, необхідних бібліотеці xphiGEMM, використовуваній Quantum Epsresso.

Є 4 змінні, що розпізнаються бібліотекою, реалізовані по одній для кожної процедури *GEMM. Коефіцієнт, значення якого більше 0 і менше 1, вказує бібліотеці, скільки обчислень виконувати на CPU і скільки на GPU. Наприклад, якщо PHI_DGEMM_SPLIT дорівнює 0,9, це означає, що 95 % обчислень буде виконано на GPU, а 5 % – на CPU.

Приклад:

```
export PHI_SGEMM_SPLIT=0.95;
export PHI_CGEMM_SPLIT=0.875;
export PHI_DGEMM_SPLIT=0.9;
export PHI_ZGEMM_SPLIT=0.925.
```

Встановлення та налагодження пакета Abinit для виконання комп'ютерного моделювання. Пакет Abinit дозволяє знаходити сумарну енергію, густину заряду і електронну структуру системи з електронів і ядер у межах теорії функціоналу густини, використовуючи псевдо потенціали, плоскі хвилі чи вейвлетний базис. Abinit також включає опції для оптимізації геометрії у відповідності до сил і коливань теорії, або для здійснення симуляцій молекулярної динаміки з використанням цих сил, або для генерації динамічних матриць, ефективних зарядів Борна, діелектричних тензорів, базованих на теорії збурень функціоналу густини тощо. Збуджені стани можуть розраховуватись у межах теорії збурень багатьох тіл (апроксимація GW і рівняння Бете – Солпітера) і часової теорії функціоналу густини (для молекул). Крім головного коду пакета Abinit, додаються ще допоміжні утиліти [9].

Висновки. Дослідження радіаційної стійкості мінералів методом молекулярної динаміки виконано у два етапи: 1) підготовка задач до запуску у грід шляхом автоматичної генерації файлів завдань та 2) виконання моделювання у грід-середовищі.

В рамках дослідження розглянуто наступні структури у розрізі пакетів моделювання:

• для моделювання з застосуванням ППП3 DL_POLY_3 моделювалися 2 структури реальних кристалів, що налічують від 6 до 20 млн. атомів, при трьох різних енергіях та у чотирьох конфігураціях. Тобто проведено 24 експерименти з частковим використанням грід-технологій;

• для моделювання з застосуванням ППП3 Quantum ESPRESSO моделювалися 5 структур реальних кристалів із застосуванням як CPU, так і GPU+CPU. Тобто проведено 10 експериментів із частковим використанням грід-технологій;

• для моделювання з застосуванням ППП3 GULP моделювалися 197 структур реальних кристалів. Тобто проведено 197 експериментів із використанням грід-технологій.

Проведення вказаних експериментів без застосування грід-технологій вимагало б 1096 діб для розрахунку на персональному комп'ютері, тобто застосування грід-технологій є критично важливим ресурсом для виконання досліджень.

А.В. Гречко, А.О. Мелащенко

ПРИМЕНЕНИЯ ГРИД-ТЕХНОЛОГИЙ И ГРАФИЧЕСКИХ УСКОРИТЕЛЕЙ ДЛЯ ИССЛЕДОВАНИЯ ПРОЦЕССОВ В МИНЕРАЛАХ

Цель работы – адаптация и внедрение программ и вычислительных комплексов для вычислений в условиях грид-инфраструктуры при решении задач наноминералогии и радиационной минералогии, связанных со свойствами наноразмерных минеральных структур и с воздействием радиационного облучения на разрушение кристаллической решетки минералов. Для выполнения работы использована виртуальная организация (ВО) «GEOPARD», созданная для развития применения ИТ. Для совершенствования пакетов прикладного программного обеспечения (ППП3), используемого в исследованиях отделения наук о Земле, применены графические ускорители GPGPU.

A.V. Hrechko, A.O. Melashchenko

APPLICATIONS OF GRID TECHNOLOGIES AND GRAPHIC ACCELERATORS TO THE RADIATION-STIMULATED PROCESSES INVESTIGATION, PHASE TRANSFORMATIONS, AND ISOMORPHIC SUBSTITUTIONS IN MINERALS

The purpose of the work is to adapt and implement the programs and computing complexes for calculations under the grid infrastructure in solving the problems of nanomineralogy and radiation mineralogy associated with the properties of nanosized mineral structures and the effect of radiation irradiation on the destruction of the crystal lattice of minerals. To carry out the work, the virtual organization (GE) "GEOPARD", created to develop the IT application, was used. Graphic accelerators of the GPGPU are used to improve the software application packages (FPGAs) used in the Earth Sciences research.

Список літератури

1. http://voms.grid.org.ua/voms/
2. The DL_POLY Molecular Simulation Package. [Електронний ресурс]. Режим доступу: http://www.stfc.ac.uk/CSE/randd/ccg/software/DL_POLY/25526.asp;
3. GULP. [Електронний ресурс] – Режим доступу: http://projects.ivec.org/gulp/;
4. Giannozzi P. et al., "QUANTUM ESPRESSO: a modular and open-source softwareproject for quantum simulations of materials". J. Phys. Condens. Matter 21 395502 (2009).
5. Quantum Espresso. [Електронний ресурс]. Режим доступу: http://www.quantum-espresso.org/?page_id=42.
6. Spiga F. and Girotto I., "phiGEMM: a CPU-GPU library for porting Quantum ESPRESSO on hybrid systems", 20th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP). 2012.
7. Xianyi Z., Qian W., Yunquan Z. "Model-driven Level 3 BLAS Performance Optimization on Loongson 3A Processor", 2012 IEEE 18th International Conference on Parallel and Distributed Systems (ICPADS). 17 – 19 Dec. 2012.
8. OpenBLAS URL: http://xianyi.github.com/OpenBLAS/
9. Abinit: The tutorials. [Електронний ресурс]. Режим доступу: http://www.abinit.org/documentation/helpfiles/for-v6.12/tutorial/welcome.html.

Одержано 12.10.2017

Про авторів:

Гречко Анастасія Валеріївна,
кандидат фізико-математичних наук,
науковий співробітник Інституту кібернетики імені В.М. Глушкова НАН України,
E-mail: grechna@gmail.com

Мелащенко Андрій Олегович,
кандидат фізико-математичних наук,
старший науковий співробітник Інституту кібернетики імені В.М. Глушкова НАН України.
E-mail: javatask@gmail.com