Problems of identification in information systems

УДК 004.9 **doi: 10.20998/2522-9052.2018.2.01**

О. Г. Толстолузька, Б. В. Паршенцев

Харківський національний університет імені В. Н. Каразіна, Харків, Україна

РІШЕННЯ ЗАДАЧІ КЛАСИФІКАЦІЇ В E-LEARNING НА ОСНОВІ МЕТОДУ ПАРАЛЕЛЬНОЇ ПОБУДОВИ ДЕРЕВ РІШЕНЬ

Актуальність. Останнім часом в розвинутих країнах питанням машинного навчання приділяється все більше уваги. З одного боку це пов'язано зі стрімким ростом вимог до майбутніх фахівців, а з іншого - з дуже швидким розвитком інформаційних технологій та Інтернет комунікацій. Однією з головних задач e-learning є задача класифікації. Математичний апарат дерев рішень гарно пристосований для рішення задачі класифікації. Однак, з ростом кількості вхідних даних стає актуальним питання зменшення часу побудови дерев рішень. Використання паралельних обчислювальних систем та паралельних технологій програмування дозволяє отримати позитивні результати, але вимагає розробки нових методів побудови дерев рішень. **Результати.** В статті розкриваються основні етапи методу паралельної побудови дерев для вирішення задачі класифікації в e-learning. На відміну від існуючих, метод дозволяє враховувати особливості архітектури і організації паралельних процесів в обчислювальних системах із загальною і розподіленою пам'яттю. В методі врахована можливість оцінки показників ефективності побудови дерев рішень та паралельних алгоритмів. Отримання показників ефективності на кожній ітерації методу допомагає обрати раціональну кількість паралельних процесорів в обчислювальній системі. Це дозволяє домогтися подальшого скорочення часу побудови дерев рішень. Проведене моделювання з використанням технології паралельного програмування MPI, мови програмування Python для архітектури обчислювальної системи DM-MIMD підтверджує достовірність отриманих результатів. Наводиться приклад організації вхідних даних. Представлено Python програму для побудови дерева рішень. **Висновок.** Розроблена візуалізація отриманих оцінок показників ефективності дозволяє користувачу обрати необхідну конфігурацію обчислювальної системи.

Ключові слова: паралельний алгоритм; дерево рішень; оцінка ефективності паралельного алгоритму; e-learning.

Вступ

Постановка проблеми. В даний час відбуваються величезні і невідворотні зміни в щоденному людському житті: вимоги до освіти зростають з кожним роком, технології щільно входять в усі сфери людської діяльності. Дані зміни зумовили необхідність створення комфортних умов для повноцінної, постійної, стрімкої, високоякісної і сучасної підготовки кожного працівника, але на жаль звичайні світові стандарти в системі навчання в повній мірі не відповідають даним потребам. У зв'язку з цим були зроблені спроби реформування старої системи та створення нової. Але з розвитком інтернету прийшло розуміння, що e-learning це не просто віддалене отримання знань. Даний вид освіти передбачає використання освітнього матеріалу і безперервне спілкування учня і викладача через глобальну мережу [1, 2]. E-learning - це можливість гетерогенної освіти (яка об'єднує кілька способів взаємодії), очної та віртуальної, яка стала основною формою взаємодії викладача та учня. В даному випадку учень інтегрований в систему де основну роль освітньої системи на себе бере електронна компонента системи але також присутня пряма взаємодія людини з людиною. E-learning – «сучасний аспект у навчанні, застосовуваний для того, щоб забезпечити добре продумане діалогове середовище навчання будь якому учню, коли і де завгодно, використовуючи ресурси різних цифрових технологій поряд з іншими формами навчальних матеріалів, які підходять для відкритого середовища навчання. E-learning здійснює перехід від системи управління даними до системи управління знаннями». Основне завдання e-learning – розробка та підтримка індивідуальної освітньої програми для кожного студента та поліпшення поточного рівня знань і отримання нових знань. Також одна з основних задач e-learning – це постійний моніторинг факторів, які впливають на успішність навчання. Міжнародна аналітична компанія IDC провела дослідження в сфері сучасного електронного навчання. Результатами даного дослідження стали такі тенденції: необхідність комплексних рішень та розробка Єдиних стандартів на систему дистанційного навчання і електронний контент; через постійний розвиток глобальної павутини потрібно також постійно розвивати інтерфейси за допомогою яких людина спілкується з машиною. Поточний стан глобальної мережі описується як WEB 2.0. Основні зміни в WEB 2.0 – це те, що контент створюється самими користувачами [3]. Дуже багато контенту перейшло в мобільне середовище. У зв'язку з цим досить великий відсоток освітнього контенту перейшло в мобільний вимір з заголовком m-learning. Розвиток Rapid e-learning (швидка розробка рішень e-learning); зростання популярності і кількості систем e-learning призвело до зменшення вартості на дані системи [4]. На відміну від початкової версії e-learning, яка передбачала використання в якості основного інструменту дистанційних курсів,

які надавалися учням з метою проведення навчання, e-learning 2.0 передбачає використання засобів Web 2.0: блоги, wiki, підкасти, соціальні мережі. На даному етапі одна з основних задач, яка вирішується в e-learning - це задача класифікації.

З даним класом задач непогано справляються дерева рішень. Для прискорення побудови дерева рішень можна використовувати технології паралельного програмування.

У статті представлені основні етапи метода паралельної побудови дерева рішень для вирішення задачі класифікації.

Метою статті є опис основних етапів метода паралельної побудови дерева рішень для задач e-learning в інтересах зменшення часу вирішення задачі класифікації та підвищення показників ефективності.

Дослідження і результати

Введемо деякі визначення і поняття.

Дерева рішень – це спосіб представлення правил в ієрархічній, послідовній структурі, де кожному об'єкту відповідає єдиний вузол, що дає рішення [5].

Алгоритм CART – алгоритм, призначений для побудови бінарного дерева рішень. Бінарні дерева також називають двійковими, це означає, що кожен вузол дерева при розбитті має тільки двох нащадків. Для алгоритму CART «поведінка» об'єктів виділеної групи означає частку модального значення вихідної ознаки. Виділені групи - ті, для яких ця частка досить висока. На кожному кроці побудови дерева правило, яке формується в вузлі, поділяє задану множену прикладів на дві частини - частина, в якій виконується правило (нащадок - right) і частина, в якій правило не виконується (нащадок - left) [6].

Індекс Джині – статистичний показник, за допомогою якого можна описувати характер зміни однієї величини відносно зміни іншої. Основним застосуванням індексу Джині є оцінка нерівномірності розподілу досліджуваної ознаки [7].

В якості початкових даних методу використовуються такі:

- база даних (обсяг даних ~ 500ГБ) (рис. 1);
- класи паралельних архітектур (VLIW - GPU , AMD/ATI Radeon(HD5770), RISC - ARM ThunderX, CISC - IBM System z10);
- характеристики архітектури: кількість NM процесорів
- час вирішення $T_{реш}$ задачі;
- технологія паралельного програмування (CUDA - архітектура паралельних обчислень від NVIDIA, OpenMP API-інтерфейс, який є галузевим стандартом для створення паралельних програм для комп'ютерів зі спільним використанням пам'яті; MPI - інтерфейс передачі повідомлень для систем з розподіленою пам’яттю) [4].

Вихідні дані повинні бути розподілені по класам і результати повинні бути візуалізовані. Також повинні бути отримані оцінки показників ефективності: прискорення, час побудови дерева рішень, складність програми.

Основні етапи метода паралельної побудови дерева рішень представлені на рис. 2.

```
StandaloneInstitution =; {id, address_line1, address_line2,
city, state_code, district_code,
website, area, constructed_area,
year_of_establishment, year_of_recognition,
nodalofficer_id, location,
awards_degree_through_university, university_id,
girl_exclusive,
staff_quarter_available, staff_quarter_id, stu-
dent_hostel_available,
no_of_student_hostel,management_id, name, survey_year,
financial_income_id,financial_expenditure_id, infrastruc-
ture_id, remarks}

StandaloneInstitutionAccredida-
tion={standalone_institution_id, survey_year,
accreditation_id}

StandaloneInstitutionDepartment = {stand-
alone_institution_id, department_id,
survey_year}

StandaloneInstitutionFaculty={standalone_institution_id
,faculty_id, survey_year }

StandaloneInstitutionNonTeachingStaff = {stand-
alone_institution_id, survey_year,
non_teaching_staff_count_id}

StandaloneInstitutionStudentHotel = { stand-
alone_institution_id, student_hostel_id,
survey_year}

StandaloneInstitutionTeachingStaff = {stand-
alone_institution_id, survey_year,
teaching_staff_id}

StandaloneInstitutionTeachingStaffSanctioned-
Strength={standalone_institution_id,

survey_year, teaching_staff_sanctioned_strength_id }
```

Рис. 1. Приклад організації вхідних даних (опис деяких таблиць і ключові поля)

Розглянемо призначення і формалізований опис основних етапів методу.

Eman 1 (блок 2 рис. 2). На цьому етапі здійснюється вибір архітектури паралельної обчислювальної системи.

Eman 2 (блоки 3 – 6 рис. 2) забезпечує вибір різновиду декомпозиції в залежності від обраної архітектури: функціональна або декомпозиція по даним.

Eman 3 (блок 7 рис. 2) забезпечує вибір технології програмування для паралельної побудови дерева рішень.

Eman 4 (блок 8 рис. 2) – розподіл отриманих даних по процесах W (процеси можуть бути виконані на окремих процесорах, або декілька процесів на одному процесорі).

Eman 5 (блок 9 рис. 2) здійснює побудову дерева рішень на кожному вузлі (рис. 3, 4).

Eman 6 (блоки 10, 11, 12 рис. 2). Пересилання всіх дерев на один процес. Оцінювання показників ефективності: час побудови дерев рішень і якщо $T_п \geq T_{реш}$ то переходимо на наступний етап.

Eman 7 (блоки 13, 14 рис. 2). Порівняння кількості процесів яке було задано і яке було використано, і, якщо $NM \leq NM_з$ то збільшення кількість процесів та перехід на четвертий етап.

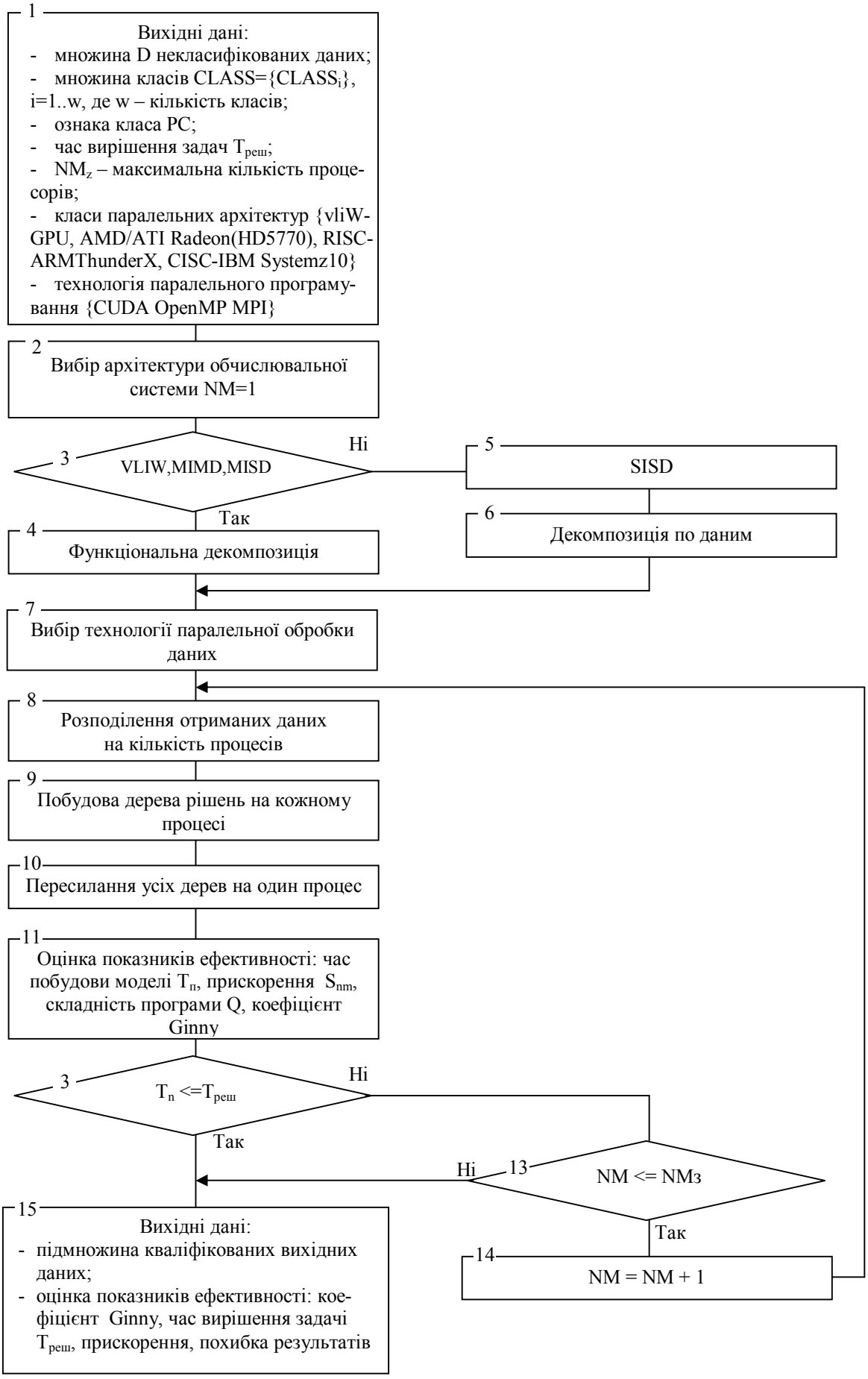

Рис. 2. Основні етапи методу паралельної побудови дерев рішень

```
def split(node, max_depth, min_size, depth):
    left, right = node['groups']
    del(node['groups'])
    if not left or not right:
        node['left'] = node['right'] = to_terminal(left + right)
        return
    if depth >= max_depth:
        node['left'], node['right'] = to_terminal(left),
to_terminal(right)
        return
    if len(left) <= min_size:
        node['left'] = to_terminal(left)
    else:
        node['left'] = get_split(left)
        split(node['left'], max_depth, min_size, depth+1)
    if len(right) <= min_size:
        node['right'] = to_terminal(right)
    else:
        node['right'] = get_split(right)
        split(node['right'], max_depth, min_size, depth+1)
def build_tree(train, max_depth, min_size):
    root = get_split(train)
        split(root, max_depth, min_size, 1)
        return root
```

Рис. 3. Python програма для побудови дерева

Рис. 4. Блок схема паралельної побудови дерева

На рис. 4 представлена блок схема розподілу рівних частин даних в пам'яті для побудови дерев рішень за допомогою алгоритму CART.

Результати використання паралельної обробки інформації при побудові дерева рішень можна прокоментувати таким чином:

- найкращий приріст в прискоренні досягається шляхом використання 8 процесів (рис. 5);
- рішення задачі класифікації з використанням паралельного підходу прискорює отримання результатів приблизно в 20 раз, проте ускладнює розробку з точки зору складності програмної реалізації;
- при паралельній побудові дерев рішень потрібно брати суму всіх коефіцієнтів Ginny;
- рівень достовірності результатів класифікації при паралельному виконанні складає 83.76%, а при використанні одного потоку рівень достовірності результатів класифікації дорівнює 76.5%.

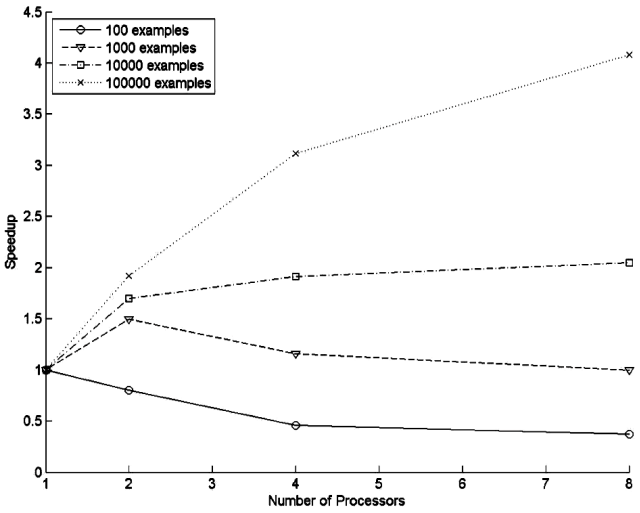

Рис. 5. Залежність прискорення від кількості процесорів

Висновки

Розглянуто основні етапи методу паралельної побудови дерева рішень для задач e-learning в інтересах створення інформаційної технології для вирішення задач класифікації.

Використання паралельності в побудові дерева рішень для задачі класифікації дає прискорення в порівнянні з однопоточною реалізацією. При моделюванні застосовувалася технологія MPI, мова програмування Python, DM-MIMD архітектура.

Розроблена візуалізація отриманих оцінок показників ефективності дозволяє користувачу обрати необхідну конфігурацію обчислювальної системи.

СПИСОК ЛІТЕРАТУРИ

1. Сергеев А. Г. Введение в электронное обучение : монография / А. Г. Сергеев, И. Е. Жигалов, В. В. Баландина. – Владимир, ВлГУ, 2012. – 182 с.
2. Шматков С. І. Модель інформаційної структури гіперконвергентної системи підтримки електронних обчислювальних ресурсів університетської e-learning / С. І. Шматков, Н. Г. Кучук, В. В. Донець // Системи управління, навігації та зв'язку : науковий журнал. – Полтава : ПНТУ, 2018. – Вип. 2(48). – С. 97-100.
3. Kuchuk G. Approaches to selection of combinatorial algorithm for optimization in network traffic control of safety-critical systems / G. Kuchuk, V. Kharchenko, A. Kovalenko, E.Ruchkov // East-West Design & Test Symposium (EWDTS). – 2016. –P. 1-6, available at: https://doi.org/10.1109/EWDTS.2016.7807655.
4. Воеводин В. В. Параллельные вычисления / В. В. Воеводин, Вл. В. Воеводин. – СПб. : БХВ-Петербург, 2002. – 608 с.

5. Breiman L. Classification and Regression Trees / L. Breiman, J.H. Friedman, R.A. Olshen, C.T. Stone. – Wadsworth, Belmont, California, 1984.
6. Wei-YinLoh. BOAT – optimistic decision tree construction / Johannes Gehrke, Venkatesh Ganti, Raghu Ramakrishnan // ACM SIGMOD International Conference on Management of Data, June 1999, p. 169-180.
7. Поляков Г.А. Синтез и анализ параллельных процессов в адаптивних времяпараметризованных вычислительных системах / Г.А. Поляков, С.И. Шматков, Е.Г. Толстолужская, Д.А. Толстолужский. – Х. : ХНУ, 2012. – С. 434-575.

REFERENCES

1. Sergeev, A., Zhigalov, I., and Balandina, V. (2012), *Introduction to e-learning*, VlSU, Vladimir, 182 p.
2. Shmatkov, S.I., Kuchuk, N.G. and Donets, V.V. (2018), "The model of information structure of the hyperconvergent system of support of electronic computing resources of university e-learning", *Control, navigation and communication systems*, PNTU, Poltava, No. 2 (48), pp. 97-100.
3. Kuchuk, G., Kharchenko, V., Kovalenko, A. and Ruchkov, E. (2016), "Approaches to selection of combinatorial algorithm for optimization in network traffic control of safety-critical systems", *East-West Design & Test Symposium (EWDTS)*, pp. 1-6, available at: https://doi.org/10.1109/EWDTS.2016.7807655.
4. Voevodin, V.V. (2002), Parallel computing, BHV-Petersburg, St. Petersburg, 608 p.
5. Breiman, L., Friedman, J.H., Olshen, R.A. and Stone, C.T. (1984), *Classification and Regression Trees*, Wadsworth, Belmont, California.
6. Gehrke, Johannes, Ganti, Venkatesh, Ramakrishnan, Raghu and Loh, Wei-Yin (1999), BOAT – optimistic decision tree construction, ACM SIGMOD International Conference on Management of Data, June 1999, pp. 169-180.
7. Polyakov, G.A., Shmatkov, S.I., Tolstoluzhskaya, E.G. and Tolstoluzhsky D.A. (2012), Synthesis and Analysis of Parallel Processes in Adaptive Time-Parameterized Computer Systems, KhNU, Kharkiv, pp. 434-575.

Received (Надійшла) 22.03.2018
Accepted for publication (Прийнята до друку) 23.05.2018

Решение задачи классификации в e-learning на основе метода параллельного построения деревьев решений

Е. Г. Толстолужская, Б. В. Паршенцев

Актуальность. В последнее время в развитых странах вопросам машинного обучения отводится все больше внимания. С одной стороны это связано со стремительным ростом требований к будущим специалистам, а с другой – с очень быстрым развитием информационных технологий и Интернет коммуникаций. Одной из главных задач e-learning есть задача классификации. Математический аппарат деревьев решений часто применяется для решения задачи классификации. Однако, с ростом количества входных данных становится актуальным вопрос уменьшения времени построения деревьев решений. Использование параллельных вычислительных систем и параллельных технологий программирования позволяет получить положительные результаты, но требует разработки новых методов построения деревьев решений. **Результаты.** В статье раскрываются основные этапы метода параллельного построения деревьев для решения задачи классификации в e-learning. В отличие от существующих, метод позволяет учитывать особенности архитектуры и организации параллельных процессов в вычислительных системах с общей и распределенной памятью. В методе учтена возможность оценки показателей эффективности построения деревьев решений и параллельных алгоритмов. Получение показателей эффективности на каждой итерации метода помогает избрать рациональное количество параллельных процессоров в вычислительной системе. Это позволяет добиться дальнейшего сокращения времени построения деревьев решений. Проведенное моделирование с использованием технологии параллельного программирования MPI, языка программирования Python для архитектуры вычислительной системы DM-MIMD подтверждает достоверность полученных результатов. Приводится пример организации входных данных. Представлена Python программа для построения дерева решений. **Вывод.** Разработанная визуализация полученных оценок показателей эффективности помогает пользователю избрать необходимую конфигурацию вычислительной системы.

Ключевые слова: параллельный алгоритм; дерево решений; оценка эффективности параллельного алгоритма; e-learning.

The solution of the classification problem in e-learning based on the method parallel construction of decision trees

O. Tolstoluzka, B. Parshencev

Topicality. Recently, more and more attention has been paid to the issues of machine learning in developed countries. On the one hand, this is due to the rapid growth of requirements for future specialists, and on the other - with the very rapid development of information technology and Internet communications. One of the main tasks of e-learning is the task of classification. The mathematical modeling system of decision trees is well adapted for the solution of the classification problem. However, as the number of input data increases, the issue of reducing the time of tree construction is becoming relevant. Using parallel computing systems and parallel programming technologies can produce positive results, but requires the development of new methods for constructing tree solutions. **Results.** The article reveals the main stages of the parallel tree construction method for solving the classification problem in e-learning. Unlike existing ones, the method allows to take into account the features of architecture and the organization of parallel processes in computing systems with shared and distributed memory. The method takes into account the possibility of evaluating performance indicators for constructing decision trees and parallel algorithms. Obtaining performance indicators for each iteration of the method helps to select the rational number of parallel processors in the computing system. This allows you to further reduce the time of building tree solutions. The simulation with the use of MPI parallel programming technology, the Python programming language for the architecture of the DM-MIMD system, confirms the reliability of the results. Here is an example of the organization of input data. Presented by Python is a program for building a decision tree. **Conclusion.** The developed visualization of the obtained estimates of performance indicators allows the user to select the necessary configuration of the computing system.

Keywords: parallel algorithm; decision tree; evaluation of the parallel algorithm efficiency; e-learning.