

## АЛГОРИТМИ ПАРАЛЕЛЬНОЇ ДЕКОМПОЗИЦІЇ ЕЛЕКТРОННИХ СХЕМ

© Базилевич Р., Влах В., Януш Д., 2011

**Пропонується алгоритмічна та програмна реалізація ієрархічної декомпозиції електронних схем високих розмірностей, що забезпечує можливість поділу схем на довільну кількість фрагментів з мінімізацією кількості зв'язків між ними.**

**Ключові слова:** ієрархічна кластеризація, розбиття схеми, паралельна декомпозиція.

**An efficient software for hierarchical decomposition of large-scale electronic circuits is proposed. Algorithm provides division of electronic circuits onto desirable number of partitions with minimization the number of common nets.**

**Keywords:** hierarchical clustering, circuits partitioning, parallel decomposition,

### Вступ

Задачі декомпозиції електронних схем належать до важковирішуваних комбінаторних класу NP [1–3]. Враховуючи високі розмірності сучасних електронних схем, а також стійкі тенденції до подальшого їх ускладнення, можна стверджувати, що дослідження задачі декомпозиції та розроблення нових, ефективніших підходів до їх розв'язання є актуальним. Алгоритм декомпозиції може вважатися ефективним, якщо він забезпечує можливість якісного поділу на необхідну (довільну) кількість фрагментів; дає змогу застосовувати різні критерії оптимізації; надає можливості формувати розв'язок з необхідним показником розподілу елементів, зокрема на рівні частини. Такі алгоритми широко застосовують, для розв'язування задач компонування, пакування, розміщення, тестування програмних засобів та інших.

### Формулювання задачі

Задача декомпозиції складних електронних схем полягає у розподілі її складових елементів між набором частин (рис. 1). Щоб отримати якісне розбиття, необхідно задовольнити ряд критеріїв та обмежень, зокрема:

- мінімізувати кількість зовнішніх зв'язків;
- забезпечити бажане число частин;
- забезпечити бажану кількість елементів у кожній частині;
- згрупувати певні елементи в заданих частинах або їх рознести;
- рівномірно розподілити елементи;
- інші.

Для отримання якісного розбиття схеми необхідно враховувати структуру її зв'язків. Важливим інструментом аналізу структури електронної схеми в алгоритмах ієрархічної декомпозиції виступає дерево її згортання, яке виділяє ієрархічно вкладені кластери [1–3]. Воно надає інформацію про її згустки, які формують набори елементів, сильніше пов'язані між собою, ніж із іншими частинами схеми. Такі вузли доцільно включати в окремі фрагменти розбиття. Це дасть змогу знизити кількість зовнішніх зв'язків між частинами.

Будуючи алгоритми розбиття, потрібно враховувати можливість поділу схеми на самостійні, незв'язані підсхеми в процесі її згортання. В задачах розбиття особливо складною є проблема оптимального розподілу елементів між фрагментами розв'язку. Однією з важливих проблем

декомпозиції, яка залишається малодослідженою, є забезпечення рівномірності розбиття, суть якої зводиться до пошуку оптимального компромісу між критеріями та обмеженнями, які в конкретних випадках можуть конкурувати. Необхідно отримати мінімальну кількість зв'язків між фрагментами, та, з іншого боку, забезпечити рівномірність поділу, тобто врахувати обмеження на кількості елементів кожної частини. Для мінімізації кількості зв'язків між фрагментами зв'язані групи елементів вносяться в спільні частини, але часто це погіршує рівномірність розбиття. І навпаки, при спробах збалансувати розмір частин неминуче збільшуються зв'язки між ними.

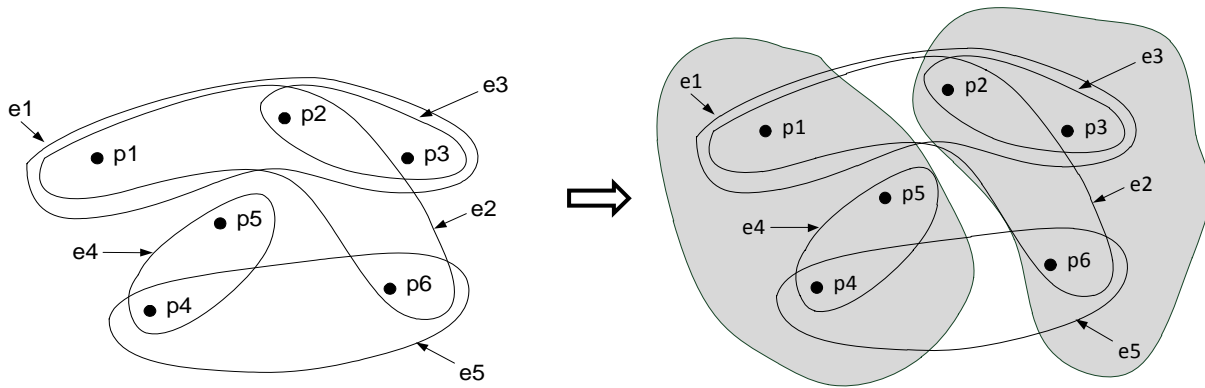


Рис. 1. Гіперграфове представлення схеми та розбиття

Враховуючи високі розмірності реальних задач, критичною є проблема мінімізації витрат часу та обчислюваних ресурсів для формування розв'язку. Частково вирішити цю проблему дає змогу підхід, описаний в [4].

Можливими є декілька підходів до формування оптимального розбиття схем:

1. Послідовне розбиття.
2. Паралельне розбиття.
3. Дихотомічне розбиття.
4. Інші.

Кожний з цих підходів має свої особливості. В алгоритмах послідовного виділення частин перші з них будуть якісними, формуватимуть сильно зв'язані групи елементів. Проте останні міститимуть елементи, які слабо або взагалі не зв'язані між собою. Алгоритм просто реалізується. Тому доцільно розробляти алгоритми, які не мають такого недоліку, проте вимагають складнішої алгоритмічної реалізації.

#### Алгоритми паралельної декомпозиції

Паралельні алгоритми розбиття мають ряд переваг над іншими, які також застосовують ієрархічну кластеризацію для початкового виділення частин. Розроблені алгоритми позбавлені таких недоліків:

- на відміну від послідовного розбиття, що здатне за одну ітерації згортки формувати один фрагмент розв'язку, дає змогу отримати декілька фрагментів. Це покращує показники продуктивності та ресурсовитратності процесів формування розв'язку;
- порівняно з дихотомічним методом розбиття не залежить від структури дерева згортки та його рівномірності;
- краще враховує особливості структури конкретної електронної схеми, оскільки продовжує формування фрагментів розбиття до повного завершення ітерації згортки.

Алгоритм складається з таких етапів:

1. Підготовка до розбиття: завантаження електронної схеми, вибір та налаштування алгоритму згортки, вибір кількості фрагментів.
2. Ієрархічне згортання схеми.
3. Аналіз розміру утвореного кластера. Якщо кластер відповідає необхідному розміру, то перехід до кроку 4, інакше перехід до кроку 5.

4. Блокування поточного кластера від подальшого згортання.
5. Якщо згортання схеми завершено, то перехід до кроку 6, інакше перехід до кроку 2.
6. Перетворення заблокованих кластерів на фрагменти результуючого розбиття. Якщо список заблокованих кластерів порожній, то перехід до кроку 9.
7. Якщо усі фрагменти розбиття сформовано, то перехід до кроку 13.
8. Утворення підсхеми із елементів, що не увійшли в сформовані фрагменти, перехід до кроку 2.
9. Впорядкування вершин утвореного лісу кластерів за зменшенням розміру.
10. Формування нового фрагмента розбиття на основі найбільшого кластера.
11. Пошук кластера (кластерів) для оптимального доповнення поточного фрагмента (оптимального за розміром).
12. Збереження новоутвореного фрагмента розбиття.
13. Формування нового фрагмента розбиття на основі останнього доповнення. Якщо ще не всі фрагменти сформовано, то перехід до кроку 11.
14. Кінець.

Блок-схему алгоритму паралельного розбиття наведено на рис. 2. Згортання виконується декілька разів, але за кожну ітерацію може бути сформовано декілька фрагментів розбиття. Це суттєво прискорює процес формування результатів, а також поліпшує їхню якість за рахунок того, що більша кількість фрагментів ґрунтуватиметься на спільному кластері.

Як показали експерименти, розроблені алгоритми групового перенесення чи обміну забезпечують значно кращі результати оптимізації, ніж перенесення чи обміну одиничними елементами. Проте оптимізаційні процеси заходять в деякі локальні екстремуми. Локальних екстремумів є значно менше при використанні обміну кластерами, ніж поодинокими елементами. Для виходу з них та продовження оптимізації необхідно розробити спеціальні алгоритми. Такими алгоритмами можуть бути «збурення» розв'язку деякими, виділеними за певними критеріями кластерами. Запропоновано та досліджено такі алгоритми збурення:

1. Випадкове збурення довільним одним кластером з обмеженням на кількість елементів. Для цього будують окремо дерева згортання для утворених початковим розв'язком частин та виділяють їхні кластери.

2. Збурення кластером з кращими абсолютним показником перенесення в іншу частину та обмеженням на кількість елементів. Всі кластери та елементи кожної з утворених частин впорядковуються за ефективністю перенесення в іншу з утворених частин – зміною кількості зовнішніх зв'язків.

3. Збурення кластером з кращими відносним показником перенесення в іншу частину та обмеженням на кількість елементів. Всі кластери та елементи кожної з утворених частин впорядковуються за значенням відносного показника ефективності перенесення в іншу з утворених частин – зміною числа зовнішніх зв'язків відносно кількості елементів кластера.

4. Збурення декількома кластерами з обмеженою сумарною кількістю їх елементів, виділеними в попередніх двох пунктах.

Після «збурення» оптимізаційний процес продовжується. Доцільним є також детальне вивчення «пограничних» областей виділених частин, які утворюються елементами, що безпосередньо зв'язують різні частин. Регруповання таких елементів сприятиме виходу з локальних екстремумів та забезпечуватиме можливості подальшої оптимізації.

#### **Експериментальні дослідження**

З метою дослідження ефективності запропонованого алгоритму виконано експериментальний поділ тестових схем  $P_{tm}$  [5] на різну кількість фрагментів. Отримані результати наведено в таблиці. Вони свідчать про те, що алгоритм уможливує поділ схем на необхідну довільну кількість фрагментів. Результати досліджень показали, що алгоритм забезпечує кількість зв'язків між фрагментами розбиття на рівні 3–30 % від загальної її кількості в схемі. Процес формування

розбиття залежить від вибраних критеріїв та методу згортання, кількості фрагментів та допустимого відхилення рівномірності результату, а також від особливостей структури схеми. Спостерігається стійка залежність між складністю схеми та результатами розбиття. В процесі побудови графіків (рис. 3) виявилось, що характер залежності кількості зв'язків розбиття від загальної кількості є подібним для декомпозиції від 2 до 32 частин. На рис. 4 наведено графік продуктивності алгоритму розбиття схеми на дві частини. Експерименті виконано на комп'ютері з процесором Core2Duo 2,33GHz з оперативною пам'яттю 3 Gb.

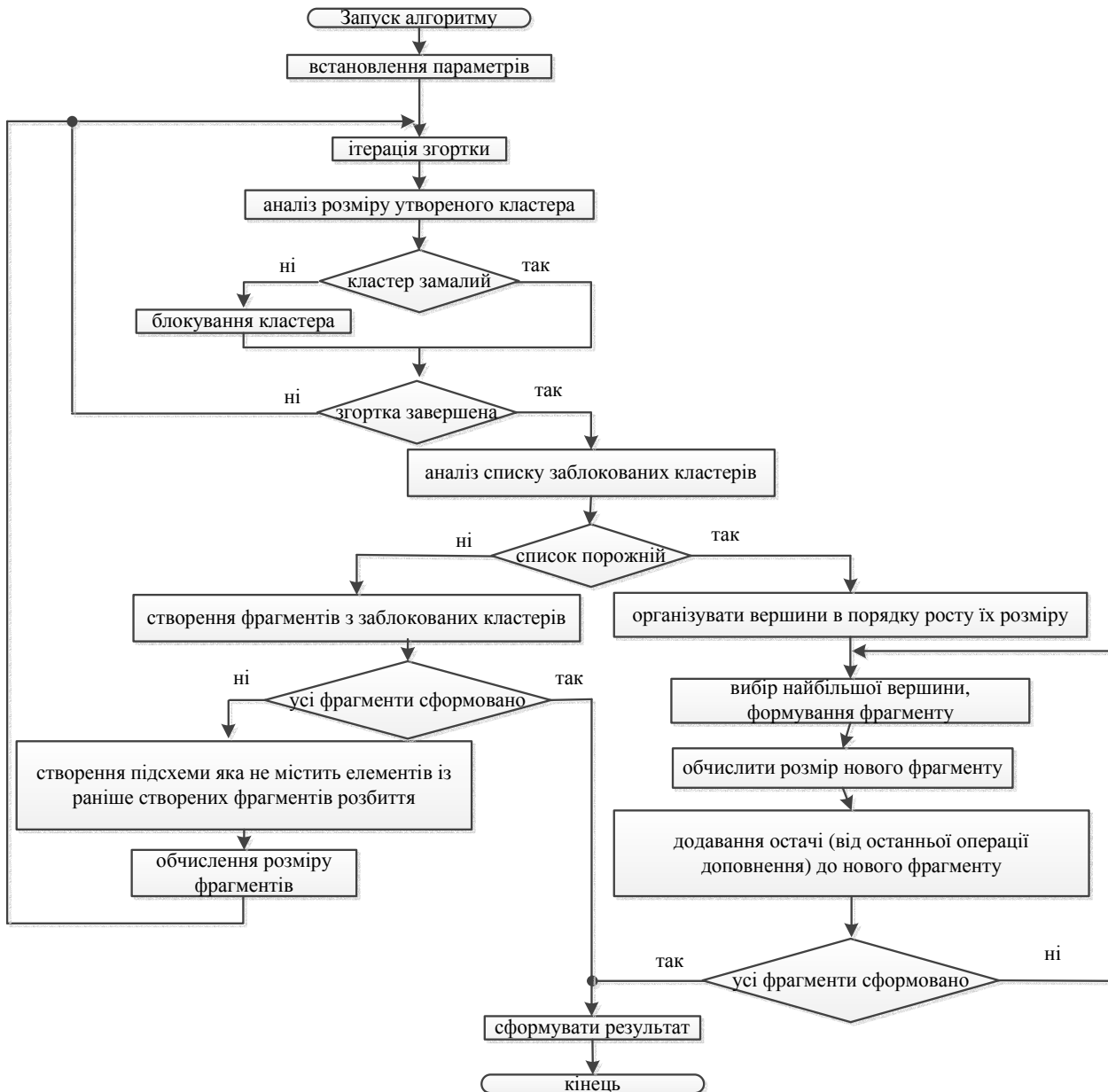


Рис. 2. Блок-схема алгоритму паралельного розбиття

Отримані результати розбиття є початковими розв'язками та вимагають подальшої оптимізації. Для цього розроблено алгоритми [4], які базуються на ідентифікації кластерів кожної з утворених частин, виявлення ефективності їх перенесення в іншу частину схеми чи обміну між двома частинами. Для виявлення кластерів кожної частин використовуються ті самі програмні засоби на основі методу оптимального згортання схеми, що і для отримання початкових розв'язків.

Схема			Кількість фрагментів										
Файл	К-ть елементів	К-ть зв'язків	2	3	4	5	6	7	8	9	10	16	32
ibm01	12505	14111	387	816	1065	1236	1421	1461	1746	1791	1893	2408	2887
ibm02	19341	10584	1478	1662	2748	2425	3568	4483	4388	4460	4695	6278	7169
ibm03	22852	27401	3164	4447	5404	5256	5783	6144	6335	6543	6665	7330	8267
ibm04	27219	31970	2305	3702	5204	5305	6748	7113	7784	7706	7704	8627	10305
ibm05	28145	28446	4290	6892	7692	8084	8528	8631	9250	9692	9998	10499	11127
ibm06	32331	34826	3620	4303	4737	5883	6117	6769	7000	6988	7370	8426	9662
ibm07	45638	48117	2499	5232	6669	6832	7888	8171	8376	9287	8765	10805	12509
ibm08	51022	50513	4009	5952	7496	7982	8454	9087	9612	9844	10328	11072	12806
ibm09	53109	60902	2809	4674	5878	6874	7881	7414	7900	8019	8540	9810	13391
ibm10	68684	75196	4833	7253	7873	9339	10472	12231	12118	12449	14087	15376	19120
ibm11	70151	81454	5307	7647	8945	9029	10004	11144	12134	11000	12311	16293	19481
ibm12	70438	77240	6858	12764	14415	15642	16859	17841	19251	18982	19743	23699	25204
ibm13	83708	99666	6376	8070	8457	9486	10950	12657	14414	14406	13372	18571	22482
ibm14	147087	152772	11466	14996	18396	19686	19913	20554	22204	24712	23651	28704	35685
ibm15	161186	186608	11566	16089	18165	20857	21582	21786	24397	24131	26044	30413	38209
ibm16	182979	190048	10646	13742	19679	22547	24140	26761	29510	29087	30253	38136	46620
ibm17	184751	189581	14661	21721	27827	31541	36592	35354	38951	39652	42729	49560	58233
ibm18	210340	201920	9947	14014	16327	17718	20522	20821	23504	26836	26784	31047	37843

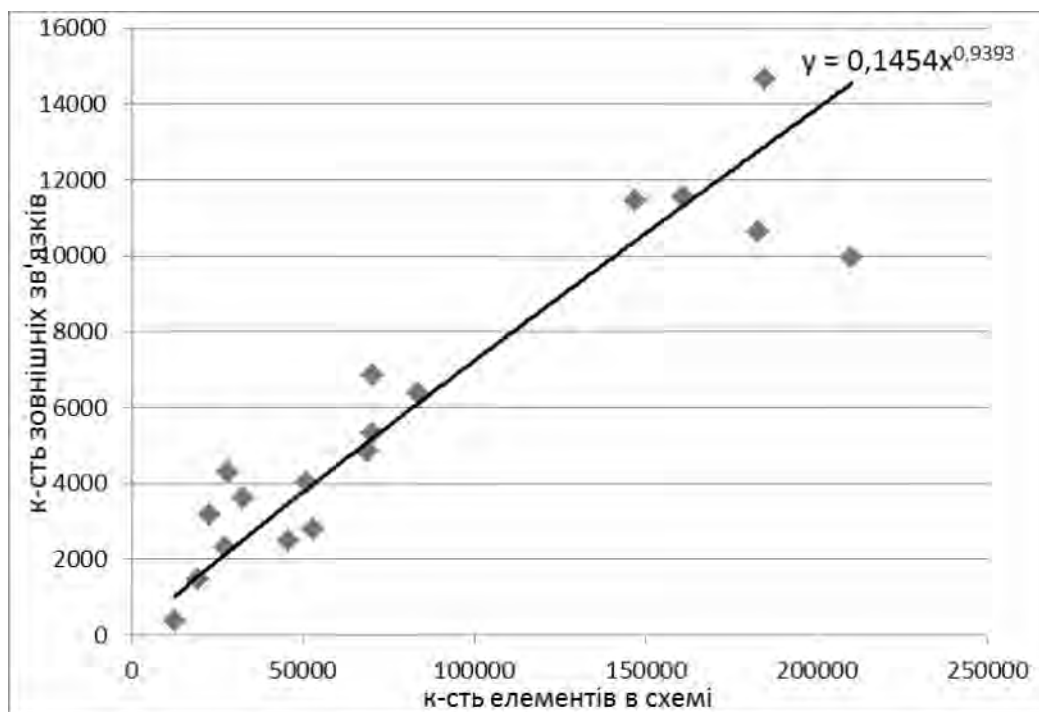


Рис. 3. Залежність кількості зв'язків від кількості елементів схеми

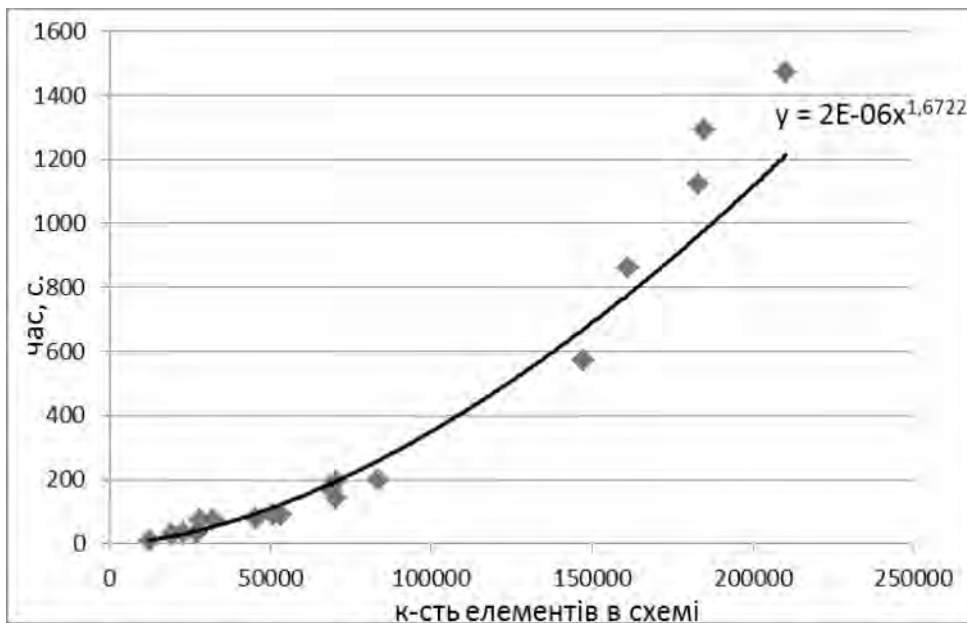


Рис. 4. Час розбиття схем на дві частини

### Висновки

Оптимальна декомпозиція електронних схем високих розмірностей на довільну кількість частин залишається маловивченою. Можливі підходи до її розв'язання все ще є недостатньо дослідженими. Неперервне зростання розмірностей електронних схем, ускладнення характеру взаємозв'язків між компонентами потребує подальшого вдосконалення наявних та розроблення нових підходів.

Запропонований алгоритм паралельного розбиття електронних схем з використанням методу оптимального згортання, виділення ієрархічно вкладених кластерів, аналізу їхніх параметрів та структури взаємного входження не має недоліків інших методів. Це робить його ефективнішим та гнучкішим для застосування. Результати досліджень характеризують цей алгоритм як перспективний засіб розв'язання задачі декомпозиції, хоча він все ще потребує подальшого удосконалення. Важливою особливістю цього алгоритму, що істотно вирізняє його з-поміж аналогів, є придатність до декомпозиції схеми на довільну кількість фрагментів з бажаною кількістю елементів та можливостями врахування інших обмежень. Подальші дослідження будуть спрямовані як на отримання кращих початкових розв'язків, так і на розроблення спеціальних алгоритмів їх оптимізації з виходами з локальних екстремумів.

1. Базилевич Р.П. Решение задач разбиения методом параллельного свертывания / Базилевич Р.П., Ткаченко С.П. // Каунасский политехнический институт им. Антанаса Снечкуса: Вычислительная техника. – Каунас, 1975. – 295 с.
2. Bazylevych R.P., Melnyk R.A. and Rybak O.G. Circuit Partitioning for FPGAs by the Optimal Circuit Reduction Method // VLSI DESIGN, OPA (Overseas Publishers Association) N.V., 2000.
3. Базилевич Р.П. Декомпозиционные и топологические методы автоматизированного конструирования электронных устройств. – Львів: Вища школа, 1981. – 168 с.
4. Базилевич Р., Влах М., Пелих Н. Особливості опрацювання даних для ієрархічної кластеризації складних схем // Вісник Нац. ун-ту “Львівська політехніка” № 672. – Львів, 2010. – С. 215–219 с.
5. Alpert C.J. The ISPD98 circuit benchmark suite. / Alpert C.J. // IBM Austin Research Laboratory: Austin TX 78758. – Monterey, CA USA, 1998.
6. Базилевич Р., Влах М., Пелих Н. Вимушена кластеризація електричних схем з використанням швидкого алгоритму згортки ланцюгами // Вісник Нац. ун-ту “Львівська політехніка” № 686. – Львів, 2010. – С.77-80.