

## РОЗВИТОК ІНТЕРНЕТ-ПОРТАЛУ МЕТЕОРОЛОГІЧНОГО ПРОГНОЗУВАННЯ НА МУЛЬТИПРОЦЕСОРНІЙ ПЛАТФОРМІ

*О.Г. Бекетов, Є.А. Вітряк, І.О. Мироненко, О.М. Овдій*

Розглянуто подальший розвиток сервісно-орієнтованого портального рішення для надання послуг метеорологічного прогнозування. Зокрема, виконано реалізацію тривимірного алгоритму вирішення задачі моделювання циркуляції атмосфери з використанням графічних прискорювачів в якості обчислювального пристрою як для чисельного розрахунку прогнозу, так і для його візуалізації.

Ключові слова: паралельні обчислення, метеорологічне прогнозування, інтернет-портал, мережа, графічний прискорювач, візуалізація.

Рассматривается дальнейшее развитие сервисно-ориентированного портального решения для предоставления услуг метеорологического прогнозирования. В частности, выполнено реализацию трехмерного алгоритма решения задачи моделирования циркуляции атмосферы с использованием графических ускорителей в качестве вычислительного устройства как для численного расчета прогноза, так и для его визуализации.

Ключевые слова: параллельные вычисления, метеорологическое прогнозирование, интернет-портал, сеть, графический ускоритель, визуализация.

The article deals with the further development of a service-oriented portal solution for providing meteorological forecasting services. In particular, the three-dimensional algorithm for solving the task of circulation atmosphere modeling was implemented using graphics processing units (GPU) as a computing device for forecast calculation as well as for its visualization.

Key words: parallel computing, meteorological forecasting, web portal, web, GPU, visualization.

### Вступ

В рамках попередньої роботи було спроектовано та реалізовано сервісно-орієнтовану портальну систему для надання послуг метеорологічного прогнозування на мультипроцесорній платформі [1]. Систему було створено за допомогою сучасних провідних технологій, таких як інтернет-портали та мікро-сервіси.

Поточна робота продовжує розвиток цієї системи шляхом реалізації більш складних алгоритмів вирішення задачі метеорологічного прогнозування на новій мультипроцесорній платформі на базі відеографічних прискорювачів. Вона покликана наочно продемонструвати перевагу використання такої технічної бази в порівнянні з центральним процесором для задачі обрахунку метеорологічного прогнозу.

Метеорологічне прогнозування окрім задоволення потреб звичайних користувачів використовується для вирішення різноманітних задач у транспортній галузі, зокрема авіації, сільському господарстві, наукових досліджень тощо. Суттєве значення має актуальність прогнозу, його достовірність та оперативність отримання. Швидкість отримання результатів прогнозу в деяких ситуаціях має критичне значення.

Задача метеорологічного прогнозування вимагає обчислень над великими масивами даних та потребує впровадження паралельних реалізацій та оптимізації обчислювальних алгоритмів.

Сервіс чисельного обрахунку здійснює обробку вихідних метеорологічних даних для потрібної користувачу території та на заданому проміжку часу. В свою чергу сервіс, що відповідає за візуалізацію результатів прогнозу, відображає отримані чисельні результати у графічному вигляді на географічній карті. Візуалізація результатів прогнозу також здійснюється за допомогою відеографічного прискорювача.

Структура роботи є наступною: перший розділ містить постановку задачі та опис математичної моделі; у розділі 2 висвітлюються архітектурні аспекти побудованої системи; у розділі 3 представлено деталі реалізації візуалізації даних метеорологічного прогнозу.

### 1. Задача метеорологічного прогнозування

В залежності від просторової протяжності всі процеси та явища, що відбуваються в атмосфері, поділяються на три типи: мікромасштабні (до кілометра), мезомасштабні (порядку десятків і сотень кілометрів) та макромасштабні (понад 1000 км). Регіональні та локальні моделі, що описують мезомасштабні та мікромасштабні метеорологічні процеси та явища, визначають формування погоди в окремих пунктах та регіонах, призначені для створення деталізованих в просторі та часі прогнозів погоди [2–5].

Математична модель стану атмосфери виражається рівняннями руху Рейнольдса для турбулентних процесів, що зв'язують прискорення в певному напрямку з компонентами об'ємних і поверхневих сил, які діють у цьому напрямку, рівнянням збереження маси, термодинамічним рівнянням енергії і рівнянням стану Шарля-Бойля. Рівняння, з яких складається модель, є переважно нелінійними тривимірними рівняннями другого порядку в частинних похідних з малим параметром при похідних другого порядку, і їх розв'язання здійснюється за допомогою чисельних методів на дискретній просторово-часовій сітці.

© О.Г. Бекетов, Є.А. Вітряк, І.О. Мироненко, О.М. Овдій, 2016

Це обумовлено складністю математичних моделей, що описують циркуляційні процеси в атмосфері. Через наявність малого параметра при старших похідних ці рівняння можуть змінювати свій тип (еліптичний, гіперболічний або параболічний) залежно від режиму руху в атмосфері, що моделюється. Крім цього треба відзначити, що математична модель циркуляції атмосфери та процесів в ній, яка сформульована на базі повних рівнянь Нав'є-Стокса та тепло-масопереносу, описує широкий спектр рухів: мезомасштабні процеси, баротропні та бароклініні хвилі Росбі, інерційно-гравітаційні хвилі, квазістаціонарні струминні течії.

Просторова дискретизація з грубим розподіленням веде до того, що вплив багатьох мезомасштабних метеорологічних процесів втрачається, оскільки у такому разі вони опиняються у класі процесів підсіткового масштабу. Ресурси сучасних ЕОМ поки що не дають змоги здійснювати просторову дискретизацію моделі атмосфери з високим розподіленням. Крім того, значною проблемою при прогнозуванні погоди на основі реалізації математичних моделей є обмеження на час розв'язання, оскільки прогноз погоди потрібно отримувати завчасно. Навіть сучасні потужні обчислювальні машини не дозволяють використовувати дрібну просторову сітку в області розв'язання задачі і часову дискретизацію та ітераційні методи, що дало б можливість підвищити точність прогнозу.

Стан атмосфери в просторі  $r = (\lambda, \varphi, \sigma)$  макромасштабної області  $G(r) \subset \bar{G}(r)$  визначається вектором  $\mathfrak{R}(r, t) = (u, v, w, \pi, T, q, q_L, q_W, k, \varepsilon)$  дискретних значень аналізу і прогнозу  $\mathfrak{R}(r, t^m) = \mathfrak{R}^m(r)$ , отриманих на основі макромасштабної моделі в моменти часу  $t = t^m$  ( $m=0, 1, \dots, M$ ) з кроком  $\tau = t^m - t^{m-1}$ , де  $t$  – час;  $\lambda$  – довгота;  $\varphi$  – широта;  $\sigma$  – приведена вертикальна координата, що відслідковує рельєф;  $u, v, w$  – складові швидкості вітру по довготі, широті і вертикальній координаті відповідно;  $\pi = C_p(p/p_0)^{R/C_p}$  – приведений тиск;  $T$  – температура;  $q$  – вологість,  $q_H$  – вологість у стані насичення;  $q_L$  – вода у рідкому стані;  $k$  – турбулентна кінетична енергія;  $\varepsilon$  – турбулентна дисипація. Для визначення стану атмосфери на обмеженій території  $\bar{G}$  при  $\forall t \in [t^{m-1}, t^m]$  виконується розв'язання задачі, що у векторному виразі має вигляд:

$$\frac{\partial \mathfrak{R}}{\partial t} = D\mathfrak{R}, \quad \forall t \in [t^{m-1}, t^m], \quad \forall r \in \bar{G}, \quad (1)$$

$$\mathfrak{R}(r, t^m) = \mathfrak{R}^m(r), \quad m = 0, 1, \dots, M.$$

Для розв'язання задачі (1) в роботі застосовується інтерполювання функції за декількома значеннями самої функції (передісторією) та похідних від неї першого порядку за допомогою алгебраїчного поліному. Використовується інтерполяція з кратними вузлами при кратності три. Виконується також апроксимація правих частин диференціальних рівнянь моделі у вузлах просторової сітки. Детальний опис застосованого чисельного методу наведено в монографії [2].

З метою підвищення точності прогнозу, що вимагає подрібнення просторової сітки в області розв'язання задачі і часу дискретизації, було реалізовано систему прогнозування [6, 7], що використовує графічний прискорювач та програмно-апаратну технологію паралельних обчислень CUDA [8], яка дозволяє істотно підвищити обчислювальну продуктивність завдяки використанню графічних процесорів. Відеографічний процесор дозволив суттєво прискорити процес отримання результатів прогнозу, у порівнянні із центральним процесором без використання прискорювача.

В рамках даної роботи з метою надання публічного доступу до користування цими розробками, було створено сервісно-орієнтований портал, що дозволяє користувачу отримати доступ до локального прогнозу погоди в межах території України.

Початкові дані об'єктивного аналізу стану атмосфери (тиск, температура, вологість та горизонтальні компоненти вітру) для території Європи надано німецькою національною службою DWD Offenbach [9]. Сервіс чисельного обчислення здійснює обробку вихідних метеорологічних даних для потрібної користувачу території та на заданому проміжку часу. В свою чергу сервіс, що відповідає за візуалізацію результатів прогнозу, відображає отримані чисельні результати у графічному вигляді на географічній карті. Візуалізація результатів прогнозу також здійснюється за допомогою графічного прискорювача.

## 2. Архітектура системи

Поточна робота продовжує розвиток реалізованої раніше сервісно-орієнтованої системи метеорологічного прогнозування [1] шляхом додавання нового модуля, що реалізує задачу розрахунку метеорологічного прогнозу на базі відеографічних прискорювачів. Окрім того було обрано новий метод для надання користувачу кінцевого результату прогнозування – статичне зображення карти з нанесеними на неї позначками, сформоване також за допомогою відеографічного процесора. Ця частина роботи буде докладніше розглянута у наступному розділі.

Потрібно зазначити, що сервісна орієнтованість розроблюваної системи розглядається як два різних поняття. По-перше – це сервісна орієнтованість щодо користувача. Тобто, система спрямована на створення максимально зручних умов для надання послуг («сервісів») метеорологічного прогнозування кінцевим користувачам. В даному випадку під сервісами розуміємо описані вище задачі метеорологічного прогнозування.

З іншого боку мається на увазі сервісно-орієнтована архітектура розроблюваної системи. Її особливість полягає в розділенні загальної функціональності «програми» на окремі модулі, які працюють максимально незалежно один від одного, є взаємозамінними та розподіленими. Зв'язок між ними підтримується за рахунок стандартизованих інтерфейсів, через які ці програмні модулі здатні спілкуватись між собою. Перевагою такої архітектури є гнучкість та простота додавання нових модулів («веб-сервісів») у систему.

Продовжуючи ідею модульності, було спроектовано поточну архітектуру системи, схему якої наведено на рис. 1. Як видно зі схеми, система поділена на такі функціональні сервіси: портал, сервіс управління метеорологічними прогнозами та сервіс надання метеорологічних прогнозів на GPU, який, в свою чергу, зв'язаний з програмами для розрахунку та відображення метеорологічних даних. При цьому, вважається, що усі сервіси фізично відокремлені. Виходячи з цього, розробка цих сервісів виконується таким чином, щоб вони були здатні працювати як одне ціле, будучи розділеними мережею Інтернет.

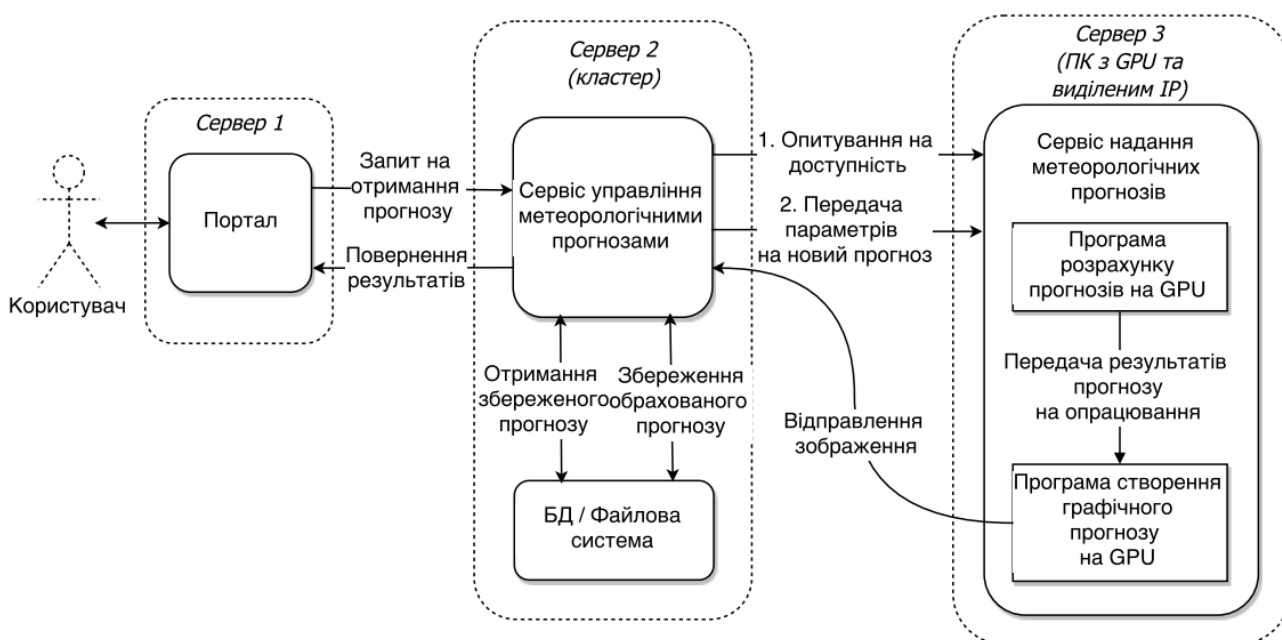


Рис. 1. Схема розроблюваної системи метеорологічного прогнозування

У даній системі вирішено використовувати REST-сервіси [10] замість SOAP-сервісів [11]. Це обумовлено необхідністю передавати різні формати даних та зменшити розмір повідомлень.

Слід зазначити, що увесь «бекенд» системи розробляється мовою Java з використанням фреймворку Spring [12]. Він дозволяє значно пришвидшити розробку веб-додатків та зробити їх більш модульними завдяки принципу «впровадження залежностей», що покладений в його основу. У якості програмних серверів для сервісів обрано контейнер сервлетів Apache Tomcat.

Інтернет-портал реалізований за допомогою відкритого програмного продукту Liferay [13]. Вибір саме порталного рішення зумовлений можливістю легко інтегрувати різноманітні сервіси та забезпечувати єдину точку доступу до них, поступово нарощуючи функціональність системи. Сервіси всередині порталу розміщуються в окремих модулях, що називаються портлетами.

Сервіс надання метеорологічних прогнозів поділяється на декілька функціональних блоків, його схема зображена на рис. 2.

«Фронт-сервлет» – об'єднує обробку запитів шляхом їх направлення через єдиний об'єкт-обробник. Цей об'єкт реалізує загальну поведінку, яка може бути змінена під час виконання за допомогою шаблону декоратор. Після цього «Фронт-сервлет» створює потрібні об'єкти відповідно до запиту та викликає методи для реалізації конкретної задачі. Запит по створенню задачі на прогноз повертає орієнтовний час очікування

та ідентифікатор замовлення, через який можна буде отримати будь-яку інформацію про замовлення та його результат.

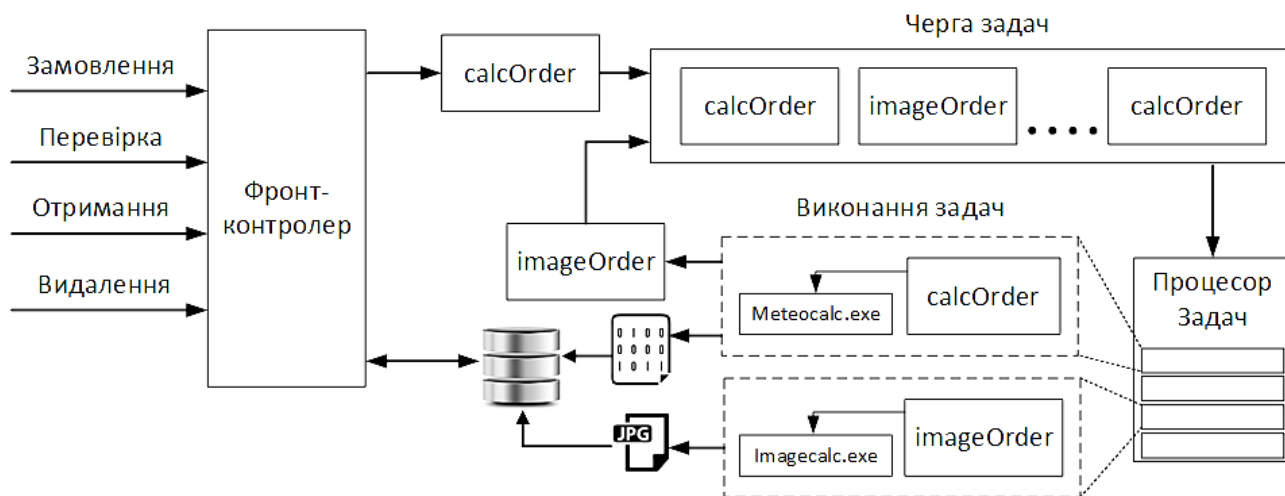


Рис. 2. Схема сервісу надання метеорологічних прогнозів

«Черга задач» – блокуюча черга реалізована на базі `LinkedBlockingQueue` пакету `java.util.concurrent`, що реалізує алгоритм «two lock queue»: одне блокування на додавання, інше на діставання елемента. Елементи черги – це задачі, які відрізняються між собою типом задачі (розрахунок метеоданих чи генерація зображення). Для кожної задачі зберігається інформація про статус («NOT\_CREATED» – задача на прогноз очікує доступу до черги (в разі її заповнення), «IN\_QUEUE» – задача перебуває в черзі на виконання, «IN\_PROCESSING» – задача на прогноз виконується в даний момент, «READY» – готовий результат).

«Процесор задач» – реалізований на базі інтерфейсу `ExecutorService` з пакету `java.util.concurrent` бере на себе роботу по запуску і зупинці потоків. Дана реалізація в найбільшій мірі підходить для паралельного асинхронного обрахунку прогнозів. Розмірність пулу автоматично вважається як 100 % завантаженості GPU. Відповідно кожна із задач має орієнтовний відсоток навантаження на GPU, що вказується у конфігураційному файлі.

Після виконання задачі обрахунку метеорологічного прогнозу, автоматично створюється задача на генерацію зображення та кладеться у чергу задач. Після виконання задачі на генерацію зображення на диску зберігається тимчасовий файл зображення як результат прогнозування.

Сервіс управління метеорологічними прогнозами виконує функцію зв'язкового між порталом та сервісом із GPU. Окрім того він зберігає у себе на дисковій системі результати усіх замовлень, що дає можливість отримати їх вдруге вже без створення виклику на сервіс обрахунку метеорологічних прогнозів.

В рамках даної роботи проведено порівняння швидкодії тієї частини програми, що виконувалась на графічному прискорювачі, із відповідною частиною, виконаною на центральному процесорі.

Випробування проводились із використанням процесора Intel Core i5-2500 (у 64-бітному режимі) та графічного прискорювача NVIDIA GeForce GTX 560, що має наступні характеристики:

- 336 stream-процесори (CUDA-ядра), базова частота 810 МГц;
- обсяг глобальної пам'яті 1024 МБ;
- базова частота пам'яті 2002 МГц.

Обчислення в експериментальній задачі для підрахунку часу виконання проводились для сіток локальних карт у вузлах просторової сітки розмірами 101×101×41, 111×111×45, 121×121×49, 131×131×53, 141×141×57, 151×151×61, 161×161×65, 171×171×69 та 191×191×77 частин. На рис. 3 показано графік залежності часу виконання обчислень послідовної та паралельної програм від розміру задачі (кількості вузлів розбиття розрахункової сітки). Відносне прискорення (відношення часу виконання послідовної до часу виконання паралельної програм, рис. 4) зростає від 11.6 для найменшої до 32.5 для найбільшої із розглянутих сіток. Таким чином, перехід на нову обчислювальну платформу з використанням GPU дозволив збільшити продуктивність виконання обчислень.

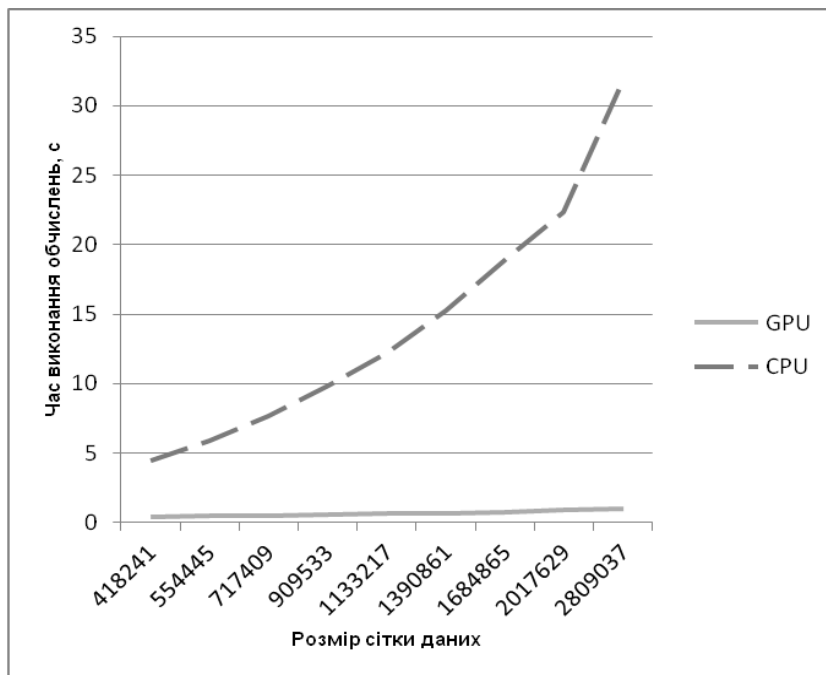


Рис. 3. Графік залежності часу виконання від розміру задачі

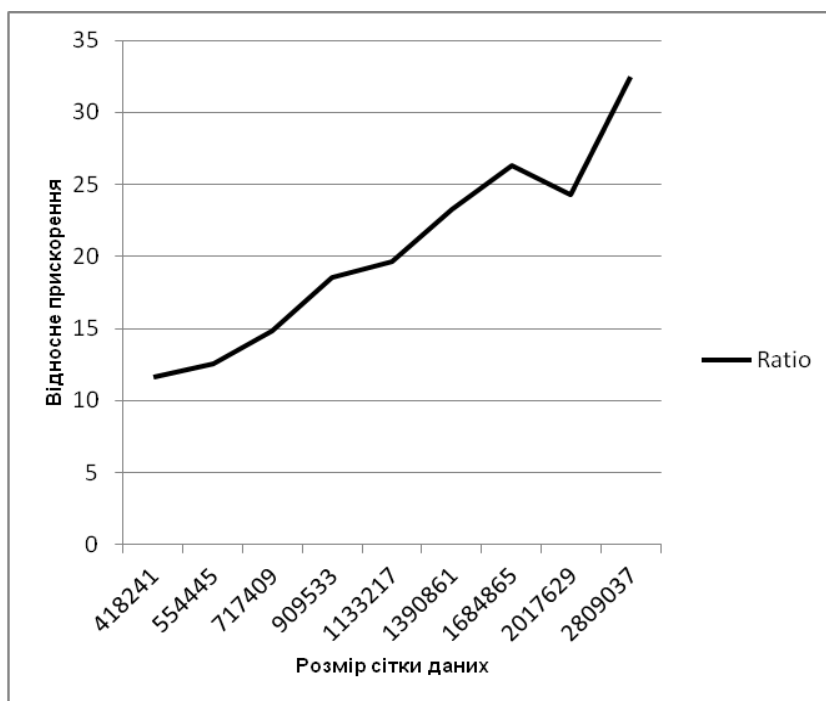


Рис. 4. Графік залежності відносного прискорення від розміру задачі

### 3. Візуалізація результатів обчислень

Існує чимало готових рішень для візуалізації масивів даних. Проте, провівши ретельний аналіз, можна стверджувати, що наразі відсутні он-лайн сервіси, які мають необхідну функціональність. Більшість представлених на даний момент програмних продуктів є платними рішеннями з закритим вихідним кодом.

Важливо зазначити, що масиви даних, отримані в результаті метеорологічного прогнозування, мають формат, що не відповідає існуючим програмним продуктам, а саме ці масиви містять лише значення досліджуваних величин в текстовому або бінарному форматі і не містять жодних координат. З урахуванням вище описаного, було створено новий програмний засіб, що дозволяє побудувати та візуалізувати такі об'єкти метеопрогнозування як ізолінії. Він виконує генерацію статичного зображення карти з нанесеними на неї позначками, за допомогою відеографічного прискорювача та зберігає його у файл.

Для побудови ізоліній необхідно описати масив даних поверхні, а також вказати рівні ізоліній які необхідно побудувати. Ця інформація передається програмному засобу, що далі виконує алгоритм обчислення відрізків, які складають криву ізолінії. Потім необхідно візуалізувати ці відрізки на будь-якому доступному графічному пристрої, використовуючи певний графічний програмний інтерфейс.

Розроблене програмне забезпечення використовує OpenGL [14], оскільки цей програмний інтерфейс є незалежним від платформи та мови програмування. Для побудови ізоліній використовується алгоритм CONREC [15], який є відносно простим у реалізації та дуже надійним.

Більшість алгоритмів побудови ізоліній розглядають прямокутну сітку даних по чотири точки за один раз, а саме прямокутник з вершинами  $d(i,j)$ ,  $d(i+1,j)$ ,  $d(i,j+1)$ , та  $d(i+1,j+1)$ . Особливістю алгоритму CONREC є те, що центру кожного прямокутника присвоюється значення, що відповідає середньому арифметичному значенню чотирьох вершин, а кожен прямокутник, в свою чергу, ділиться на чотири трикутні області, отримані розрізанням уздовж діагоналей. Кожна з цих трикутних областей може бути розділена горизонтальною площиною ізолінії. Перетин кожного трикутника з площиною ізолінії є прямим відрізком – частиною кривої ізолінії на поточному рівні ізолінії. Також реалізовано оптимізацію шляхом початкової перевірки того, чи проходить певна ізолінія через поточний прямокутник, а вже потім перевіряється проходження ізолінії через кожен з чотирьох трикутників.

Для створення файлів зображень візуалізації метеорологічного прогнозу використовується бібліотека DevIL, що підтримує велику кількість форматів зображень. Проте доцільно використовувати BMP, PNG або JPG. Налаштування засобу візуалізації здійснюється за допомогою файлу конфігурації у якому, зокрема, вказуються такі параметри як кількість рівнів ізоліній, найменше та найбільше значення рівня ізоліній і т. ін.

В реалізованому засобі візуалізації використовуються шейдери двох типів – вершинні та фрагментні. Всього розроблено шість шейдерних програм – по одному вершинному та фрагментному шейдеру для візуалізації, ізоліній та текстури. В якості текстури використовується карта, наприклад, карта України чи світу.

На рис. 5 показано результат роботи засобу візуалізації.

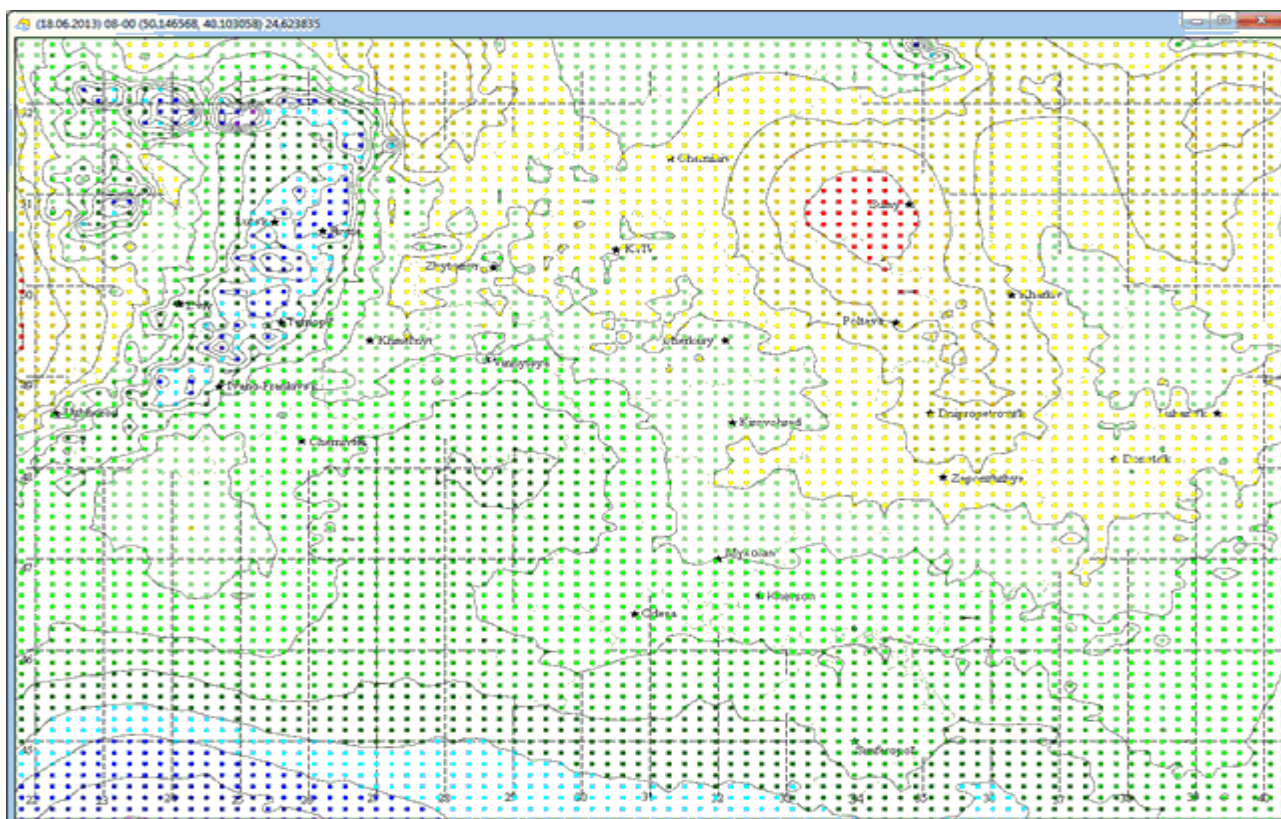


Рис. 5. Мапа 13 ізоліній в інтервалі від 15°C до 30°C

Обраний підхід має як свої переваги, так і недоліки. Серед недоліків можна вказати його статичність та необхідність виконання запиту на генерацію і пересилку нового зображення при бажанні отримати прогноз для іншої області. Але серед переваг можна зазначити можливість формування зображення будь-якої складності та використання будь-яких об'ємів даних, нанесених на мапу. Аналогічна процедура для динамічного відображення у попередньому підході була б проблематичною через обмеження ширини каналу, об'єму доступної пам'яті, та значне навантаження на процесор клієнтської машини. Тому при виборі підходу відображення метеорологічних даних кінцевому користувачу потрібно виходити з поточних цілей донесення бажаної інформації та можливостей технічного забезпечення серверної та клієнтської сторін.

Окрім сервісної реалізації засобу візуалізації також розроблено десктопний додаток для відображення даних метеорологічних прогнозів, який дозволяє інтерактивно виконувати візуалізацію даних, отримати детальну інформацію по конкретній точці на мапі, а також переглянути динаміку зміни прогнозу з плином часу (рис. 6).

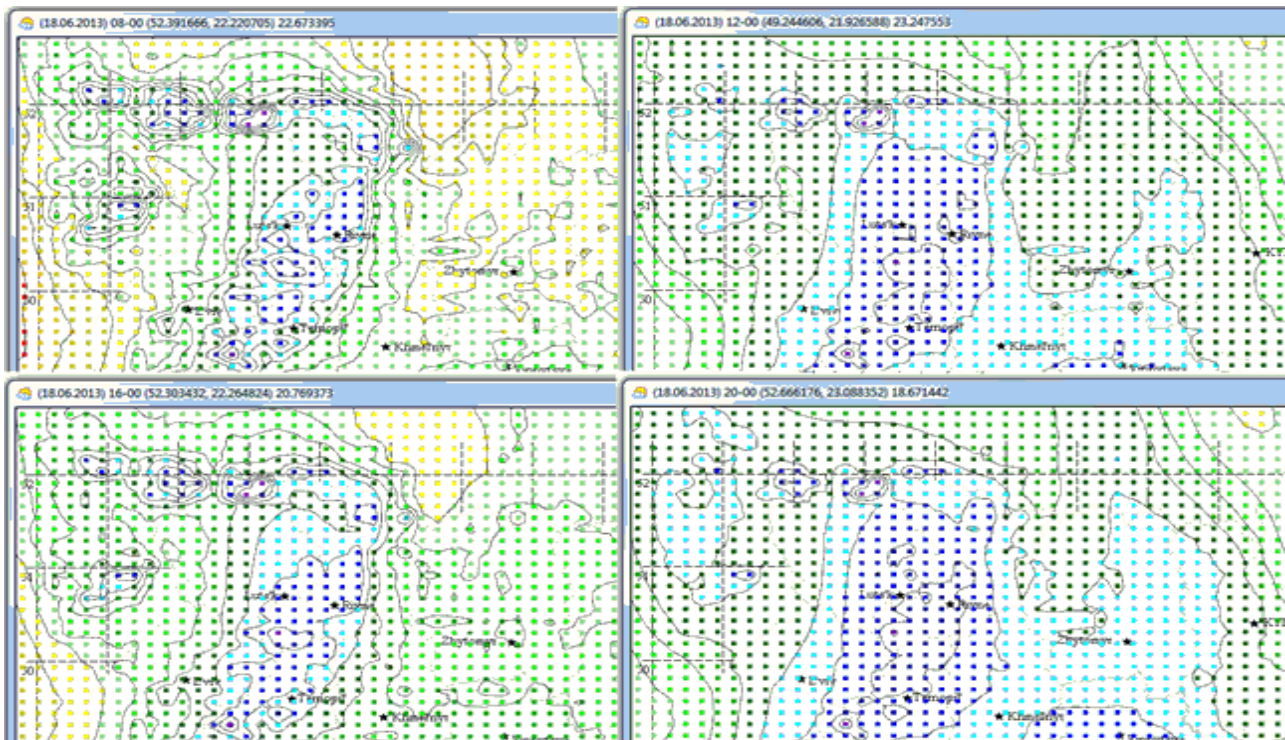


Рис. 6. Зміна метеорологічних показників у часі

## Висновки

У роботі розглянуто подальший розвиток реалізованого раніше сервісно-орієнтованого портального рішення для надання послуг метеорологічного прогнозування [1]. Було додано модуль, для розрахунку тривимірного алгоритму вирішення задачі моделювання циркуляції атмосфери з використанням нової обчислювальної платформи на базі графічних прискорювачів. Перехід на нову обчислювальну платформу дозволив значно збільшити продуктивність виконання обчислень. Окрім того було обрано новий ефективний метод для візуалізації результатів метеорологічного прогнозування – статичне зображення карти з нанесеними на неї позначками, також сформоване за допомогою відеографічного прискорювача. Створено ефективний засіб візуалізації, що застосовує кросплатформні бібліотеки, а також має можливість запускатися ззовні іншим додатком чи скриптом, що дозволяє інтегрувати засіб візуалізації в сервіс.

Представлену систему реалізовано за допомогою сучасних провідних технологій, таких як інтернет-портали та мікро-сервіси. Завдяки сервісно-орієнтованій архітектурі вона інтегрує різноманітні незалежні сервіси.

В подальшому планується розвивати цю систему покращуючи її розширювальну здатність, додаючи нові сервіси, нові більш складні та потужні обчислювальні алгоритми та моделі та функціонально збагачувати існуючі сервіси.

1. Дорошенко А.Ю., Іваненко П.А., Овдій О.М., Павлючин Т.О., Вітряк Є.А. До створення Інтернет-порталу надання послуг метеорологічного прогнозування на мультипроцесорній платформі // Проблеми програмування. – 2015. – № 3. – С. 24–32.
2. Прусов В.А., Дорошенко А.Ю. Моделювання природних і техногенних процесів в атмосфері. – К.: Наукова думка, 2006. – 542 с.
3. Толстих М. А., Фролов А. В. Некоторые современные проблемы численного прогноза погоды // М.: Известия РАН. Физика атмосферы и океана. – 2005. – Т. 41, № 3. – С. 315–327.
4. Прусов В.А., Сніжко С. І. Математичне моделювання атмосферних процесів. – Київ: Ніка-Центр, – 2005. – 496 с.
5. Ківганов А.Ф., Хоменко Г.В., Хохлов В.М., Бондаренко В.М. Гідродинамічні методи прогнозу погоди і сіткові методи їх реалізації. – Одеса: Одеський державний екологічний університет. – 2002. – 179 с.
6. Дорошенко А.Ю., Бекетов О.Г., Прусов В.А., Турчак Ю.М., Яценко О.А. Формалізоване проектування та генерація паралельної програми чисельного прогнозування погоди // Проблеми програмування. – 2014. – № 2–3. – С. 72–81.
7. Дорошенко А.Ю., Яценко О.А. Бекетов О.Г. Застосування графічних прискорювачів до задач метеорологічного прогнозування. // Теоретичні та прикладні аспекти побудови програмних систем. – 2014. – С. 101–105.
8. CUDA Zone [Електронний ресурс]. – Режим доступу: <https://developer.nvidia.com/cuda-zone>. – 25.02.2016 р.
9. Wetter und Klima – Deutscher Wetterdienst – Startseite (веб-сайт метеорологічного центру у м. Оффенбах) [Електронний ресурс]. – Режим доступу: <http://www.dwd.de>. – 25.02.2016 р.

10. *Fielding, Roy Thomas*. Architectural Styles and the Design of Network-based Software Architectures. Doctoral dissertation, – University of California, Irvine, 2000. [Електронний ресурс]. – Режим доступу: <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm> – 25.02.2016 р.
11. SOAP Specifications - W3C. [Електронний ресурс]. – Режим доступу: <https://www.w3.org/TR/soap/>. – 25.02.2016 р.
12. Spring Framework. [Електронний ресурс]. – Режим доступу: <http://projects.spring.io/spring-framework/>. – 25.02.2016 р.
13. Liferay – Enterprise open source portal and collaboration software. [Електронний ресурс]. – Режим доступу: <http://www.liferay.com/>. – 25.02.2016 р.
14. OpenGL – The Industry Standard for High Performance Graphics. [Електронний ресурс]. – Режим доступу: <https://www.opengl.org/>. – 25.02.2016 р.
15. Bourke P. CONREC: A contouring subroutine // Byte: The Small Systems Journal. – 1987. – 12(6). – P. 43–150.

## References

1. Doroshenko, A.Yu., Ivanenko, P.A., Ovdii, O.M., Pavliuchyn, T.O. & Vitriak, I.A. (2015) Creation of an Internet portal providing meteorological forecasting services on multiprocessor platform. Problems in programming. (3). P. 24–32. (in Ukrainian)
2. Prusov, V.A. & Doroshenko, A.Yu. (2006) Simulation of natural and anthropogenic processes in the atmosphere. Kyiv: Naukova Dumka. (in Ukrainian)
3. Tolstykh, M.A. & Frolov, A.V. (2005) Some current problems in numerical weather prediction. Izvestiya. Atmospheric and Oceanic Physics. (3). P. 285–295. (in Russian)
4. Prusov, V.A. & Snizhko, S.I. (2005) Mathematical modeling of atmospheric processes. Kyiv: NikaTsentr. (in Ukrainian)
5. Kivganov, A.F., Khomenko, G.V., Khokhlov, V.M. & Bondarenko, V.M. (2002) Hydrodynamic methods of weather forecasting and grid methods for their implementation. Odessa: Odessa State Environmental University. (in Ukrainian)
6. Doroshenko, A.Yu., Beketov, O.G., Prusov, V.A., Tyrchak, Yu.M. & Yatsenko, O.A. (2014) Formalized design and generation of parallel programs for numerical weather forecast. Problems in programming. (2-3). P. 72–81. (in Ukrainian)
7. Doroshenko, A.Yu., Yatsenko, O.A. & Beketov, O.G. (2014) The use of graphics processing units for meteorological forecasting tasks. Theoretical and applied aspects of building software systems. (in Ukrainian)
8. CUDA (2016) CUDA Zone. [Online] Available from: <https://developer.nvidia.com/cuda-zone>. [Accessed: 25th February 2016]
9. DWD (2015) Wetter und Klima – Deutscher Wetterdienst. [Online] Available from: <http://www.dwd.de> [Accessed: 25th February 2016]
10. Fielding, R. T. (2000) Architectural Styles and the Design of Network-based Software Architectures. Doctoral dissertation, – University of California, Irvine. [Online] Available from: <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>. [Accessed: 25th February 2016]
11. SOAP (2016) SOAP Specifications - W3C. [Online] Available from: <https://www.w3.org/TR/soap/>. [Accessed: 25th February 2016]
12. SPRING (2016) Spring Framework. [Online] Available from: <http://projects.spring.io/spring-framework/>. [Accessed: 25th February 2016]
13. LIFERAY (2016) Liferay - Enterprise open source portal and collaboration software. [Online] Available from: <http://www.liferay.com/>. [Accessed: 25th February 2016]
14. OpenGL - The Industry Standard for High Performance Graphics. [Online] Available from: <https://www.opengl.org/>. [Accessed: 25th February 2016]
15. Bourke P. (1987) CONREC: A contouring subroutine. Byte: The Small Systems Journal, 12(6): P. 43–150.

## Про авторів:

*Бекетов Олексій Геннадійович*

молодший науковий співробітник.

Кількість наукових публікацій в українських виданнях – 9.

<http://orcid.org/0000-0003-4715-5053>,

*Вітряк Євгеній Андрійович,*

аспірант Інституту програмних систем НАН України.

Кількість наукових публікацій в українських виданнях – 2.

<http://orcid.org/0000-0002-7198-6634>,

*Мироненко Ігор Олексійович,*

аспірант Інституту програмних систем НАН України.

Кількість наукових публікацій в українських виданнях – 2.

<http://orcid.org/0000-0003-3768-4226>,

*Овдій Ольга Михайлівна,*

молодший науковий співробітник.

Кількість наукових публікацій в українських виданнях – 17.

Кількість наукових публікацій в іноземних виданнях – 3.

<http://orcid.org/0000-0002-8891-7002>.

## Місце роботи авторів:

Інститут програмних систем НАН України.

03187, Київ-187, проспект Академіка Глушкова, 40, корпус 5.

Тел.: +380 (44) 526 3559, факс +380 (44) 526 6263.

E-mail: [beketov.oleksii@gmail.com](mailto:beketov.oleksii@gmail.com),

[steplerbush@gmail.com](mailto:steplerbush@gmail.com),

[mioe@ukr.net](mailto:mioe@ukr.net),

[olga.ovdiy@gmail.com](mailto:olga.ovdiy@gmail.com).