

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ЕКОНОМІЧНИЙ ФАКУЛЬТЕТ  
КАФЕДРА ЕКОНОМІЧНОЇ КІБЕРНЕТИКИ**

**Кваліфікаційна робота  
магістра**

на тему **МОДЕЛЮВАННЯ РАЦІОНАЛЬНОГО РОЗКРОЮ МАТЕРІАЛІВ У  
МЕТАЛУРГІЙНІЙ ПРОМИСЛОВОСТІ**

Виконав: студент II курсу, групи 8.0512-ек  
спеціальності 051 «Економіка»  
освітньої програми «Економічна кібернетика»  
Семілетов Деніс Володимирович  
Керівник: професор кафедри економічної  
кібернетики, д.ф.-м.н., професор Козін І.В.  
Рецензент: завідувач кафедри економічної  
кібернетики, д.е.н., професор Максишко Н.К..

Запоріжжя  
2023



## 6. Консультанти розділів роботи

| Розділ   | Прізвище, ініціали та посада консультанта | Підпис, дата   |                  |
|----------|---|----------------|------------------|
|          |   | завдання видав | завдання прийняв |
| Розділ 1 | д.ф.м.н. проф. Козін І.В.                 | 20.09.2023     | 08.10.2023       |
| Розділ 2 | д.ф.м.н. проф. Козін І.В.                 | 09.10.2023     | 29.10.2023       |
| Розділ 3 | д.ф.м.н. проф. Козін І.В..                | 30.10.2023     | 20.11.2023       |
|          |   |                |                  |
|          |   |                |                  |
|          |   |                |                  |
|          |   |                |                  |

7. Дата видачі завдання 18.09.2023

**КАЛЕНДАРНИЙ ПЛАН**

| № з/п | Назва етапів кваліфікаційної роботи               | Строк виконання етапів роботи | Примітка |
|-------|---|-------------------------------|----------|
| 1.    | Формування мети та завдань кваліфікаційної роботи | 18.09-23.09.2023              |          |
| 2.    | Складання плану роботи                            | 25.09-30.09.2023              |          |
| 3.    | Підготовка I розділу                              | 02.10-21.10.2023              |          |
| 4.    | Підготовка II розділу                             | 23.10-04.11.2023              |          |
| 5.    | Підготовка III розділу                            | 06.11-26.11.2023              |          |
| 6.    | Оформлення висновків та роботи                    | 27.11-30.11.2023              |          |
| 7.    | Підготовка до захисту і нормоконтролю             | 28.11-01.12.2023              |          |

Студент

\_\_\_\_\_

(підпис)

Семілетов Д.В.

Керівник роботи (проекту)

\_\_\_\_\_

(підпис)

Козін І.В.

**Нормоконтроль пройдено**

Нормоконтролер

\_\_\_\_\_

(підпис)

Макаренко О.І.

## РЕФЕРАТ

Кваліфікаційна робота магістра містить три розділи, 113 с., 33 рис., 13 табл., 83 джерела.

Об'єкт дослідження – задачі розкрою матеріалів в металургійній промисловості.

Предмет дослідження – моделі і методи розв'язання задачі розкрою матеріалів в металургії.

Мета роботи – побудова нових методів пошуку субоптимальних розв'язків задачі плоского розкрою стрічки на предмети нескладної геометричної форми на базі генетичних алгоритмів.

Методи дослідження – науковий аналіз, порівняння і групування даних, узагальнення і систематизація, стохастична оптимізація, векторна алгебра та геометрія, чисельні методи.

У роботі запропоновано генетичний алгоритм для вирішення актуальної задачі оптимального розкрою матеріалів у металургійній галузі. Ефективний розкрій дозволяє знизити витрати металу, собівартість продукції та підвищити конкурентоспроможність підприємств. Алгоритм базується на моделюванні процесу біологічної еволюції шляхом генерації розв'язків-особин, їх схрещування, мутації та селекції в популяції з використанням спеціально розробленого декодера для трансформації даних у карти розкрою. Розроблене програмне забезпечення дозволяє задавати вхідні параметри, візуально аналізувати й оцінювати отримані розв'язки. Експериментально підтверджено ефективність методу для широкого класу задач. Запропонований підхід дає змогу оптимізувати розкрій, знизити витрати матеріалів та підвищити рентабельність у металургійній галузі.

РОЗКРІЙ МАТЕРІАЛІВ, МЕТАЛУРГІЯ, ГЕНЕТИЧНИЙ АЛГОРИТМ,  
ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, КАРТИ РОЗКРОЮ МАТЕРІАЛІВ,  
ДЕКОДЕР.

## SUMMARY

The master's qualification work contains three sections, 113 p., 33 fig., 13 table, 83 sources.

The object of the research is the tasks of cutting materials in the metallurgical industry.

The subject of research is models and methods of solving the problem of cutting materials in metallurgy.

The purpose of the work is to develop new methods of searching for suboptimal solutions of the problem of flat cutting of tape into objects of simple geometric shape based on genetic algorithms.

Research methods – scientific analysis, comparison and grouping of data, generalization and systematization, stochastic optimization, vector algebra and geometry, numerical methods.

The paper proposes a genetic algorithm for solving the current problem of optimal cutting of materials in the metallurgical industry. Effective cutting allows you to reduce metal consumption, the cost of production and increase the competitiveness of enterprises. The algorithm is based on modeling the process of biological evolution by generating solution-individuals, their crossing, mutation and selection in the population using a specially developed decoder for data transformation into cut-out maps. The developed software allows you to set input parameters, visually analyze and evaluate the obtained solutions. The effectiveness of the method for a wide class of problems has been experimentally confirmed. The proposed approach makes it possible to optimize cutting, reduce material costs and increase profitability in the metallurgical industry.

DEVELOPMENT OF MATERIALS, METALLURGY, GENETIC ALGORITHM, SOFTWARE, MAPS OF DEVELOPMENT OF MATERIALS, DECODER.

## ЗМІСТ

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

РЕФЕРАТ

SUMMARY

|   |     |
|---|-----|
| ВСТУП .....   | 7   |
| РОЗДІЛ 1 ОСОБЛИВОСТІ ЗАДАЧІ РОЗКРОЮ МАТЕРІАЛІВ В МЕТАЛУРГІЙНОЇ<br>ПРОМИСЛОВОСТІ.....  | 10  |
| 1.1 Сучасний стан та проблеми розкрою матеріалів у вітчизняній металургії.....  | 10  |
| 1.2 Аналіз наукових публікацій з проблематики розкрою матеріалів у металургії.....  | 18  |
| 1.3 Аналіз розрахункової складності задачі розкрою матеріалів в металургійній<br>промисловості.....   | 31  |
| РОЗДІЛ 2 МОДЕЛІ ТА МЕТОДИ РОЗВ’ЯЗКУ ЗАДАЧІ РОЗКРОЮ МАТЕРІАЛІВ   | 40  |
| 2.1 Розв’язок задачі прямокутного розкрою .....   | 40  |
| 2.2 Розв’язок задачі круглого розкрою .....   | 49  |
| 2.3 Аналіз застосування генетичних алгоритмів розв’язку задачі розкрою матеріалів.  | 62  |
| РОЗДІЛ 3 ПОБУДОВА ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ПОШУКУ<br>СУБОПТИМАЛЬНИХ РОЗВ’ЯЗКІВ ЗАДАЧІ ПЛОСКОГО РОЗКРОЮ СТРІЧКИ НА<br>ПРЕДМЕТИ НЕСКЛАДНОЇ ГЕОМЕТРИЧНОЇ ФОРМИ В МЕТАЛУРГІЇ ..... | 76  |
| 3.1 Генетичний алгоритм вирішення задачі плоского розкрою стрічки на предмети<br>нескладної геометричної форми.....   | 76  |
| 3.2 Програмна реалізація розв’язку задачі плоского розкрою стрічки на предмети<br>нескладної геометричної форми на базі генетичний алгоритму .....                              | 80  |
| 3.3 Оцінка ефективності розробленого алгоритму для розв’язання задач плоского<br>розкрою стрічки на предмети складної геометричної форми.....                                   | 85  |
| ВИСНОВКИ.....   | 101 |
| ПЕРЕЛІК ДЖЕРЕЛ ТА ПОСИЛАННЯ .....   | 104 |
| Додаток А Приклад задання пакета замовлень .....  | 112 |

## ВСТУП

Актуальність задачі раціонального плоского розкрою матеріалів на сьогодні зростає у зв'язку з потребою оптимізації виробничих процесів та скороченням витрат на виробництво в умовах жорсткої конкуренції. З огляду на підвищення цін на матеріали та комплектуючі, а також зростання тарифів на енергоносії, питання раціонального використання сировини набуває особливого значення.

Плоский розкрій – це оптимізаційна задача пошуку щільного розміщення множини менших за розміром деталей, що називаються заготовками, на великих об'єктах, які називаються заготовками. Вона ставиться у багатьох галузях промисловості. Мета постановки та вирішення такої задачі – максимальне використання заготовок або мінімізація втрат розкрою, тобто скорочення об'єму незайнятого простору тощо. Це завжди було особливо важливим у галузях масового виробництва, адже невеликі вдосконалення розміщення можуть призводити до значної економії матеріалу та зниження собівартості продукції.

Ручний підбір розміщень – процес суб'єктивний, неефективний за великої кількості деталей і, відповідно, далеко не завжди дає задовільний результат. За розкрою дорогих матеріалів чи матеріалів з особливими фізичними властивостями доцільність мінімізації втрат має очевидне економічне обґрунтування.

Необхідність розробки раціональних методів розкрою була обґрунтована ще у 40-х роках ХХ ст. Перші розробки належать академіку Л.В. Канторовичу. Також слід відзначити роботи професора Ю.Г. Стояна, котрий розробив ефективні евристичні алгоритми для ортогонального розкрою.

Серед зарубіжних дослідників вагомий внесок зроблено групою під керівництвом професора Е. Бенса з Південно-Каліфорнійського університету,

яка запропонували метаевристики та пристосовний генетичний алгоритм для розв'язання складних задач двовимірного розкрою.

Проте незважаючи на численні розробки, проблема раціонального розкрою лишається не до кінця вирішеною. Актуальним залишається створення універсальних та ефективних алгоритмів, котрі б працювали для різних видів матеріалів та різної геометрії деталей. Останнім часом низка металургійних та машинобудівних підприємств вперше придбали або замінили застаріле обладнання для лазерного розкрою листового металу. Лазерний розкрій має низку переваг: висока точність, можливість зменшити втрати металу за рахунок програмного управління, висока якість, автоматизованість тощо.

Область застосування лазерної різки дуже широка: від деталей для залізничних вагонів до побутової техніки та декоративних елементів. Для такого сучасного обладнання доцільна розробка ефективного методу плоского розкрою на набори деталей різних форм, що вирішуватиме задачу для різної геометричної форми деталей.

Метою роботи є побудова нових методів пошуку субоптимальних розв'язків задачі плоского розкрою стрічки на предмети нескладної геометричної форми на базі генетичних алгоритмів.

Для реалізації мети поставлено та виконано такі завдання:

- проаналізовано застосування задач розкрою матеріалів в металургійній промисловості;
- досліджено застосування генетичних алгоритмів розв'язку задачі розкрою матеріалів в металургійній промисловості;
- розроблено генетичний алгоритм вирішення задачі плоского розкрою стрічки на предмети нескладної геометричної форми;
- розроблено програмне забезпечення для розв'язання поставлених задач плоского розкрою стрічки на предмети нескладної геометричної форми;



– досліджено ефективність розробленого програмного забезпечення та генетичного алгоритму розв’язку задачі розкрою матеріалів в металургійній промисловості ;

Наукова і практична новизна кваліфікаційної роботи полягає у розробці генетичного алгоритму вирішення задачі плоского розкрою стрічки на предмети нескладної геометричної форми та програмного забезпечення, що імплементує даний алгоритм, які дозволяють оптимізувати споживання металопрокату на етапі розкрою, скоротити втрати та покращити рентабельність виробництва металопродукції.

Результати кваліфікаційної роботи були представлені на ІХ Всеукраїнської науково-практичній конференції студентів і молодих вчених «Сталий розвиток економіки на засадах ресурсоефективності».

## РОЗДІЛ 1

### ОСОБЛИВОСТІ ЗАДАЧІ РОЗКРОЮ МАТЕРІАЛІВ В МЕТАЛУРГІЙНОЇ ПРОМИСЛОВОСТІ

1.1 Сучасний стан та проблеми розкрою матеріалів у вітчизняній металургії

Ефективне використання матеріальних ресурсів посідає чільне місце серед ключових факторів конкурентоспроможності та рентабельності сучасних металургійних підприємств на світовому ринку. У структурі собівартості металопродукції вартість металопрокату та напівфабрикатів становить левову частку. Тому мінімізація відходів сировини при розкрої, штампуванні та інших стадіях обробки має вирішальне значення для прибутковості галузі.

В умовах постійного зростання цін на метали та матеріали ощадливе витрачання ресурсів набуває особливої гостроти. Тільки за останні 2-3 роки спостерігається стрімке подорожчання всіх видів прокату чорних металів. Збільшення собівартості сировини призводить до зниження рентабельності та конкурентоспроможності металургійних комбінатів. Отже, питання комплексної оптимізації та модернізації технологій розкрою листового, сортового та фасонного прокату набуває особливої гостроти та актуальності.

За різними оцінками, втрати металу лише на стадії розкрою та розділення заготовок сягають 10-20%, а в окремих випадках до 30% від загального обсягу оброблюваного прокату. Це призводить до колосальних збитків та недоотриманої вигоди для меткомбінатів. Тому вдосконалення технологій та операцій раціонального розкрою листових, рулонних та трубних матеріалів має величезне техніко-економічне значення.

На превеликий жаль, технології та обладнання для розкрою металопрокату, що використовуються на більшості українських металургійних комбінатів, істотно поступаються сучасним світовим стандартам. Значна частка підприємств задіює застарілі верстати, успадковані ще з радянських часів.

Зокрема, надто поширеними лишаються малопродуктивні ножиці та слабоавтоматизовані пиляльні верстати з низькою швидкістю обробки заготовок. Внаслідок цього до 70% операцій розкрою виконується вручну. Практично не використовуються високошвидкісні та високоточні технології різання матеріалів (лазер, плазма, гідроабразив). Процеси проектування схем розташування деталей на листах, стрічках чи трубах також здійснюються за допомогою застарілого програмного забезпечення або взагалі “вручну”.

Такий стан справ не дозволяє комплексно підійти до оптимізації процесів розкрою та досягти мінімально можливого рівня відходів і втрат матеріалу. За різними оцінками, через недосконалі технології 10-30% металопрокату йде у відходи вже на початковій стадії виробничого процесу, що призводить до величезних фінансових втрат.

Окрім того, бракує фахівців та інженерів, які б володіли знаннями з сучасних систем автоматизованого проектування та раціонального розкрою. Переважна більшість кадрів працює за застарілими методиками та стандартами. Підприємствам бракує розуміння важливості впровадження новітніх технологій задля підвищення ефективності виробництва.

Таким чином, проблема технологічного відставання галузі розкрою матеріалів в українській металургії стоїть вкрай гостро. Для її вирішення необхідні системні зусилля з комплексної модернізації обладнання, оновлення програмного забезпечення, навчання персоналу та переосмислення підходів до організації виробництва з боку власників та менеджменту металургійних комбінатів.

Вихідною сировиною в металургійному виробництві слугує різноманітний прокат чорних металів. Залежно від подальшого призначення розрізняють кілька основних типів: листовий, сортовий, трубний, фасонний прокат, а також катанку й бруски. Кожен з цих видів має свої особливості, які безпосередньо впливають на вибір технологій подальшої обробки та розкрою.

Листовий прокат застосовується у виробництві металовиробів, деталей машин, конструкцій тощо. Він характеризується порівняно невеликою

товщиною (від долі міліметрів до десятків сантиметрів) при ширині до 2 метрів. Листи виготовляють з різних марок сталі - вуглецевих і легованих, інструментальних, нержавіючих, жароміцних та інших. Механічні властивості кожної з цих сталей можна цілеспрямовано змінювати шляхом легування та термообробки. Це дає змогу задовольнити широке коло вимог за міцністю, пластичністю, зносостійкістю тощо. Листовий прокат може надходити у рулонах масою 10-25 тон.

Сортовий прокат складається з простих геометричних профілів – куточків, швелерів, балок, таврів тощо. Застосовується для виготовлення металоконструкцій, рам, ферм, вагонів. Найбільш характерними параметрами є висота поперечного перерізу 50-400 мм та товщина полиці/стілки 4-20 мм. Випускається масово з вуглецевих конструкційних та легованих сталей.

Трубний прокат включає гаряче- та холоднодеформовані труби діаметром від декількох мм до 1420 мм, товщиною стінки до 40 мм, довжиною до 12,5 м. Застосовується у теплоенергетиці, машинобудуванні, будівельній сфері. Матеріали – вуглецеві та леговані сталі, чавуни тощо...

Отже, металургійна промисловість виробляє широку номенклатуру прокату різних геометричних форм, розмірів та матеріалів. Їх раціональний розкрій потребує урахування всіх цих особливостей для обґрунтованого вибору найбільш підходящої технології та оптимальних режимів обробки з метою мінімізації відходів сировини.

Існує кілька базових технологій, які традиційно застосовуються для розкрою металопрокату в металургійному виробництві.

Різання ножицями. Найбільш поширений та простий метод. Використовується для розкрою листового прокату невеликої товщини (до 16 мм), переважно з вуглецевих сталей. Здійснюється на гільйотинних ножицях або прес-ножицях. Характерна продуктивність 5-10 м/хв. Переваги – простота, низька вартість. Недоліки – низька точність (зазор до 3 мм), порівняно невисока продуктивність.

Різання на пиляльних верстатах. Можлива обробка широкого спектру листових і профільованих матеріалів товщиною до 300 мм. Продуктивність до 40 м/хв. Висока точність (до 0,5 мм). Але потребує кваліфікованого персоналу та суттєвих витрат на ріжучий інструмент.

Термічне різання. Розрізняють автогенне, лазерне та плазмове різання. Застосування стисненої дуги або вузько-спрямованого лазерного променя дає змогу обробляти листи товщиною до 200 мм. Висока якість кромки різки, продуктивність. Але вимагає значних інвестицій в устаткування.

Гідро-абразивне та водоструминне різання. Використовує енергію сфокусованого потоку абразивних частинок або води під високим тиском до 6000 атм. Дає змогу обробляти складно-профільовані заготовки різних матеріалів (зокрема високоміцні сталі) товщиною до 200 мм, з високою швидкістю та якістю. Потрібні суттєві інвестиції в устаткування.

Серед сучасних методів розкрою металопрокату все більшого поширення набуває лазерна технологія. Завдяки високій концентрації потужності, локальності та швидкодії процесу вона дозволяє досягти неперевершеної якості та чистоти різки при високій продуктивності.

На відміну від традиційних механічних методів, лазер формує абсолютно гладку кромку без задирок чи грату, з вузькою (0,1-0,5 мм) зоною термічного впливу. Швидкість обробки сягає десятків метрів за хвилину. Можлива автоматизація та програмування складних траєкторій.

З огляду на ці переваги, все більше сучасних меткомбінатів впроваджують лазерні технології розкрою. Їх особливо ефективно поєднувати з традиційними методами - для чистової обробки після грубого попереднього розділення заготовок. Такий комбінований підхід дозволяє оптимально використовувати можливості різних технологій для досягнення максимальної продуктивності та економічності процесів розкрою.

Лазерне різання металу базується на використанні висококонцентрованого пучка монохроматичного когерентного випромінювання, що генерується спеціальним квантовим генератором - лазером. У фокальній плямі потужності

лазерного променю досягають колосальних значень - до  $105-107 \text{ Вт/см}^2$ , що призводить до миттєвого розплавлення і випаровування (абляції) матеріалу. Завдяки локальності та швидкості процесу термічний вплив на оточуючі ділянки мінімальний.

Переваги лазерного різання - висока концентрація енергії, чистота різу, мінімальна ширина розплаву (0,1-0,5 мм) та зона термічного впливу (до 1 мм), можливість обробки важкодоступних ділянок, простота автоматизації. Сучасні лазерні комплекси з ЧПК забезпечують швидкість до 25-30 м/хв, точність  $\pm 0,1$  мм. Це дає змогу ефективно різати листи з вуглецевих, легованих, високоміцних і нержавіючих сталей завтовшки до 20-30 мм.

Розрізняють 3 основні типи лазерів: газові  $\text{CO}_2$  лазери (10,6 мкм), твердотільні на основі алюмо-ітрієвого гранату з домішками неодиму (Nd:YAG) та волоконні лазери, довжина хвилі яких становить 1,07-1,1 мкм. Кожен тип має свої особливості за потужністю, якістю променю, ресурсом роботи, вимогами до живлення тощо.

Процес різання відбувається в інертних ( $\text{N}_2$ , Ar) або активних ( $\text{O}_2$ ) газах під тиском. Газ під дією лазерного променю іонізується, утворюючи плазму, що сприяє ефективнішому випаровуванню металу.

Характеристики лазерної техніки представлені у табл. 1.1 – 1.3 в порівнянні також ефективності роботи різного обладнання [76].

Для різання використовуються матеріали: метали, деревина, пластик, пароніт, скло, кераміка тощо. Площа розкрою залежно від пристрою до 2,5х6 м. Застосування лазера у 10 разів дешевше порівняно з традиційним способом виробництва.

За наведеними таблицями можна сформулювати наступний висновок: переваги лазерного різання – висока точність при високій швидкості обробки; недоліки – порівняно висока вартість обладнання та менша максимальна товщина матеріалу, що розрізається.

Таблиця 1.1 – Порівняльна характеристика технологій розкрою металопрокату

| Параметри                                     | Плазмова        | Гідроабразивна                       | Киснева                          | Лазерна                         |
|---|-----------------|--------------------------------------|----------------------------------|---------------------------------|
| Максимальна товщина матеріалу, t(мм)          | ~150            | ~300                                 | > 1000                           | ~40                             |
| Швидкість різання, м/хв (маловуглецева сталь) | 0,6 при t=50 мм | 0,04 при t=50 мм<br>0,5 при t=1,6 мм | 0,2 при t=50 мм<br>0,6 при t=3мм | 0,6 при t=25 мм<br>4 при t=2 мм |
| Ширина різку, мм                              | 4-7             | > 2                                  | ~8                               | 0,1-1                           |

Таблиця 1.2 – Якісне співвідношення інших способів різання відносно до лазерного

| Параметри                 | Плазмова     | Гідроабразивна  | Киснева      |
|---------------------------|--------------|-----------------|--------------|
| Вартість обладнання       | Дуже дорожче | Порівнянна      | Дорожче      |
| Вартість обслуговування   | Порівнянна   | Порівнянна      | Дуже дорожче |
| Швидкість                 | Швидше       | Дуже повільніше | Повільніше   |
| Якість                    | Дуже гірше   | Дуже гірше      | Дуже гірше   |
| Максимальна товщина листа | Товстіше     | Товстіше        | Товстіше     |
| Ширина різку              | Ширше        | Ширше           | Ширше        |
| Зона термічного впливу    | Дуже більше  | Менше           | Більше       |

Таблиця 1.3 – Характеристики товщин матеріалів, що розрізаються, при потужності лазера 5 кВт

| № | Вид матеріалу                                      | Параметри, мм. |
|---|--|----------------|
| 1 | листовий прокат з вуглецевих та легированих сталей | до 40          |
| 2 | мідь   | 5              |
| 3 | алюмінієві сплави                                  | 12             |
| 4 | латунь   | 12             |
| 5 | нержавіюча сталь                                   | 25             |
| 6 | склотекстоліт                                      | 30             |

Із допомогою лазерних установок можна вирізати найрізноманітніші деталі та вироби з будь-яких матеріалів, наприклад, літери для рекламних конструкцій з пластику чи комплектуючі для залізничних вагонів, автомобілів, мотоциклів, побутової техніки з металу. Сучасні лазерні комплекси дають змогу з високою точністю обробляти листові матеріали, виконуючи розкрій і вирізання складних форм. Гнучкі налаштування дозволяють нарізати як поодинокі деталі,

так і великі партії однакових виробів, що добре підходить для потреб промислового виробництва. Лазерна різка металу та пластику відрізняється високою швидкістю, мінімальним тепловим впливом та чистотою крайок в порівнянні з іншими видами різання. Тому все більше підприємств застосовують цю передову технологію.

Лазерне різання сталевих листів товщиною до 6 мм по складному контуру є найбільш поширеним технологічним процесом лазерної обробки у промисловості. Його застосовують для вирізання таких деталей, як прокладки, кронштейни, панелі, приладові щитки, двері, декоративні ґрати, дискові пили. Найчастіше застосовується в металургійній промисловості для таких видів робіт:

- виготовлення коробчатих деталей (металошаф, сейфів, стелажів, підставок, столів, барних стійок, дверей, коробів для гральних автоматів, обладнання, приладових панелей, піддонів машин і верстатів);
- виготовлення деталей інтер'єру (годинників, портретів, ікон, сувенірів);
- ландшафтний дизайн і рекламні конструкції;
- виготовлення декоративних огорож та оздоблення з орнаментом будь-якої складності;
- виготовлення складних монтажних конструкцій;
- виготовлення запчастин до вітчизняної та імпортової техніки. В металургії, машинобудуванні, приладобудуванні лазерний розкрій металевих листів здійснюється у заготівельних цехах та сервісних металоцентрах.

Лазерна технологія різання металопрокату попри численні переваги має певні особливості, які необхідно враховувати при організації виробництва. Зокрема, вартість високопотужних волоконних чи твердотільних лазерів сягає сотень тисяч доларів. Витратні матеріали - захисні стекла, відбивачі, направляючі, а також технічні гази (азот, кисень, аргон) для здійснення процесу - досить дорогі. Термін окупності комплексних лазерних систем розкрою складає 3-6 років. Тому для ефективного використання цього устаткування потрібно докласти максимум зусиль на етапі підготовки та планування виробництва.



Особливого значення в цьому разі набуває оптимізація саме процесу розкладання листового матеріалу на окремі деталі - тобто проектування раціональних схем їх взаємного розміщення. Адже метою є досягнення мінімально можливого рівня відходів сировини та виходу готової продукції.

Для цього необхідно врахувати багато факторів: габарити та форму листового прокату, геометрію деталей, їх кількість, послідовність розкладання, можливість поворотів тощо. Всі ці змінні утворюють колосальний простір пошуку можливих комбінацій. Ручний підхід до проектування схем розкрою тут вже не ефективний. На допомогу приходять спеціальні САПР - системи автоматизованого проектування, які дозволяють за лічені секунди знаходити найраціональніші схеми розміщення елементів шляхом оптимізаційних обчислень.

Застосування лазерного комплексу в поєднанні з САПР для розкрою дають змогу комплексно підійти до організації вискоєфективного та економного виробництва на основі найсучасніших технологій. Правильний вибір обладнання та оптимальне планування розкладки деталей дозволяють протягом короткого часу окупити високовартісне лазерне устаткування за рахунок скорочення відходів сировини та витрат на випуск готової продукції.

Ефективне використання сучасного високоточного лазерного обладнання для розкрою металу неможливе без залучення систем автоматизованого проектування технологічних процесів. Адже оптимальне та раціональне розташування елементів є запорукою мінімізації відходів сировини та собівартості готової продукції. Втім наявні на ринку САПР часто орієнтовані на інші галузі, мають універсальний функціонал і не дозволяють повною мірою врахувати всі особливості саме задачі розкрою листових та рулонних матеріалів у металургії.

Тому питання розробки спеціалізованих алгоритмів та програмного забезпечення для оптимізації схем розкладки елементів в межах заданої форми листа з урахуванням усього розмаїття вимог металургійного виробництва є вкрай

актуальним. Адже потенціал скорочення відходів і витрат при цьому сягає десятків відсотків.

Ефективне програмне рішення в цій сфері має реалізовувати низку спеціалізованих алгоритмів: розташування прямокутних елементів, фігур неправильної форми, при наявності обмежень і заборонених зон, з урахуванням можливості поворотів та відбитків. Для глобальної оптимізації доцільно застосовувати евристичні методи: генетичні алгоритми, імітацію відпалу, мурашині алгоритми тощо. Вони дозволяють ефективно знаходити наближені до оптимальних розв'язки складних багатовимірних задач за прийнятний час.

Результати розрахунків мають візуалізуватися у зручному графічному режимі, а також у вигляді керуючих програм для подальшої реалізації розкрою на відповідному обладнанні (лазерні комплекси, прес-ножиці тощо).

Таким чином розробка спеціалізованого програмного забезпечення оптимізації розкрою матеріалів з широким набором алгоритмів та функцій адаптованих під задачі металургійної промисловості є вкрай необхідним засобом радикального підвищення ефективності виробництва та конкурентоспроможності підприємств галузі.

## 1.2 Аналіз наукових публікацій з проблематики розкрою матеріалів у металургії

Задача оптимального розкрою матеріалів посідає одне з центральних місць серед завдань підвищення ефективності виробництва в металургійній промисловості. Від її вирішення безпосередньо залежать показники ресурсозбереження, продуктивності праці, собівартості та рентабельності підприємств чорної металургії.

Традиційно задача раціонального використання сировини є ключовою для матеріаломістких галузей промисловості. Адже вартість матеріалів (металопрокату, напівфабрикатів тощо) часто складає лівову частку в собівартості готової продукції. Відповідно, будь-які технологічні та організаційні заходи,

спрямовані на економію матеріальних ресурсів та мінімізацію відходів, мають велике значення.

Особливо гостро ця проблема стоїть в металургії. Адже вартість металопродукції як сировини для інших галузей промисловості невинно зростає. До того ж, в Україні питома вага металургійного комплексу у валовому внутрішньому продукті є однією з найвищих у світі. Відповідно, будь-які резерви підвищення ефективності галузі мають величезне макроекономічне значення.

Оптимальне використання листового прокату, мінімізація обрізків і відходів дозволяє, з одного боку, знизити собівартість виробництва готової продукції. А з іншого – підвищити обсяг товарного випуску без нарощування потужностей та видобутку сировини.

Крім того, скорочення обсягів втрат матеріалу сприяє не лише економічному, але й екологічному ефекту. Адже зменшується навантаження на природне середовище в результаті видобутку та переробки металургійної сировини.

Всі ці чинники зумовлюють першочергову увагу фахівців металургійних підприємств до оптимізації процесів розкрою листового та сортового прокату. Разом з тим, впровадження сучасних математичних методів та інформаційних технологій дозволяє значно підвищити ефективність вирішення цієї задачі.

Зокрема, розвиток обчислювальної техніки та алгоритмічних підходів відкрив можливості для створення спеціалізованих оптимізаційних систем розкрою матеріалів для потреб металургії.

Задача розкрою матеріалів є надзвичайно актуальною для металургійних підприємств в сучасних умовах. Її вирішення дозволяє суттєво підвищити ефективність використання металопрокату та знизити матеріальні витрати виробництва.

По-перше, раціональний розкрій листового та рулонного прокату забезпечує мінімізацію відходів, що мають значну вартість для металургійних комбінатів. За різними оцінками, оптимізація розкрою дозволяє скоротити обсяг

відходів на 5-15%. При масштабах виробництва це означає економію тисяч та мільйонів гривень на одному заводі.

По-друге, ефективний розкрій збільшує випуск товарної продукції без нарощування обсягів виробництва сировини. Це також має значний економічний ефект та підвищує рентабельність металургійних підприємств.

По-третє, за рахунок скорочення обсягів відходів, які потребують утилізації, знижується й негативний вплив металургійної промисловості на довкілля. Цей аспект також набуває все більшого значення в сучасному світі.

По-четверте, ефективний розкрій сталевого прокату сприяє підвищенню задоволеності замовників та поліпшенню іміджу металургійної компанії на ринку. Адже раціональне використання сировини дозволяє забезпечити постійний контроль якості готової продукції та стабільність поставок.

Серед українських вчених, що досліджували цю тематику, можна відзначити І.В. Анікіна, А.М. Долгова, В.О. Залого. Вони аналізували задачу розкрою сталевого прокату на металургійних комбінатах, пропонували методи її вирішення з урахуванням технологічних особливостей галузі.

Серед зарубіжних фахівців слід згадати таких науковців, як Wäscher G., Haims M., Cui Y. Вони розглядали задачу розкрою як складну оптимізаційну NP-повну задачу і пропонували для її вирішення евристичні алгоритми, методи лінійного, нелінійного та динамічного програмування.

Задачі розкрою та пакування описують моделі, що складаються з геометричних комбінацій великих об'єктів (наприклад, листів, смуги) та малих предметів (наприклад, пакунка замовлень, що складається з набору прямокутних заготовок різних розмірів). У випадку задачі пакування великі об'єкти визначені як порожні й мають бути заповнені малими предметами пакунка замовлень. Задачі розкрою, натомість, характеризуються великими об'єктами, які потрібно розрізати на малі предмети. Залишкові об'єкти, наявні в моделі, але не належать до пакунка замовлень, називають втратою розкрою. Мета більшості задач розкрою та пакування – мінімізувати втрату розкрою чи збиток.

Як зазначено в огляді [84], німецький дослідник Дікофф акцентував на сильному зв'язку між проблемами розкрою та пакування, що впливає з дуальності матеріалу та простору. Тобто задачу розкрою можна розглядати як пакування простору, зайнятого малими предметами, у великі об'єкти. І навпаки, задачу пакування можна розглядати як розкрій великих об'єктів на малі предмети. У зв'язку з цим терміни «розкрій» і «пакування» надалі вживаються як синоніми.

За останні 30 років досліджень проблеми розкрою та пакування широко описані в літературі. Вони часто виникають у багатьох галузях промисловості та не обмежені сектором виробництва. Проблеми пакування, наприклад, що зустрічаються в дослідженні операцій та фінансовому секторі, найчастіше мають абстрактну форму. Через це різноманіття проблем та прикладних галузей публікації на подібні теми з'являються в літературі під найрізноманітнішими назвами. Аналіз показує, що багато з них мають одну й ту саму основну логічну структуру, хоча з ними стикаються у різних прикладних галузях.

Щоб полегшити інформаційний обмін між різними дисциплінами, Дікофф виявив спільні характеристики та атрибути проблем розкрою і пакування та запропонував систему класифікації. Він провів різницю між проблемами пакування, що включають просторові та непросторові виміри. Перша група складається із задач вирізання, пакування чи завантаження, в яких визначено до трьох вимірів у Евклідовому просторі (наприклад, проблема розкрою смуги, завантаження транспортного засобу та завантаження піддону). Інша група охоплює 11 абстрактних проблем розкрою та пакування, до яких включено непросторові виміри типу ваги, часу чи фінансових вимірів.

Система класифікації Дікоффа описує чотири важливі характеристики проблем пакування і розкрою [79]:

а) найважливіша характеристика – вимірність, що визначає мінімальне число вимірів, необхідних для опису геометрії задачі. Проблеми з більш ніж трьома вимірами виходять, коли мають місце непросторові виміри, наприклад, час чи вага;

б) вид замовлення описує, чи повинні всі предмети бути запаковані, чи тільки деякі;

в) асортимент об'єктів. Ця характеристика проводить різницю між проблемами, де об'єкти, в які відбувається пакування, ідентичної та різної форми;

г) асортимент предметів стосується форми та числа предметів, якими запаковуються об'єкти. Задачу можна розглядати стосовно порівняно великих чи малих за чисельністю наборів предметів; наборів зіставних чи якісно різних за розміром предметів; предметів різної чи однакової геометричної форми.

Мета задачі пакування – ефективний розподіл фігур в обмеженій області без перекриття. Отже, складність задачі пакування суворо пов'язана з геометричною формою предметів, які будуть запаковані. Щодо геометрії можна розрізняти два типи форм: регулярні (правильні) форми, які описані кількома параметрами (наприклад, прямокутники, кола) та неправильні форми, що включають асиметрії та увігнутості.

Регулярні проблеми пакування значною мірою стосуються пакування набору прямокутників у прямокутний об'єкт. Пакування неправильних форм також відоме як «nesting», наприклад, у суднобудівній промисловості та як проблема планування розкладки лекал у текстильній промисловості (рис. 1.1).

У двовимірному прямокутному пакуванні на основі геометрії предметів можна розрізняти такі типи розміщення [82]. У випадку розкрою об'єктів на регулярні предмети план розкрою може містити розрізи ортогональні сторонам об'єкта (смути, листа) та неортогональні (рис. 1.2). За ортогонального розкрою додатково розрізняють гільйотинне та негільйотинне розміщення.

Гільйотинний розкрій відрізняється обмеженням, яке приписує, що план розкрою не повинен містити порожніх областей, крім кінця об'єкта (рис. 1.3). Цей тип задачі розкрою зустрічається, наприклад, в скляній та паперовій промисловості. За негільйотинного розкрою предмет може бути розміщений в будь-якій відповідній за розміром області (рис. 1.4).

Задачі пакування зустрічаються в різних прикладних галузях, де ставляться різні обмеження та цілі. Загалом, задачі пакування смуги і прямокутника можуть розрізнятися залежно від типу об'єкта.

У паперовій та текстильній промисловості сировина є у формі рулонів. Відповідно, процес пакування повинен прагнути до зменшення довжини розміщення. Ця задача відома як пакування смуги. Рисунок 1.5 ілюструє задачу пакування смуги прямокутниками.

Пакування прямокутника розглядає пакування множини прямокутників і зустрічається в галузях, де смуга матеріалу є у формі листів. Мета зазвичай - знайти набір прямокутників такий, щоб вмістити всі частини списку замовлень при мінімізації загального використаного матеріалу. Залежно від додатку, листи можуть бути ідентичні або мати різний розмір (рис. 1.6).

В одновимірному випадку пакування прямокутника – розподіл предметів, чия ширина є ідентичною ширині прямокутників. Отже, для такого процесу пакування важливий лише один вимір. Використання предметів різної ширини призводить до задачі двовимірного пакування прямокутника. Крім того, задачі регулярного пакування прямокутника також часто зустрічаються у тривимірному вигляді в задачах завантаження піддону і контейнера.

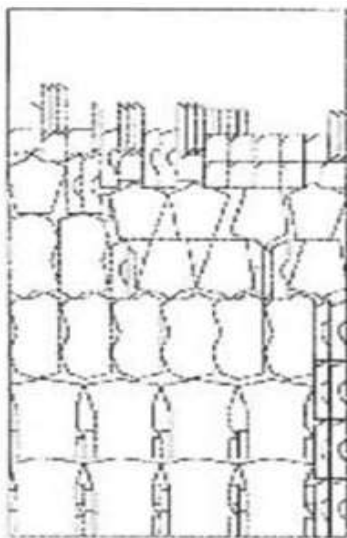


Рисунок 1.1 – Проблема нерегулярної упаковки

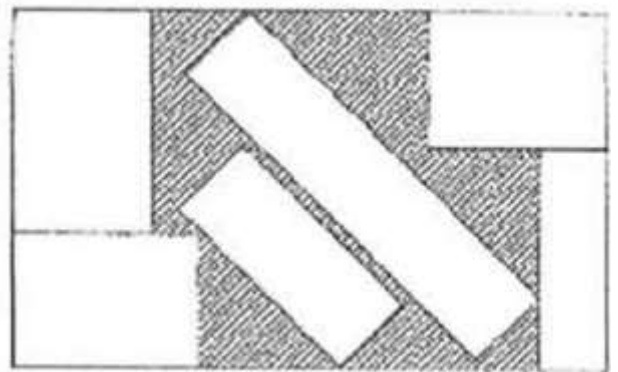


Рисунок 1.2 – Неортогональне розміщення

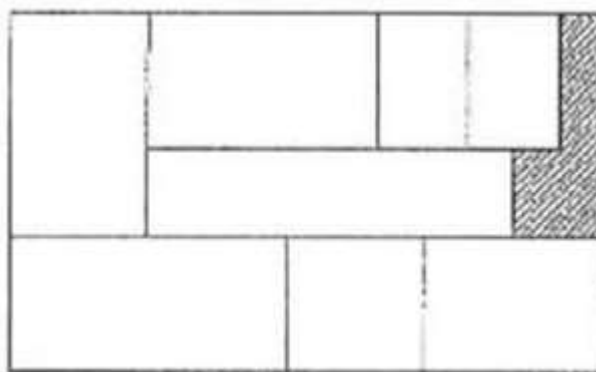


Рисунок 1.3 – Гільйотинне розміщення

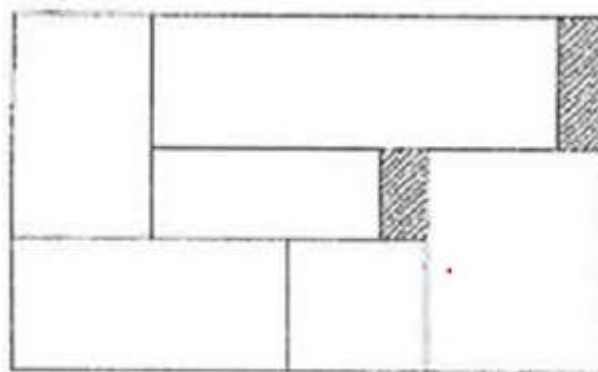


Рисунок 1.4 – Негільйотинне розміщення

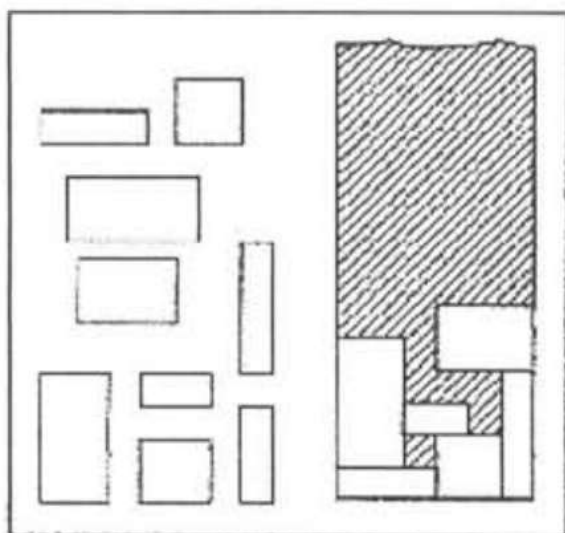


Рисунок 1.5 – Проблема упакування двовимірної смуги

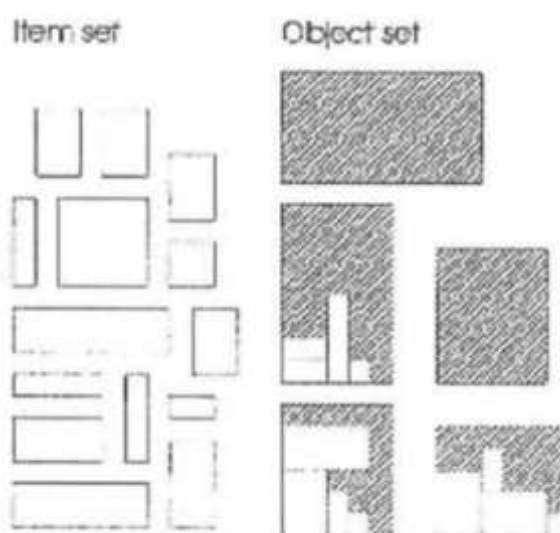


Рисунок 1.6 – Проблема двовимірної упаковки листів

Крім мети зведення збитку до мінімуму, існує низка інших факторів, що визначають остаточне розміщення. Найчастіше великий вплив на кінцевий результат мають різні обмеження на властивості матеріалу, процес розкрою, аспекти планування та інші вимоги.

У промисловості листового металу неоднорідні властивості металу типу орієнтації зерна обмежують число можливих орієнтацій предметів. Якщо відбудуться операції згинання, то предмети можуть бути повернуті лише під певним кутом. Тканина також має деякі напрямні властивості і можливий шаблон, який обмежує орієнтацію частин лише під кутами  $0^\circ$  і  $180^\circ$ . Не завжди можливо



дзеркально відбити частини, оскільки тканина може мати різні властивості на іншому боці.

Природні матеріали типу шкіри складаються з ділянок різної якості. Відмінність може бути зумовлена дефектами і кольоровими відмінностями. Процес розкрою повинен узгоджувати необхідну якість предметів із відповідними зонами якості на об'єкті.

Методика розкрою, що використовується для отримання предметів, має великий вплив на бажане розміщення. Залежно від технології вирізання (наприклад, лазерного та плазмового розкрою, штампування) потрібна певна мінімальна відстань між частинами. Цей параметр згадується як ширина перемички [79]. Щоб не пошкодити самі предмети, необхідна деяка відстань між суміжними фігурами.

У процесі штампування, якщо ширина перемички занадто мала, має місце тенденція до ковзання ріжучої кромки. Інший важливий параметр, що визначає процес розкрою - повна довжина шляху ріжучого елемента. Розміщення можуть бути оптимізовані так, щоб вирізання всіх предметів могло бути проведене з мінімізацією загального шляху різання.

Крім того, для подальшого процесу виробництва може бути важлива сама послідовність, в якій вирізані предмети. Якщо обсяг замовлення великий, то спеціальний розподіл предметів щодо об'єктів полегшує процес виробництва. Послідовність предметів також може бути важлива для пакування та пересилання.

Геометричні обмеження і обмеження ваги можуть вимагати, щоб предмети були запаковані в певному визначеному порядку.

У металургійній промисловості часто виникає задача оптимального розкрою матеріалу [84]. Матеріал може надходити на виробництво у вигляді певних одиниць – профілів, листів, рулонів, брусків тощо, які надалі розкроюються на заготовки заданих розмірів та комплектності. Для випадку, коли основною характеристикою матеріалу є його довжина, задача особливо актуальна, коли довжина профілю (стрижня, стрічки, стержня) відрізняється від довжини одержуваних заготовок не більше ніж на порядок.

Проблема раціонального розкрою довгомірних матеріалів містить у собі низку різних за своєю математичною постановкою задач. Смісловий зміст полягає в тому, щоб забезпечити найменші витрати матеріалу при отриманні з нього необхідного комплекту заготовок.

На математичну модель і вибір методів її розв'язання впливають різні чинники, серед яких найбільш істотними є характер виробництва, вид матеріалу, що надходить, особливості обладнання тощо. Основні чинники, що впливають на модель розкрою, представлені на рис. 1.7.

Залежно від виду матеріалу задачі розкрою підрозділяються на задачі розкрою мірного матеріалу, тобто профілів певної, заздалегідь відомої, стандартної довжини, і задачі розкрою немірного матеріалу, довжина якого випадкова і наперед невідома. Їх відносять відповідно до задач детермінованого і стохастичного розкрою.

Залежно від виду виробництва виділяють задачі розкрою в умовах масового (багатосерійного) виробництва і задачі розкрою в умовах одиничного (дрібносерійного) виробництва. В масовому виробництві число потрібних комплектів заготовок досить велике. Внаслідок циклічного характеру масового виробництва, відсутня вимога отримання певної кількості заготовок на заданий момент часу, в результаті чого ми отримуємо безперервну модель лінійного розкрою. Умови одиничного та/або дрібносерійного виробництва з жорстким дотриманням комплектності заготовок породжують цілочисельну модель. Тому методи лінійного програмування, успішно застосовувані за умов масового виробництва, стають тут непридатними.

Крім того, з точки зору обчислювального ефекту, принциповим є відмінність у постановці самої задачі розкрою. Виділяють задачу генерування розкрою і задачу планування розкрою. Зрозуміло, що перша задача є одним з етапів розв'язання другої. Слід, однак, зауважити, що розв'язання задачі генерування розкрою зводиться до пошуку локального екстремуму, який в загальному випадку може не збігатися з глобальним екстремумом, знаходження якого потрібне для розв'язання задачі планування розкрою.



Рисунок 1.7 – Чинники, що впливають на модель розкрою

Задача планування розкрою зазвичай виникає на етапі технологічної підготовки виробництва, розв’язання ж задачі генерування розкрою найчастіше потрібно вже на етапі оперативного управління технологічним процесом. Проте, в тому і іншому випадку математичні моделі виявляються зануреними в конкретний виробничий фон, врахування якого може призвести до нових моделей і методів розв’язання. Усі дослідники відзначають, що класичні математичні постановки задачі розкрою не завжди придатні для застосування на металургійних підприємствах через специфіку технологічного процесу.

Основними особливостями задачі, що ускладнюють її вирішення, є:

– необхідність врахування анізотропії властивостей металопрокату вздовж і впоперек напрямку прокатки. це накладає обмеження на можливі напрямки розкрою.

- наявність дефектних зон на заготовках, які потребують вирізання.
- різні вимоги до якості готових виробів та необхідність групування заготовок за якісними характеристиками.
- потреба врахування теплових деформацій листів під час подальшої термічної обробки готових деталей.

Для вирішення задачі розкрою в металургійній галузі з урахуванням цих особливостей потрібні спеціалізовані оптимізаційні моделі та алгоритми.

Одним з найпоширеніших підходів є застосування евристичних алгоритмів. На практиці часто використовують жадібні евристики та методи локального пошуку. Вони дозволяють отримати прийнятні, хоч і не оптимальні розв'язки за прийнятний час.

До евристичних методів відносять алгоритми найближчого розташування, найпершого розміщення, зменшення відстаней та інші. Їх основна ідея полягає у розміщенні деталей на листовому прокаті в певній послідовності за деякими евристичними правилами.

До методів локального пошуку в задачах розкрою належать метод табу-пошуку, метод імітації відпалу, генетичні алгоритми. Їх можна використовувати для пошуку кращих варіантів розкрою серед отриманих евристичним способом.

Іншим підходом є застосування математичного програмування: лінійного, цілочисельного, нелінійного або динамічного. Такі методи дають оптимальний за якимось критерієм розв'язок, але потребують значних обчислювальних ресурсів.

Також активно досліджуються методи комбінаторної оптимізації: гілок і меж, послідовного включення обмежень, розгалуження та оцінки, рою частинок та інші. Вони є ефективними для задач великої розмірності.

Усі ці методи потребують модифікації та адаптації для застосування в металургійній галузі з урахуванням її особливостей: анізотропії, дефектних зон, групування за якістю і так далі.

Вважається перспективним розробка гібридних методів, які поєднували б сильні сторони різних підходів. Наприклад, спочатку застосовувати швидкий евристичний пошук, а потім оптимізувати результат математичним програмуванням.

На основі цих методів створюються спеціалізовані програмні системи оптимального розкрою для потреб металургійних підприємств. Вони дозволяють істотно підвищити ефективність використання металопрокату.

Задача плоского розкрою стрічки на предмети нескладної геометричної форми є важливим завданням оптимізації виробництва в металургії.

Серед українських вчених, що займалися цією проблематикою, можна відзначити роботи Жукова Є.П., Ямшинського М.М., Крижанівського Є.І. Вони досліджували задачу розкрою сталеві стрічки з урахуванням анізотропії механічних властивостей металу та наявності дефектів.

Серед зарубіжних науковців варто згадати таких фахівців як Wang C., Murata K., Liu D. В своїх публікаціях вони аналізували різні алгоритмічні та обчислювальні методи для вирішення NP-складних задач двовимірного розкрою.

Усі дослідники підкреслюють, що ефективний розкрій стрічки дозволяє суттєво знизити відсоток відходів та підвищити продуктивність на металургійних підприємствах. В той же час, ця задача є досить складною через свою комбінаторну природу.

Під час розкрою сталеві стрічки на нескладні деталі (прокат, листи, смуги тощо) застосовують як точні методи оптимізації (лінійне та цілочисельне програмування), так і евристичні алгоритми.

Серед останніх найбільш ефективними є метод найщільнішої упаковки елементів, жадібні евристики, метод послідовного розміщення та інші. Вони дозволяють швидко будувати прийнятні, хоч і наближені розв'язки задачі.

Перспективним є поєднання евристичних та точних методів – спочатку швидкий пошук наближеного рішення евристикою, а потім його поліпшення за допомогою математичного програмування. Серед методів вирішення задачі розкрою стрічки особливе місце посідають еволюційні та генетичні алгоритми. Вони є відносно новим підходом оптимізації, який набув поширення в останні два десятиліття.

Серед зарубіжних науковців, які займалися розробкою генетичних алгоритмів для задач розкрою, відомі Bennell J.A., Oliveira J.F., Jakobs J. Вони

адаптували класичну схему генетичних алгоритмів для ефективного подолання проблеми комбінаторного вибуху в двовимірних задачах розміщення та пакування.

Серед українських вчених внесок у застосування генетичних алгоритмів для задачі розкрою металопрокату зробили Анікін І.В., Павлов О.А., Рамазанов С.К. Вони вдосконалювали оператори генетичних алгоритмів з урахуванням особливостей металургійного виробництва.

Сутність генетичного підходу до оптимізації розкрою полягає в наступному:

- формується початкова популяція випадкових варіантів розкрою (особин);
- за допомогою генетичних операторів (схрещування, мутації) формуються нові варіанти розкрою на основі найкращих представників попереднього покоління;
- процедура повторюється доти, доки не буде отримано задовільний варіант або вичерпано ліміт ітерацій.

Такий підхід дозволяє ефективно досліджувати всю множину можливих варіантів розкрою та знаходити наближені до оптимального рішення без повного перебору. Це є особливо важливим для задач великої розмірності та складної геометрії, до яких відноситься і розкрій металопрокату.

Важливою перевагою генетичних алгоритмів є можливість їх ефективної паралелізації на багатоядерних та багатопроцесорних системах. Це дозволяє істотно пришвидшити пошук оптимального або близького до оптимального рішення.

Отже, генетичні та еволюційні алгоритми є потужним інструментом для вирішення складних задач оптимізації, зокрема плоского розкрою стрічки в металургії. Їх розвиток та поєднання з традиційними методами оптимізації відкриває широкі можливості для створення ефективних програмних систем оптимального розкрою металопрокату.

### 1.3 Аналіз розрахункової складності задачі розкрою матеріалів в металургійній промисловості

У сучасній металургійній промисловості, ефективність використання матеріалів є ключовим фактором для зниження витрат та підвищення продуктивності. Однією з важливих задач у цьому контексті є оптимізація процесів розкрою матеріалів. Ця задача вимагає глибокого аналізу розрахункової складності, адже вона впливає на вибір стратегій та методів обробки даних для ефективного управління ресурсами.

Важливість цього аналізу підкреслюється численними дослідженнями, як в Україні, так і за кордоном. Українські науковці, такі як Анікін І.В., Долгов А.М., Залога В.О., внесли значний вклад у розробку методологій і алгоритмів, спрямованих на оптимізацію процесів розкрою. З іншого боку, міжнародні дослідження Wäscher G., Haims M., Cui Y. надають широкий огляд методів та підходів, що використовуються у світовій практиці. Це включає в себе розробку комплексних програмних рішень, що дозволяють автоматизувати процеси розкрою і забезпечити максимальну ефективність використання матеріалів.

Розрахункова складність задачі розкрою матеріалів у металургійній промисловості полягає у визначенні найбільш ефективного способу розрізання сировинних матеріалів, таких як металеві листи або блоки, для мінімізації відходів та оптимізації використання ресурсів. Ця задача є складною з кількох причин.

По-перше, це багатовимірність. Багатовимірність у задачі розкрою матеріалів в металургійній промисловості є одним з ключових аспектів, який робить цей процес особливо складним. Ця багатовимірність означає, що для визначення оптимального способу розкрою необхідно враховувати ряд різноманітних параметрів.

Розмір матеріалу. Різні вироби вимагають різних розмірів металевих листів чи блоків. Визначення оптимального розміру матеріалу для конкретного замовлення є важливим для мінімізації відходів і збільшення ефективності використання ресурсів.

Форма матеріалу. Метал може бути потрібно розкроїти у різні форми, в залежності від вимог до кінцевого продукту. Наприклад, округлі, прямокутні або складніші форми вимагають різних підходів до розкрою, що значно ускладнює процес планування.

Якість матеріалу. Різні види металів мають різну твердість, гнучкість, та інші властивості, які впливають на процес розкрою. Наприклад, тверді метали вимагають більш потужного обладнання для різання, тоді як м'які метали потребують більш делікатного підходу.

Вимоги до кінцевого продукту. Різні продукти можуть мати різні технічні характеристики та стандарти якості, які необхідно враховувати при плануванні розкрою. Наприклад, в деяких випадках може бути потрібна висока точність розмірів або особливі вимоги до поверхні матеріалу.

Кожен з цих параметрів вносить свої унікальні виклики у процес розкрою, і їх потрібно ретельно аналізувати та балансувати для досягнення оптимального результату. Це вимагає складних алгоритмів та програмного забезпечення, яке може обробляти великі обсяги даних та швидко адаптуватися до змінних умов виробництва. В результаті, ефективне управління цією багатовимірністю є ключовим фактором для підвищення продуктивності та ефективності в металургійній промисловості.

По-друге, це оптимізація. Метою є знаходження оптимального плану розкрою, який максимізує використання матеріалу та мінімізує відходи. Для досягнення цієї мети необхідно вирішувати ряд складних задач

Розробка алгоритмів оптимізації. Визначення найбільш ефективного способу розкрою вимагає застосування алгоритмів, які можуть обробляти великі обсяги даних та враховувати численні змінні та обмеження. Ці алгоритми часто повинні вирішувати NP-складні задачі, тобто такі, вирішення яких у найгіршому випадку вимагає часу, який експоненціально зростає зі збільшенням розміру вхідних даних.

Мінімізація відходів. Одним з основних завдань оптимізації є зменшення кількості відходів, що генеруються під час розкрою матеріалу. Це включає в себе



вибір таких патернів розкрою, які дозволять використовувати кожен лист або блок матеріалу з максимальною ефективністю.

Адаптивність до різних сценаріїв. Алгоритми оптимізації повинні бути гнучкими, щоб адаптуватися до різних типів матеріалів, виробів та виробничих вимог. Наприклад, виробничий план для тонкого листового металу може істотно відрізнятись від плану для важкого промислового блоку.

Оптимізація часу та ресурсів: Крім мінімізації відходів, ефективний план розкрою повинен також враховувати час, необхідний для вирізання, та використання обладнання. Це означає, що ідеальний план оптимізує не тільки матеріальні ресурси, але й час та енергію, витрачені на виробництво.

Застосування цих складних алгоритмів оптимізації вимагає високого рівня експертизи у сфері математики та програмування, а також глибокого розуміння виробничих процесів. Вони включають в себе не лише стандартні алгоритми, але й передові методи, такі як:

- генетичні алгоритми та евристичні методи. Ці методи дозволяють знаходити рішення для оптимізаційних задач, які занадто складні для точного алгоритмічного розв'язання. Генетичні алгоритми, наприклад, використовують принципи еволюції для пошуку найкращих рішень, а евристичні методи забезпечують прийнятні рішення за прийнятний час;

- моделювання та симуляція: Іноді для знаходження оптимального рішення необхідно спочатку моделювати різні сценарії розкрою, щоб побачити, як вони вплинуть на використання матеріалів та відходи. Це дозволяє визначити найбільш ефективні стратегії перед їхнім впровадженням у виробництво.

- інтеграція з системами управління виробництвом (ERP): Ефективні рішення з оптимізації також вимагають інтеграції з іншими системами управління виробництвом, щоб забезпечити повний облік всіх ресурсів, витрат і виробничих потреб.

Кожен з цих елементів вносить важливий вклад у розробку оптимального плану розкрою, який не тільки максимізує використання матеріалу та мінімізує

відходи, але й враховує час, витрати на обладнання, та інші ключові виробничі фактори.

По-третє, це варіативність матеріалів. Різні види металів мають різні властивості, що впливають на процес розкрою. Наприклад, твердість, гнучкість, і теплопровідність кожного металу можуть істотно впливати на стратегію розкрою.

По-четверте, це технологічні обмеження. Обладнання, яке використовується для розкрою, має свої обмеження, такі як максимальний розмір аркуша, який воно може обробляти, або точність різання. Урахування цих технологічних обмежень є критично важливим для планування ефективних та економічно вигідних процесів розкрою. Це вимагає комплексного підходу, що включає аналіз поточних можливостей обладнання, визначення потреб у модернізації або оновленні, а також планування ресурсів та навчання персоналу для оптимального використання наявних технологій.

Переходячи від аналізу практичних особливостей задачі оптимального розкрою матеріалів у металургії до теоретичного підґрунтя проблеми, слід звернути увагу на її обчислювальну складність.

З математичної точки зору задача розкрою належить до класу оптимізаційних NP-складних проблем. Це пов'язано з її комбінаторною природою: експоненційно великою кількістю можливих варіантів розміщення заготовок на листах металопрокату, з яких потрібно обрати найкращий за критерієм мінімізації відходів.

Такі задачі дискретної оптимізації характерні тим, що на сьогодні не відомо жодних ефективних точних алгоритмів їх розв'язання. Тобто алгоритмів, які дозволяли б гарантовано знаходити оптимальний результат за поліноміальний час. Навіть за умови стрімкого зростання потужності обчислювальної техніки перебір усіх можливих варіантів лишатиметься практично нездійсненним.

Ця фундаментальна властивість NP-складності впливає з теорії алгоритмів та обчислювальної складності. Вона стосується не лише задачі розкрою в металургії, а й багатьох інших практичних оптимізаційних проблем в різних

сферах: від розподілу ресурсів та складання розкладів до проектування схем та логістичних задач.

Усвідомлення цієї фундаментальної складності дає ключ до розуміння необхідності розробки спеціальних евристичних методів та наближених алгоритмів для ефективного розв'язання задачі розкрою матеріалів у реальних умовах металургійних виробництв. У загальному випадку задачі розкрою є NP-складні [25], оскільки навіть частковий випадок задачі розкрою зводиться до задачі про 3-розбиття. Тому не існує точного алгоритму для розв'язання поставленої задачі за поліноміальний час. Основним методом розв'язання NP-повних задач є повний перебір допустимих варіантів та вибір з них оптимального. Проте, незважаючи на трудомісткість знаходження оптимального результату, на практиці найчастіше виникає необхідність його отримати: це задачі розкрою в дрібносерійному та одиничному виробництві, задачі залишку в масовому та багатосерійному виробництві.

Наведемо кілька прикладів дискретних оптимізаційних задач. Нехай є кілька міст і відомі попарні відстані між містами. Два міста вважаються сусідніми, якщо є дорога, що з'єднує ці міста і не проходить через інше місто. Потрібно знайти найкоротший шлях між певною парою міст (задача про найкоротший шлях). Ще один приклад: є  $p$  верстатів і  $t$  деталей, кожен деталь можна обробляти на будь-якому верстаті, але час обробки деталі на одному верстаті може відрізнятися від часу її обробки на іншому верстаті. Припустимо, що для кожної пари верстат-деталь ці часи задані. Потрібно так організувати виробництво деталей, тобто розмістити деталі по верстатах таким чином, щоб сумарний час роботи був найменшим (задача оптимального призначення).

Нарешті, однією з найпопулярніших дискретних оптимізаційних задач є наступна задача. Мандрівник хоче об'їхати  $p$  міст, побувавши в кожному рівно по одному разу, і повернутися в початкове місто, затративши при цьому мінімальну суму на поїздку. Витрати на поїздку складаються з витрат на переїзди між парами міст, причому ці витрати заздалегідь відомі (задача комівояжера).

Усі наведені вище задачі мають низку спільних властивостей, характерних для дискретних оптимізаційних задач. По-перше, у кожній задачі є лише скінченне число варіантів (шляхів між містами, способів розподілу деталей по верстатах, маршруту пересування мандрівника), з яких потрібно здійснити вибір. По-друге, кожному варіанту співставлена певна числова характеристика (довжина шляху, сумарний час роботи, вартість поїздки). По-третє, потрібно вибрати варіант, числова характеристика якого досягає екстремуму.

Найбільш очевидний спосіб розв'язання подібних задач – це повний перебір усіх варіантів. Однак цей спосіб найменш вдалий, оскільки всі цікаві з практичної точки зору ситуації виникають саме тоді, коли число можливих варіантів надзвичайно велике. Повний перебір усіх варіантів потребував би настільки великого часу, що став би практично нереалізованим навіть на найшвидкодійючих комп'ютерах. На щастя, для багатьох важливих задач дискретної оптимізації існують методи розв'язання, набагато економніші, ніж повний перебір.

За усталеною термінологією розрізняють масові та індивідуальні задачі. Всі вищенаведені приклади дають зразки саме масових задач. Індивідуальна задача отримується з масової за допомогою фіксації набору умов, наприклад, з масової задачі комівояжера вийде індивідуальна задача, якщо зафіксувати число міст і визначити витрати на переїзди між усіма парами міст [1].

Кожна масова задача (надалі просто задача) характеризується розміром. Розмір задачі слугує мірою кількості вхідних даних і подається одним або кількома цілочисельними параметрами. Наприклад, розмірністю задачі комівояжера природно вважати число  $p$  міст, які збирається відвідати мандрівник. Під алгоритмом розуміється загальний, крок за кроком виконуваний спосіб отримання розв'язку даної задачі.

Зауважимо, що існує кілька формалізацій поняття алгоритму (машини Тьюрінга, рекурсивні функції, нормальні алгоритми Маркова), проте їх розгляд виходить далеко за рамки даної роботи. Для визначеності можна вважати, що алгоритм вважається програмою на деякій мові високого рівня.

Для оцінки якості алгоритмів можна застосовувати різні критерії. Однією з найважливіших характеристик алгоритму є часова складність у гіршому випадку. Нехай розмір масової задачі визначається одним цілочисельним параметром  $n$ . Розглянемо всі індивідуальні задачі розміру  $n$  і позначимо через  $t(n)$  максимальне число дій, яке необхідно для розв'язання будь-якої такої індивідуальної задачі. Функція  $t(n)$  – це і є часова складність алгоритму в гіршому випадку (або просто складність алгоритму) [1].

Під дією, що її виконує алгоритм, розумітимемо виконання простої «операції», зазвичай апаратно реалізованої на будь-якому комп'ютері, а саме, будь-якої арифметичної операції, операції порівняння, пересилання тощо.

Зрозуміло, що за такого визначення дії складність алгоритму залежить від конкретного виду машинних команд. Тому нас завжди цікавитиме лише асимптотична складність алгоритму, тобто порядок зростання складності за умови, що розмір задачі необмежено зростає.

При порівнянні швидкості зростання двох невід'ємних функцій  $f(n)$  і  $g(n)$  зручно використовувати такі позначення. Говоритимемо, що  $f(n) = O(g(n))$ , якщо існують такі додатні сталі  $c$  і  $N$ , що  $f(n) \leq c \cdot g(n)$  при всіх  $n \geq N$ . У цій самій ситуації можна використовувати і позначення  $g(n) = \Omega(f(n))$  [1]. Наприклад, справедливі співвідношення:  $\log_2 n = O(n)$   $n = \Omega(\log_2 n)$ . Важливість поняття складності алгоритму добре ілюструють наступні таблиці [1].

Таблиця 1 побудована з припущенням, що один крок роботи алгоритму потребує для свого виконання 1 мілісекунду. Як видно з таблиці 1, при збільшенні часу з 1 секунди до 1 години, тобто у  $3,6 \cdot 10^9$  разів, розмір задачі, що розв'язується алгоритмом складності  $2^n$ , збільшується лише у  $21/9$  разів, а для алгоритму складності  $n$  – рівно у  $3,6 \cdot 10^9$  разів.

Може здатися, що зростання швидкості обчислень, викликане появою нових комп'ютерів, зменшить значення ефективних, тобто тих, що мають «невелику» складність 17 алгоритмів. Проте, як видно з табл. 1.4, насправді відбувається саме протилежне. Зі зростанням швидкодії комп'ютерів стає можливим розв'язання задач дедалі більшого розміру, і саме від складності

алгоритму залежить, наскільки збільшення швидкості комп'ютера впливає на збільшення розміру задачі.

Таблиця 1.4 – Швидкість алгоритмів

| Складність алгоритму | Максимальний розмір задачі |                |                  |
|----------------------|----------------------------|----------------|------------------|
|                      | за 1 с                     | за 1 хв        | за 1 год         |
| $n$                  | 1000                       | $6 \cdot 10^4$ | $3,6 \cdot 10^9$ |
| $n \log_2 n$         | 140                        | 4893           | $2 \cdot 10^5$   |
| $n^2$                | 31                         | 244            | 1897             |
| $n^3$                | 10                         | 39             | 153              |
| $2^n$                | 9                          | 15             | 21               |

Розглянемо таблицю 1.5. Нехай для розв'язання певної задачі є алгоритм складності  $n^3$ . Тоді десятикратне збільшення швидкодії комп'ютера дозволить збільшити розмір задачі, що розв'язується за 1 хвилину, у 2,15 рази. Перехід до алгоритму зі складністю  $n^2$  дозволить розв'язати задачу у 6 разів більшого розміру (див. таблицю 1). Якщо взяти за одиницю часу 1 годину, то порівняння буде ще більш вражаючим.

Таблиця 1.5 – Швидкість обчислювальної техніки

| Складність алгоритму | Макс. розмір задачі, розв'язуваний за од. часу |  |  |
|----------------------|--|--|--|
|                      | на сучасних ЕОМ                                | на ЕОМ зі швидкістю у 10 разів більшою | на ЕОМ зі швидкістю у 1000 разів більшою |
| $n$                  | $K$  | $10K$                                  | $1000K$                                  |
| $n \log_2 n$         | $L$  | майже $10L$                            | майже $1000L$                            |
| $n^2$                | $M$  | $3,16M$                                | $31,6M$                                  |
| $n^3$                | $N$  | $2,15N$                                | $10N$                                    |
| $2^n$                | $P$  | $P+3,3$                                | $P+9,97$                                 |

Зауважимо, що для алгоритму складності  $n^2$  десятикратне збільшення швидкості комп'ютера додає до розміру розв'язуваної задачі лише три одиниці,

тоді як для алгоритму складності  $n$  відбувається збільшення у три рази, а для алгоритму складності  $n^2$  – у десять разів.

З практичної точки зору ціннішими, бажанішими є поліноміальні алгоритми нижчих степенів, що забезпечують повільне зростання обсягів обчислень за значного зростання розмірів задач. Саме такі алгоритми 18 слід шукати в першу чергу, і саме до них належать два важливих питання: у яких випадках пошук виявиться успішним, чи в усіх випадках успіх може бути досягнутий? Відповідь на них намагається дати теорія NP-повноти, що набула розвитку в останні 10-15 років та робить істотний внесок у розуміння проблем обчислювальної складності.

Ця теорія дає визначення поняття NP-повних задач і зараховує їх до найскладніших з тієї точки зору, що для них немає поліноміальних детермінованих алгоритмів пошуку точного розв'язку [12, 71, 73].

Проблема прямокутного розкрою є NP-повною [84]. Оскільки круглий, нерегулярний чи тривимірний розкрій є складнішими, то ці версії проблеми тим паче можуть бути кваліфіковані як NP-повні.

Залежно від застосування в задачі розкрою можуть з'являтися різні обмеження. Додавання обмежень може лише додати їй складності і, отже, підлеглі обмеженням версії також є NP-повними.

Згідно з визначенням, клас NP-повноти має таку важливу характеристику: усі відомі на даний час алгоритми для знаходження оптимальних розв'язків задачі вимагають низки обчислювальних кроків, число яких зростає експоненціально з розміром задачі, тобто швидше, ніж поліноміально [12, 71, 73].

Пошук точного алгоритму для розв'язання таких задач не має сенсу, оскільки вважається неможливим знайти будь-який ефективний оптимальний розв'язок. Тому при розв'язанні NP-повних проблем замість точних алгоритмів розглядаються альтернативні підходи, які не гарантують знаходження оптимального розв'язку, але, втрачаючи в якості розв'язку, отримують ефективність у обчисленнях. Ці міркування приймалися як базові при розв'язанні різних задач розкрою і пакування та вели до розробки алгоритмів апроксимації.

## РОЗДІЛ 2

### МОДЕЛІ ТА МЕТОДИ РОЗВ'ЯЗКУ ЗАДАЧІ РОЗКРОЮ МАТЕРІАЛІВ

#### 2.1 Розв'язок задачі прямокутного розкрою

Проблемі прямокутного розкрою та упаковки присвячено величезну кількість літератури. Цю задачу розглянуто в багатьох постановках і вирішено різними способами. Одна з причин успіху дослідників у цій галузі - особливості геометрії задачі. Упаковка прямокутних предметів у прямокутні області за певної різноманітності предметів дозволяє досягати дуже високих показників щільності упаковки. У даній роботі розглядається і вирішується задача двовимірного ортогонального негільйотинного розкрою напівнескінченної смуги на набір прямокутників різного розміру.

*Змістовна постановка задачі:* Є прямокутна смуга заданої ширини та необмеженої довжини, а також набір прямокутних предметів. Ширина смуги, кількість і розміри предметів задані. Потрібно визначити параметри оптимального розташування предметів у смугі за таких умов:

- ребра упакованих предметів паралельні ребрам смуги (наприклад, як на рис. 2.1, рис. 2.2; приклад недопустимого розташування на рис. 2.3);
- упаковані предмети не перетинаються один з одним (приклад недопустимого розташування на рис. 2.4);
- упаковані предмети не перетинаються зі сторонами смуги.

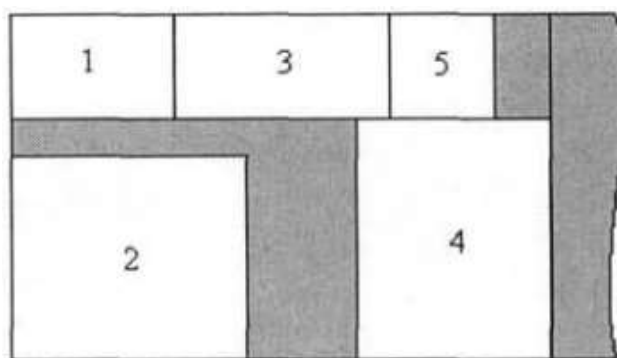


Рисунок 2.1 – Допустимий (нещільний) розкрій



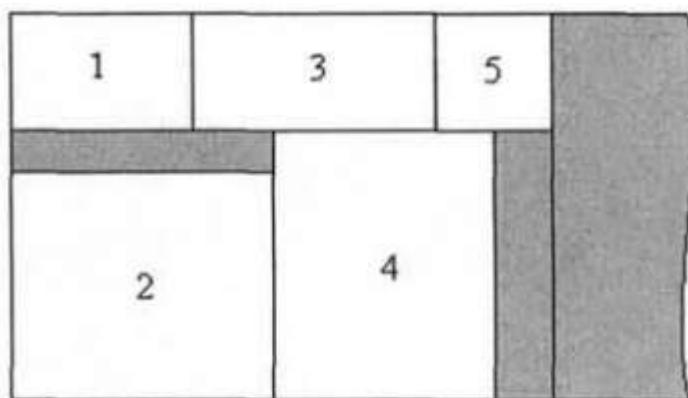


Рисунок 2.2 – Допустимий (щільний) розкрій

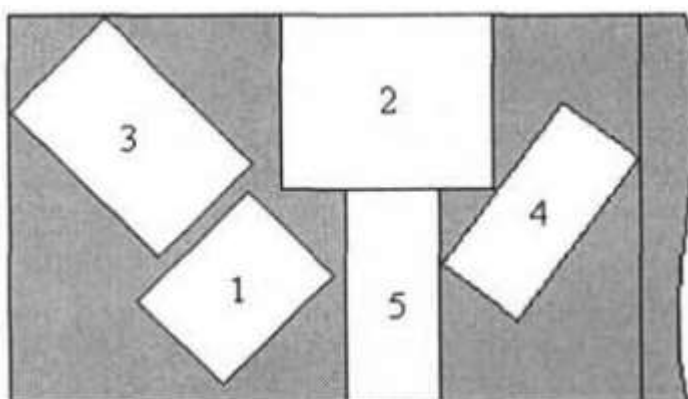


Рисунок 2.3 – Неприпустимий (неортогональний) розкрій

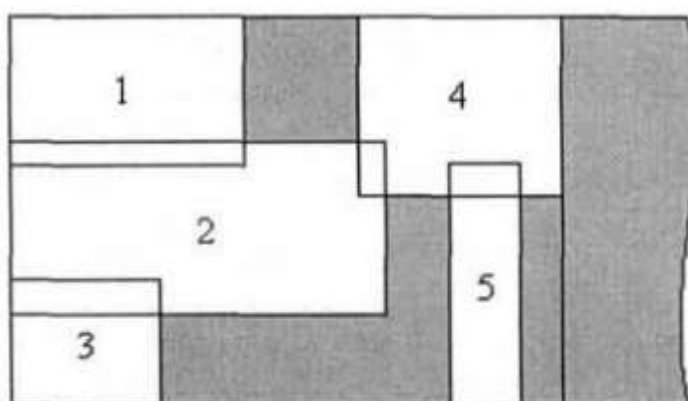


Рисунок 2.4 – Неприпустимий розкрій (з перекриттями)

*Математична модель задачі:* Є прямокутна стрічка заданої ширини  $W$  необмеженої довжини, а також набір прямокутних предметів заданих розмірів  $w_i$  (ширина),  $l_i$  (довжина),  $i = 1, 2, \dots, m$ . Введемо прямокутну систему координат: осі  $OX$  і  $OY$  збігаються зі сторонами стрічки. Положення  $i$ -го прямокутника

назвемо горизонтальним, якщо його сторона  $l_i$  паралельна необмеженій грані стрічки, а друга сторона перпендикулярна їй. Горизонтальне положення кожного прямокутника  $i$  задамо вектором  $(x_i, y_i)$  з мінімальними координатами.

Набір векторів  $(x_i, y_i)$  називається прямокутним розкриєм, якщо виконується обмеження (2.1) та (2.2):

$$(x_i \geq x_j + l_j \vee x_j \geq x_i + l_i) \vee (y_i \geq y_j + w_j \vee y_j \geq y_i + w_i) \quad (2.1)$$

для  $i, j = 1, 2, \dots, m, i \neq j$  та

$$x_i \geq 0 \wedge y_i \geq 0 \wedge y_i + w_i \leq W \quad (2.2)$$

для  $i = 1, 2, \dots, m$ .

Потрібно знайти прямокутним розкриєм ( $PR$ ) з такими параметрами, щоб:

$$L = \max(x_i + l_i) \rightarrow \min \quad (2.3),$$

де  $L$  – довжина зайнятої частини стрічки.

Прямокутним розкриєм, що задовольняє умовам (2.1)-(2.3), називається оптимальним розкриєм і є розв'язком задачі прямокутного розкрою. Відповідна умовам (2.1) та (2.2) карта розміщення прямокутних предметів у стрічці також називається прямокутним розкриєм.

На рис. 2.5 наведено приклад прямокутного розкрою, для якого позначені вихідні дані і значення цільової функції.

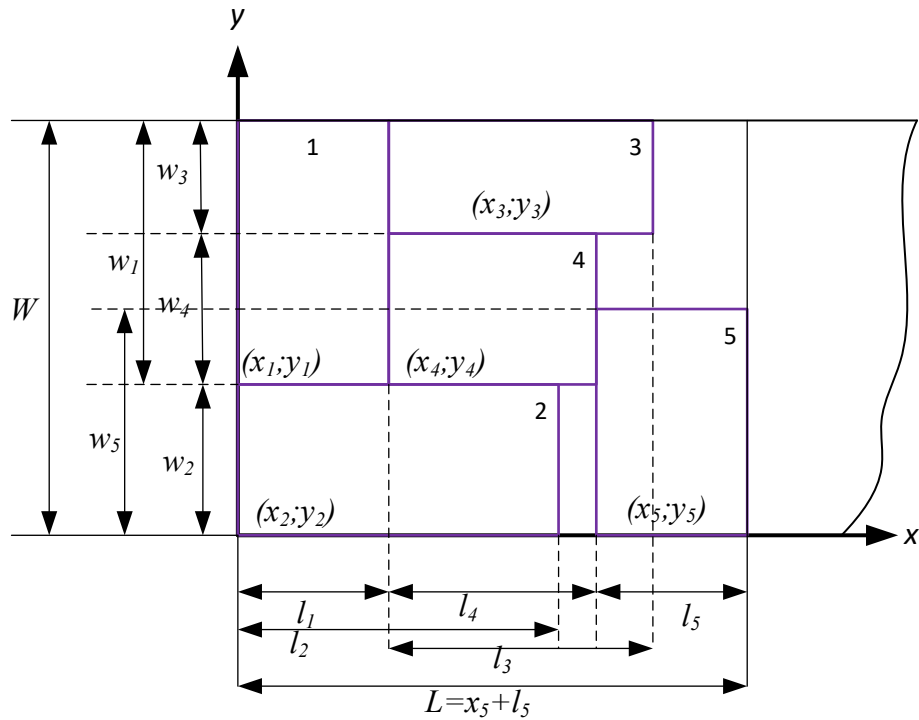


Рисунок 2.5 – Приклад прямокутного розкрою

Оскільки ширина стрічки  $W$  задана, а сумарну площу предметів можна обчислити виходячи з їхніх розмірів, то значення частки корисної площі  $\eta$  і довжина розкрою  $L$  пов'язані наступною рівністю (2.4):

$$\eta = \frac{\sum_{i=1}^m l_i w_i}{WL} \quad (2.4)$$

При побудові і програмній реалізації математичної моделі поставленої задачі як цільову функцію зручніше використовувати  $L$ , а значення  $\eta$ , що є загальноприйнятим показником якості розкрою, обчислювати за потреби за формулою (2.4).

З огляду на NP-складність задачі прямокутного розкрою розробка різних конкуруючих алгоритмів навряд чи може бути припинена. Для розв'язання таких задач відомі трудомісткі точні алгоритми: однопрохідні евристики з апріорними верхніми оцінками отримуваних розв'язків; проблемно-орієнтовані евристичні методи і метаевристики.

Точні методи базуються на алгоритмах типу «гілок і меж». Серед таких фундаментальними є закордонні роботи [85]. До точних алгоритмів можна віднести також «метод зон» Ліповецького [36]. На блоковій структурі розкрою базується метод Картака і Мухачової [39]. Його перевага полягає в тому, що метод є загальним щодо розкрою  $L$ -мірних об'єктів.

Відомі численні однопрохідні евристичні методи [36]. Найбільш популярні «перший відповідний (First Fit, FF)» і «перший відповідний з впорядкуванням (First Fit with Decreasing, FFD)». Вони використовують пріоритетний список (Priority List,  $PL$ ), в якому прямокутники розташовані згідно з обраним правилом. Тоді  $PL$  є шифром упаковки. Остаточний вигляд  $RP$  та її довжина  $L(RP)$  залежать від декодера - алгоритму, що розшифровує  $PL$ . Найпростіший з декодерів – «нижній лівий (Bottom Left,  $BL$ )». Також зустрічається «вдосконалений нижній лівий (Improve Bottom Left,  $IBL$ )» [36]. Застосування  $FF$  зі зміною  $PL$  і повторенням являє метод «послідовного уточнення оцінок (Sequential Value Correction,  $SVC$ )» [41].

Серед простих евристик виділяють також «жадібні» алгоритми. У роботі [37] ідея «жадібного» алгоритму розвинута і застосовується в розгалуженому процесі методу «динамічного перебору». До складних рівневих алгоритмів належить спосіб послідовно-одиночного розміщення (Sequential-Individual Allocation,  $SIA$ ) [67]. Метод полягає в тому, що всі прямокутники розміщуються по одному, причому раніше розміщені вважаються нерухомими, тобто параметри їхнього розміщення мають певні фіксовані значення. Для кожного розміщуваного прямокутного предмета обирається положення з мінімальним значенням функції мети лише за змінними параметрами цього прямокутника.

Закордонними дослідниками запропонований теоретичний графовий метод [36]. Теоретично обґрунтовані перебірні методи «перебудови» (Reconstruction,  $REC$ ) [40]. Вони базуються на блоковій структурі упаковки та її лінійній апроксимації.

Серед різних метаевристик виділяються генетичні алгоритми. Вони використовують різні способи структуризації  $RP$ : пріоритетні списки з

перестановкою прямокутних елементів [36] і блок-структури з перестановкою елементів всередині блоків [40].

Генетичні алгоритми для розв'язання задачі прямокутної упаковки інтерпретуються як еволюційний процес задля відшукування глобального мінімуму. Як класичний генетичний алгоритм розглядається алгоритм з процедурою декодера «вдосконалений нижній лівий», *IBL*. Він володіє багатьма недоліками, серед них – низька середня щільність упаковки.

Істотне підвищення точності розв'язку отримано в методології з [30], яка зводиться до застосування послідовності простих евристик. Для задачі прямокутного розкрою дослідницький колектив професорки Мухачової Е.А. розробив евристичну процедуру, що називається блоковий декодер.

Блок-структури упаковок. Нехай є деяка прямокутний розкрій  $RP$ . Розіб'ємо його вертикальними лініями на прямокутні блоки однієї й тієї ж ширини  $W$  та різної довжини. При цьому початок першого блоку збігається з початком стрічки, а кінець – з кінцем найкоротшого прямокутника, що входить до блоку. Кожен наступний блок  $j = 2, \dots$  починається, як тільки закінчується попередній.

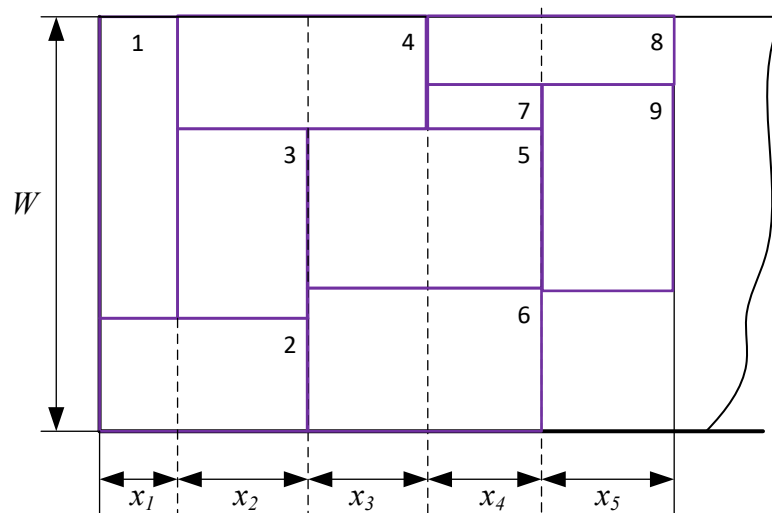


Рисунок 2.6 – Вертикальна блок-структура

На рис. 2.6 наведено безвідхідний розкрій дев'яти різних прямокутників у стрічки ширини  $W$ . Штриховою лінією позначено розбиття упаковки на п'ять

різних блоків. Позначимо довжину кожного  $j$ -го блоку  $x_j$ . Прямокутна упаковка може бути подана у вигляді сукупності кортежів

$$(1_j, 2_j, \dots) x_j \quad j = 1, 2, \dots, \quad (2.5)$$

де  $x_j$  – довжина відповідного кортежа.

Позначимо  $i_k$  – номер прямокутника, який закінчується в  $k$ -му блоці, а  $a_{i_k j}$  – ознака перетину прямокутника  $i_k$  та блоку  $j$

$$a_{i_k j} = \begin{cases} 1, & \text{якщо прямокутник } i_k \text{ та блок } j \text{ перетинаються} \\ 0 & \text{– у протележному випадку} \end{cases} \quad (2.6)$$

Тоді для обчислення довжини  $x_j$  блоку  $k$  використовуються такі рекурентні співвідношення (2.7):

$$x_1 = l_{i_1}; \quad x_k = (l_{i_k} - \sum_{j=1}^{k-1} a_{i_k j} x_j), \quad k = 2, 3, \dots \quad (2.7)$$

Наведеному на рис. 2.6 розкрію та його вертикальній блок-структурі відповідає такий список кортежів (2.8):

$$\begin{aligned} (1; 2) x_1 &= l_1, \\ (4; 3; 2) x_2 &= l_2 - x_1, \\ (4; 5; 6) x_3 &= l_4 - x_2, \\ (8; 7; 5; 6) x_4 &= l_7, \\ (8; 9) x_5 &= l_9, \end{aligned} \quad (2.8)$$

де довжина розкрою  $L = \sum_{j=1}^5 x_j$ .

Аналогічно, провівши лінії, паралельні довжині смуги, отримаємо горизонтальний розподіл  $RP$  на блоки однакової довжини  $L$ . Ширина  $y_j$  кожного

блоку  $j$  визначається через ширину прямокутника, що закінчується в  $j$ -му блоці, за рекурентними співвідношеннями, аналогічними (2.7). Прямокутна упаковка буде подана таким списком кортежів:

$$(1_j, 2_j, \dots) n_j \quad j = 1, 2, \dots, \quad (2.9)$$

Горизонтальна блок-структура безвідхідної упаковки зображена на рис. 2.7. Їй відповідає такий список кортежів:

$$\begin{aligned} (2; 6) n_1 &= w_2, \\ (1; 3; 6) n_2 &= w_6 - n_1, \\ (1; 3; 5; 9) n_3 &= w_5, \\ (1; 4; 7; 9) n_4 &= w_7, \\ (1; 4; 8) n_5 &= w_8, \end{aligned} \quad (2.10)$$

Таким чином, будь-якому  $RP$  відповідають дві блок-структури, вертикальна та горизонтальна, подані списками кортежів (2.8) і (2.10).

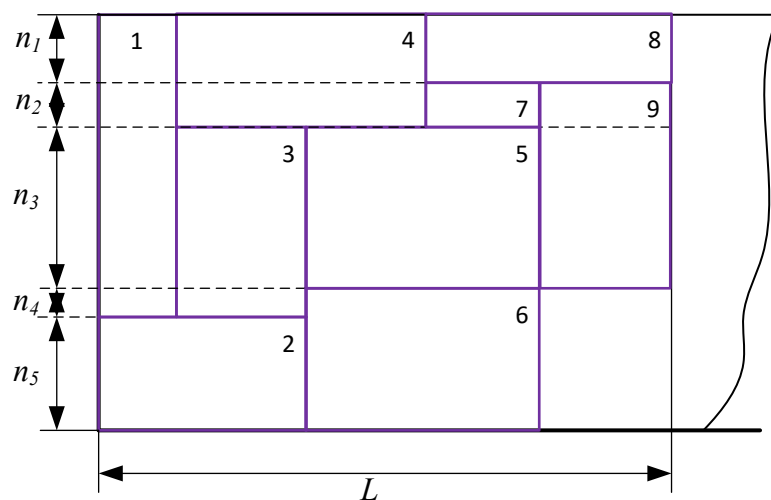


Рисунок 2.7 – Горизонтальна блок-структура

Якщо задана деяка блок-структура, то вона не обов'язково являє допустимий розкрій. Разом з тим, за будь-якою блок-структурою неважко сформуванати відповідний пріоритетний список  $PL$  прямокутників. Для цього використовується список кортежів (2.8), початковий список  $PL1$  прямокутників та формований список  $PL$ .

Спочатку в  $PL$  записують номери всіх прямокутників, що входять до першого кортежа списку (2.8), і видаляють ці номери з  $PLL$ . Потім при перегляді  $j$ -го кортежа записують лише номери прямокутників, що належать цьому кортежу та  $PLL$  викреслюючи їх з  $PLL$ . Так триває допоки всі блоки не переглянуто. Так, для наведеного на рис. 2.6, 2.7 прикладу:  $P1 = \{1,2,4,3,5,6,8,7,9\}$ .

Блоковий декодер. Декодер застосовується на етапі розшифрування  $PL$ , отриманого на основі блок-структури упаковки. Разом з тим блоковий декодер орієнтований на формування блокової структури  $RP$ . Як початковий блок приймається смуга нескінченної довжини. У міру упаковки прямокутників згідно з  $PL$  формують і модифікують блоки. У блок-структурі зберігаються фрагменти прямокутників.

Для додавання в блок-структуру чергового прямокутника необхідно також знати розміри порожніх ділянок, що залишаються в блоках. Для цього запропоновано зберігати ширину такої ділянки в самій блок-структурі, але з деякою позначкою, щоб відрізнити від номерів прямокутників. Довжина порожньої ділянки дорівнює довжині блоку, в якому вона перебуває.

Якщо відомо, що  $PL$  – пріоритетний список;  $W$  – ширина стрічки;  $(w_i, l_i) i = 1, 2, \dots, m$  – розміри прямокутників, тоді блоковий декодер може мати структуру, описану на рис. 2.8. Під модифікацією блок-структури розуміються коректна (відповідно до розмірів прямокутників, що розміщуються) зміна довжини блоків, ширини порожніх ділянок, а також додавання нових блоків.



|  |  |   |                           |
|--|--|---|---------------------------|
| <i>Ініціалізація блок-структури: створюється порожній фіктивний блок нескінченної довжини та і ширини <math>W</math></i>   |  |   |                           |
| <i>Для <math>i = 1, 2, \dots, m</math></i>   |  |   |                           |
| <i><math>k=1</math></i>  |  |   |                           |
| <i>Поки не розміщено прямокутник <math>i</math> [PL]</i>   |  |   |                           |
| <i>Пошук у блок - структурі блоку, що містить порожню ділянку шириною <math>\geq</math> ширини <math>w_i</math> прямокутника <math>i</math> [PL]. Пошук починається з <math>k</math>-го блоку (такий блок завжди знайдеться, оскільки є порожній фіктивний блок, з якого можна створити новий)</i> |  |   |                           |
| <i>Довжина <math>l</math> прямокутника <math>i</math> [PL] <math>\leq</math> Довжина знайденого блоку</i>  |  |   |                           |
| <i>Так</i>   |  | <i>Ні</i>   |                           |
| <i>Занесення прямокутника в блок-структуру, модифікація блок - структури</i>   |  | <i>Допустимо розміщення прямокутника <math>i</math> [PL] за довжиною у наступних блоках</i> |                           |
|  |  | <i>Так</i>  | <i>Ні</i>                 |
|  |  | <i>Занесення прямокутника в блок-структуру, модифікація блок - структури</i>                | <i><math>k=k+1</math></i> |
| <i>Видалення фіктивного блоку, розрахунок довжини отриманого розкрою</i>   |  |   |                           |

Рисунок 2.8 – Структура модель блокового декодера

## 2.2 Розв'язок задачі круглого розкрою

Під задачею круглого розкрою розумітимемо задачу, що відрізняється від задачі прямокутного розкрою лише формою предметів. Якщо вище розглядалася задача розкрою напівнескінченної стрічки на прямокутні предмети різних розмірів, то тут мова піде про розкрій напівнескінченної стрічки на круглі предмети різних радіусів.

Слід зауважити, що прямокутний розкрій завжди був привабливішим для дослідників, оскільки за певної різноманітності предметів можливо досягти досить високих показників якості розкрою. Крім того, для прямокутного розкрою існують такі набори предметів, які упаковуються в стрічку єдиним великим прямокутником без втрат корисної площі. Очевидно, що для задачі круглого розкрою таких постановок не існує.

*Змістовна постановка задачі:* Є прямокутна стрічка заданої ширини та необмеженої довжини, а також набір круглих предметів заданих радіусів.

Потрібно визначити параметри оптимального розташування предметів у стрічці за таких умов (умов неперетинання):

- упаковані предмети не перетинаються один з одним;
- упаковані предмети не перетинаються зі сторонами стрічки (приклад недопустимого розташування на рис. 2.9, допустимих розташувань - на рис. 2.10 і рис. 2.11).

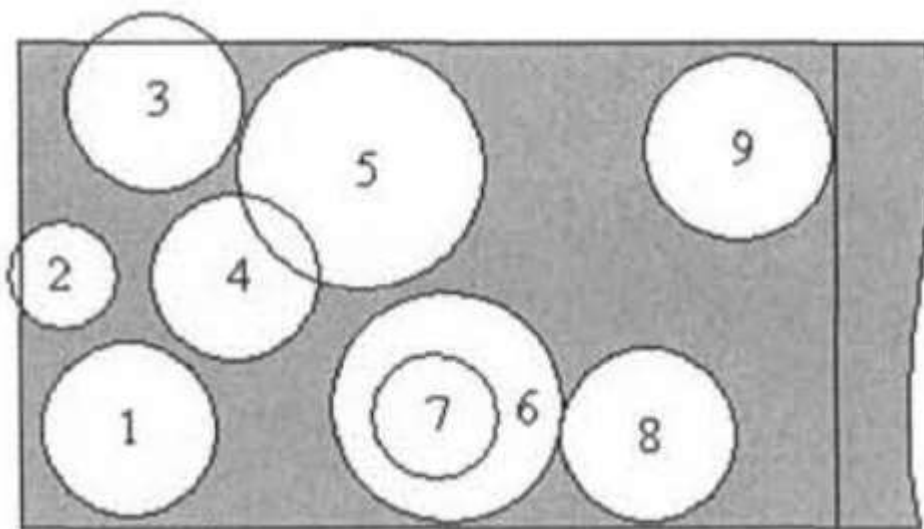


Рисунок 2.9 – Неприпустимий розкрій

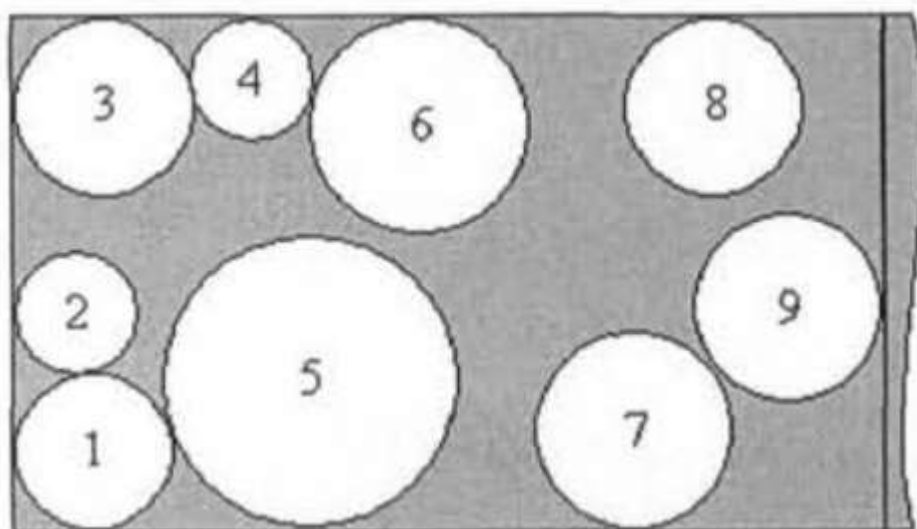


Рисунок 2.10 – Допустимий (нешільний) розкрій

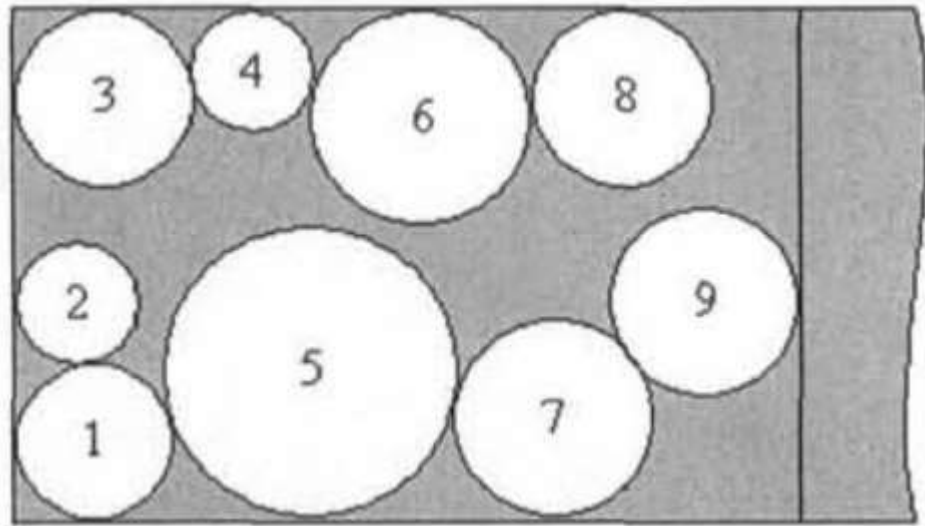


Рисунок 2.11 – Допустимий (щільний) розкрій

*Математична постановка задачі.* Є прямокутна стрічка заданої ширини  $W$  необмеженої довжини, а також набір круглих предметів заданих радіусів  $r_i$ ,  $i = 1, 2, \dots, m$ . Введемо прямокутну систему координат: осі  $OX$  і  $OY$  збігаються зі сторонами стрічки. Положення кожного кола  $i$  задамо вектором  $(x_i, y_i)$  з координатами його центру. Набір векторів  $(x_i, y_i)$  називається круговим розкритом ( $CR$ ), якщо (2.11):

$$(x_i - x_j)^2 + (y_i - y_j)^2 \geq (r_i - r_j)^2 \quad (2.11)$$

для  $i, j = 1, 2, \dots, m, i \neq j$  (умова неперетинання кіл одне з одним, у тому числі умова невкладеності кіл одне в одне) та

$$x_i - r_i \geq 0 \wedge y_i - r_i \geq 0 \wedge y_i + r_i \leq W \quad (2.12)$$

для  $i = 1, 2, \dots, m$ . (Умова неперетинну з краями стрічки)

Потрібно знайти п розкритом ( $CR$ ) з такими параметрами, щоб:

$$L = \max(x_i + r_i) \rightarrow \min \quad (2.13),$$

де  $L$  – довжина зайнятої частини стрічки.

$CP$ , що задовольняє умовам (2.11)-(2.13), називається оптимальним розкритом та є розв’язком задачі круглого розкрою напівнескінченної стрічки.

Відповідна умовам (2.11)-(2.13) карта розміщення круглих предметів у стрічці також називається круговим розкритом. На рис. 2.12 наведено приклад кругової упаковки, для якої позначені вихідні дані і значення цільової функції.

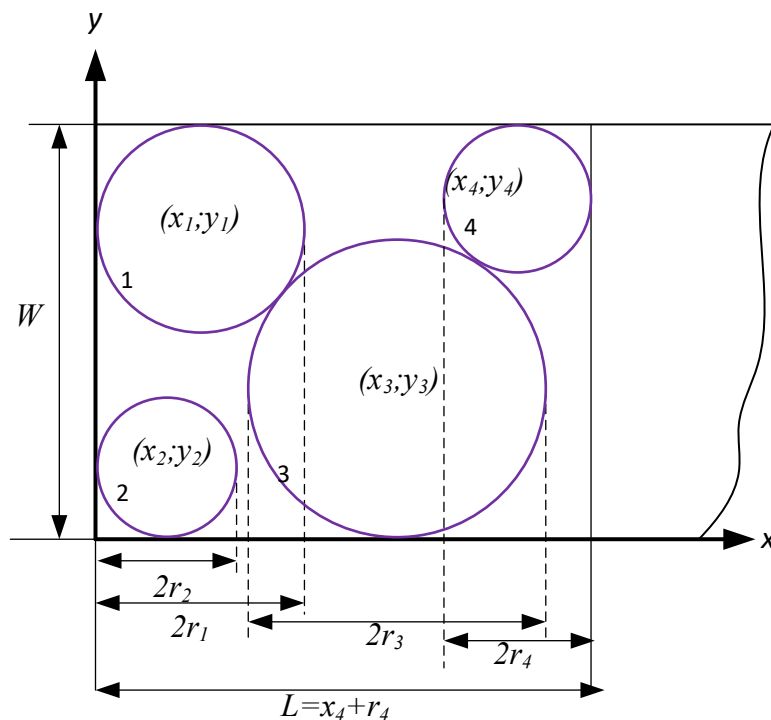


Рисунок 2.12 – Приклад кругового розкрою

Методи розв’язання задачі круглого розкрою. Задача круглого розкрою досить часто зустрічається у промисловості, щоб було доцільно розробляти методи оптимізації такого розкрою. Наприклад, штампування круглих заготовок у металургії. Листове штампування - один з найбільш прогресивних методів обробки металів тиском. Один з технологічних процесів у листоштампувальному виробництві - розкрій круглих листових заготовок. Тут вартість матеріалу становить 60-80% від вартості виробів. У зв’язку з цим велике значення має економія металу. Собівартість деталей зменшується на 0,4-0,5% на кожний

відсоток зменшення відходів. Витрата матеріалу багато в чому визначається раціональним розташуванням предметів на листі. Ефективність розкрою на однакові круглі заготовки залежить від вибору розмірів вихідного листа (або стрічки), в разі класичної схеми розкрою - розташування смуг на листі (повздовжнього, поперечного або косого), розташування деталей у смузі чи стрічці та призначення перемичок оптимальної величини [2]. Найпростіша задача ставиться так: потрібно оптимальним чином розмістити круглі предмети однакового радіуса на листі. У металургійному виробництві при штампуванні круглих деталей в такому разі лист металу розмічається на повздовжні або поперечні смуги однакової ширини (рис. 2.13). А безпосередньо розташування предметів оптимізується на такій смузі. Предмети на смузі можна розташовувати у паралельному (рис. 2.14) або шаховому (рис. 2.15) порядку. При розрахунку розкладки слід враховувати, наприклад, такі технологічні параметри як діаметр деталі що штампується, величину перемички від краю смуги, величину перемички між деталями, односторонній допуск на ширину смуги, гарантований зазор між напрямними у штампі та найбільшої можливої ширини смуги. Для будь-якого знайденого розміщення розраховують коефіцієнт використання матеріалу за даних технологічних параметрів. Він дорівнює відношенню загальної площі "укладених" предметів до площі поверхні листа і визначає якість розміщення [2]. Для штампування деталей великого (але одного і того ж) діаметра з попередньо нарізаних смуг, коли деталь укладається поперек листа всього кілька разів, раціональним може виявитися косий розкрій (рис. 2.16). При побудові косого розкрою нахил смуги визначається з умови укладання найбільшого цілого числа круглих деталей у смузі. Коефіцієнт використання матеріалу визначається за тою ж формулою. В оптимізаційній задачі він є цільовою функцією, яку потрібно максимізувати [2].

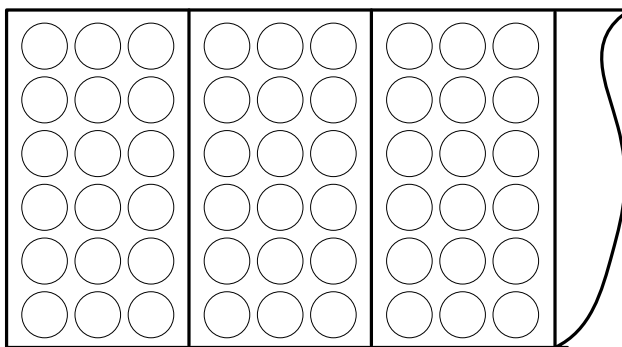


Рисунок 2.13 – Технологічна схема поперечного розкрою

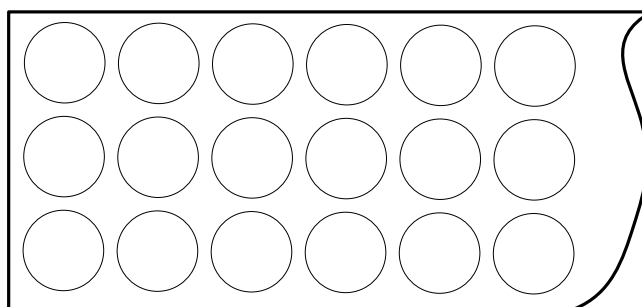


Рисунок 2.14 – Технологічна схема паралельного розкрою стрічки

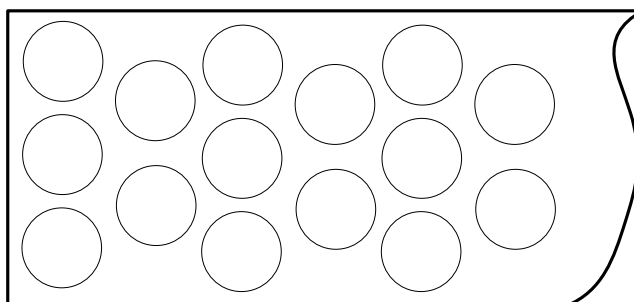


Рисунок 2.15 – Технологічна схема шахового розкрою стрічки

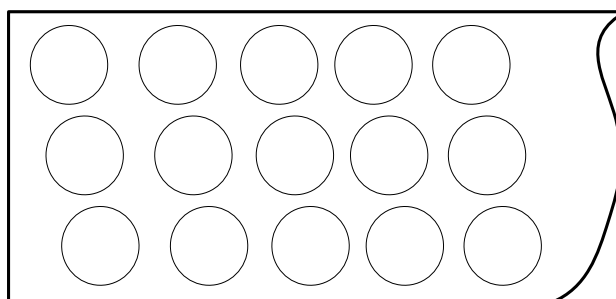


Рисунок 2.16 – Технологічна схема косого розкрою стрічки

Розглянуті підходи не застосовні для вирішення задачі з предметами різного діаметра. Для вирішення цієї проблеми слід розробляти спеціальні підходи. Водночас, для кіл однакового діаметра оптимальність шахового розташування на (необмеженій) площині суворо доведена в геометрії [18]. Таким чином, для ефективного вирішення задачі круглого розкрою потрібно розробляти спеціальні, що враховують різноманітність геометрії предметів методи. Оскільки задача NP-складна, то доцільно використання евристик, що виграють у швидкості роботи за рахунок прийняттого програшу в якості рішення, таких як генетичний алгоритм. Природний розвиток технологій вирізання такий, що до недавнього часу на виробництві відмовлялися від ідеї викроювати з одного листа кола 48 різних розмірів. Однак сьогодні лазерне різання витісняє такий прийом як штампування (вирубка) і постановка подібної задачі стала актуальною. Проте, в даний час знайти публікації у вітчизняних джерелах на цю тему не видається можливим, а ознайомлення із зарубіжними літературними джерелами, анотації яких здаються відповідними проблемі, вимагає фінансової підтримки [47, 49, 50].

Структура краю круглої упаковки. Назвемо проміжним розміщенням часткову упаковку, отриману в результаті декодування частини пріоритетного списку, тобто в результаті укладання на смугу кількох предметів зі списку поспіль у порядку слідування. Назвемо краєм круглої упаковки послідовність предметів з часткового розміщення з наступними властивостями:

- кожен предмет краю торкається двох інших предметів, або одного іншого предмета і одного чи декількох країв смуги, або декількох країв смуги;
- якщо послідовно від одного напівнескінченного краю смуги до іншого в порядку торкання предметів краю з'єднати їх центри векторами, то всі інші предмети упаковки виявляться по один і той же бік цього ланцюжка [47, 49, 50].

Для кращого розуміння поняття краю можна провести наступну «фізичну» аналогію. Нехай смуга - це двовимірний (плоский) келих, торець смуги - це дно келиха. Тоді якщо в цю уявну ємність зверху налити води, то всі кола, які виявляться хоча б частково змоченими і становлять край. Зрозуміло, подібна аналогія наводиться за припущення, що вода не проникає між фігурами в точках

торкання. Межі смуги включаються в край упаковки додаванням ще двох векторів, що з'єднують точки торкання відповідних кіл з межами смуги та центри цих кіл.

Приклад краю круглої упаковки проілюстровано на рис. 2.17. Поняття краю упаковки необхідне для розрахунку позиції наступних предметів зі списку.

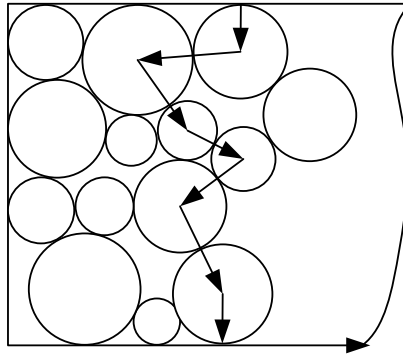


Рисунок 2.17 – Приклад краю кругового розкрою

Позначимо краї смуги наступними символами:  $W$  – торець, лівий край смуги,  $L_t$  – верхній напівнескінченний край смуги,  $L_b$  – нижній напівнескінченний край смуги. Тоді межу будь-якої упаковки можна задати цими символами та послідовністю номерів кіл, що становлять цю межу. Наприклад, на рис. 2.18 межа упаковки задається наступною послідовністю:  $\{W, 9, 2, 5, 12, 8, 11, 4, L_b\}$ .

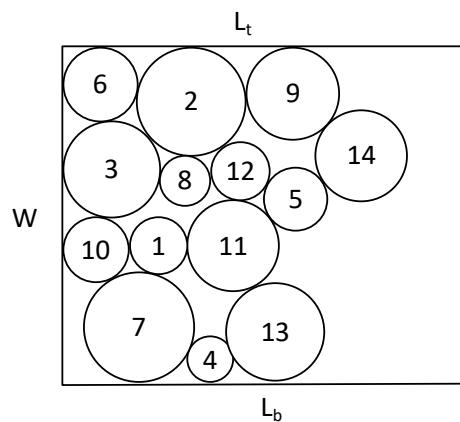


Рисунок 2.18 – Приклад позначень для запису краю розкрою



Загалом, вся упаковка, представлена на рис. 2.17, відповідає такому пріоритетному списку:  $\{6,3,10,7,2,13,1,12,8,11,5,9,4\}$ . Оскільки крайовий декодер викладає кола на смугу по одному, то слід окремо визначити, як задається межа упаковки на початку роботи декодера, наприклад, коли кіл ще взагалі немає або дуже мало. Ініціалізується межа упаковки послідовністю такого виду:  $\{L_t, W, L_b\}$ . У випадку малої кількості кіл, коли межа упаковки ще не закрила торець смуги, її задають, використовуючи позначення цього краю смуги, наприклад, як на рис. 2.19:  $\{L_t, 6,3,10,7, W, L_b\}$  [47,49,50].

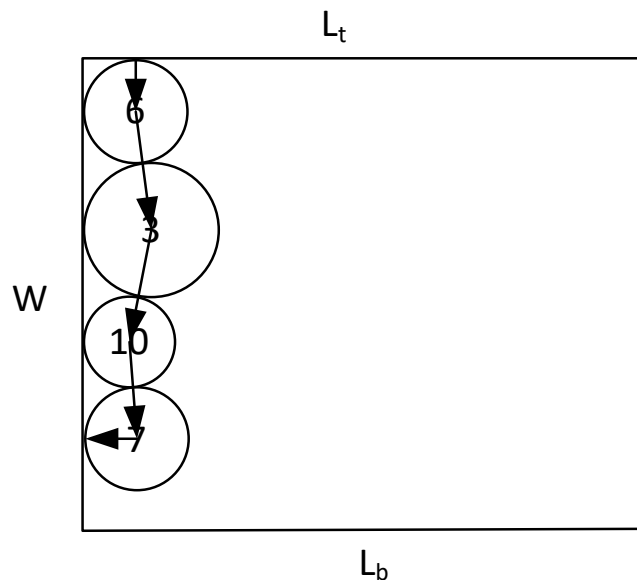


Рисунок 2.19 – Приклад послідовності векторів

Крайовий декодер. Отже, порядок слідування кіл заданий пріоритетним списком. Крайовий декодер бере з нього номери  $i$  по черзі викладає відповідні кола на смугу. Перед тим, як розмістити кожне наступне коло, декодер розглядає всі можливі позиції кола, в які його можна помістити, притиснувши «ззовні» до краю часткового розміщення. На рис. 2.20 наведено приклад упаковки, в якому чергове коло можливо розмістити 5-ма способами.

Для остаточного розміщення кола декодер вибирає ту з розглянутих позицій, яка мінімізує загальну довжину упаковки (рис. 2.16) [47, 49, 50].

Якщо межа упаковки задається описаним вище способом, то пошук оптимального розміщення кожного наступного кола здійснюється за схемою, проілюстрованою на рис. 2.19 [47, 49, 50]. Слід зазначити, що якщо блоковий декодер вузько спеціалізований і придатний лише для укладання прямокутних предметів, то крайовий декодер більш універсальний як ідея. У ньому достатньо змінити кілька геометричних формул, що відповідають за торкання предметів, і можна укладати на смугу одночасно і прямокутні, і круглі предмети. Причому обчислювальні витрати (час роботи, наприклад) при цьому не збільшаться.

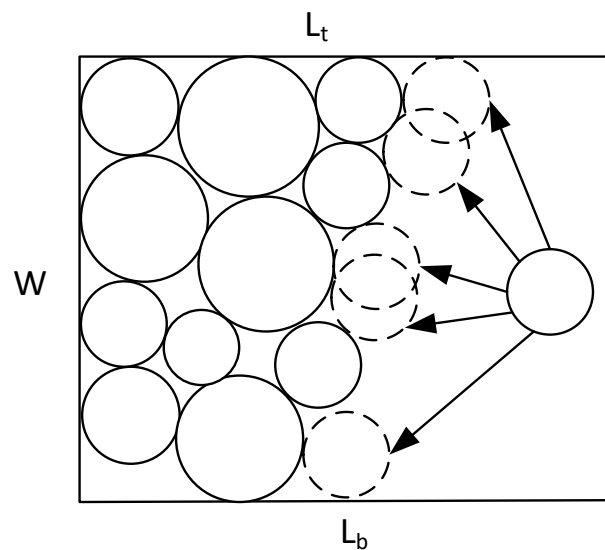


Рисунок 2.20 – Приклад додавання нового кола

Кожна можлива позиція кола визначається тими двома елементами з межі упаковки, яких торкається розглянуте коло в цій позиції. Теоретично можливі такі позиції, де торкань буде більше двох. У таких випадках позиція визначається першим та останнім елементом зі списку дотичних.

Очевидним недоліком крайового декодера є відсутність будь-якої спроби враховувати утворені раніше пустоти для розміщення нових відносно невеликих кіл. Кожне нове коло розміщується суворо зовні вже наявної упаковки. У блоковому декодері для упаковки прямокутників, розглянутому в попередньому розділі, такого недоліку немає.

|   |  |
|---|--|
| <b>Ініціалізувати край упаковки <math>\{L, W, L_b\}</math> та загальну довжину упаковки (<math>CL = 0</math>)</b>                                   |  |
| <b>Для кожного кола з <math>PL\ i = 1+t</math></b>  |  |
| <b>Ініціалізувати перший елемент торкання (<math>p_1</math>) як перший із краю упаковки та другий елемент торкання (<math>p_2</math>) як другий</b> |  |
| <b><math>i</math>-е коло, торкаючись <math>p_1</math> п <math>p_2</math>, не порушує умов (2.8) і (2.9)</b>   |  |
| <b><u>Ні</u></b><br>$p_2 =$ наступний елемент з межі упаковки   | <b><u>Так</u></b><br>Зберегти у списку позиції координати центру $i$ -ого кола, $p_1$ і $p_2$ , відповідне значення $CL$<br>$p_1 =$ наступний елемент з краю упаковки;<br>$p_2 =$ наступний елемент із краю упаковки |
| <b>Упорядкувати список позицій щодо зростання <math>CL</math></b>   |  |
| <b>Вибрати позицію, що відповідає мінімуму <math>CL</math></b>  |  |
| <b>Перетворити край упаковки відповідним чином, додавши новий елемент і видаливши закриті ним, якщо такі є</b>                                      |  |
| <b>Як результат роботи підрахувати підсумкове значення <math>CL</math></b>  |  |

Рисунок 2.21 – Структура модель крайового декодера

Для поліпшення роботи крайового декодера і як спроба частково компенсувати вищеописаний недолік, була розроблена наступна процедура. З точки зору планіметрії, в задачі пошуку координат центру кола, що торкається одночасно двох інших, існує два розв'язки (рис. 2.21). Те саме стосується і задачі з торканням не трьох кіл, а двох кіл і прямої (рис. 2.22).

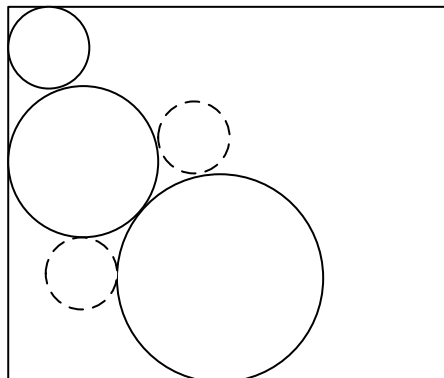


Рисунок 2.22 – Торкання з двома колами

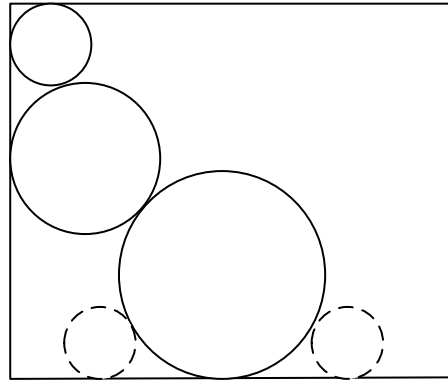


Рисунок 2.23 – Торкання з прямою і колом

При кожній перевірці коректності пробної позиції кола пропонується перевіряти і парну до неї позицію, що розміщує коло не ззовні межі упаковки, а всередині. Оскільки з майже немінною межею «попрацюють» кілька кіл, перш ніж вони її повністю закриють або видозмінять, то за певного різноманіття упакуваних форм, цей підхід призводить до поліпшення якості упаковки. Крім того, додавання описаної процедури нікчемно збільшує обчислювальні витрати при пошуку рішення, що є ще одним аргументом на користь її застосування [47, 49, 50].

Безпосередньо розв'язання поставлених геометричних задач здійснюється наступним чином.

*Задача 1 (про торкання кола з двома іншими)* [47, 49, 50]. Є два кола, для яких відомі координати центрів  $(x_1, y_1)$  та  $(x_2, y_2)$  і радіуси  $r_1, r_2$ . Потрібно знайти координати центру третього кола  $(x_0, y_0)$ , відомого радіуса  $r_0$ , що торкається перших двох. Очевидно, що обидва розв'язки задачі є точками перетину двох кіл: кола з центром у точці  $(x_1, y_1)$  та радіусом  $r_1 + r_0$  кола з центром у точці  $(x_2, y_2)$  та радіусом  $r_2 + r_0$ . З алгебраїчної точки зору, значення  $(x_0, y_0)$  знаходяться як розв'язок системи з двох рівнянь з двома невідомими:

$$\begin{cases} (x_0 - x_1)^2 + (y_0 - y_1)^2 = (r_0 + r_1)^2 \\ (x_0 - x_2)^2 + (y_0 - y_2)^2 = (r_0 + r_2)^2 \end{cases} \quad (2.14),$$

*Задача 2 (про торкання кола з колом і прямою) [47,49,50].* Є коло з радіусом  $r_1$  та координатами центру  $(x_1, y_1)$  і пряма, задана рівнянням виду  $x = 0$ , або  $y = 0$ , або  $y = W$  (залежно від конкретних умов задача ставиться щодо різних меж смуги). Потрібно знайти координати центру другого кола  $(x_0, y_0)$ , відомого радіуса  $r_0$ , що торкається заданого кола і прямої. Очевидно, що обидва розв'язки задачі є точками перетину кола з центром у точці  $(x_1, y_1)$  та радіусом  $r_1 + r_0$  і прямої, заданої рівнянням виду  $x = r_0$ , або  $x = y_0$ , або  $y = w - r_0$  (відповідно). З алгебраїчної точки зору значення  $(x_0, y_0)$  знаходяться як розв'язок системи з двох рівнянь з двома невідомими:

$$\begin{cases} (x_0 - x_1)^2 + (y_0 - y_1)^2 = (r_0 + r_1)^2 \\ x = r_0, \text{ або } x = y_0, \text{ або } y = w - r_0 \end{cases} \quad (2.15),$$

Після розв'язання всіх поставлених задач залишається лише одне питання: як в загальному вигляді з двох розв'язків задачі 1 (за умови допустимості обох) розрізнити «внутрішнє» та «зовнішнє» щодо межі упаковки розв'язання. З точки зору людини - це очевидно, але з точки зору комп'ютерної програми критерій вибору має бути формалізований і описаний в термінах, наприклад, мови програмування. Для вирішення цієї проблеми (так званої задачі 3 [47,49,50]) використана векторна алгебра.

Нехай вектор  $\vec{a}(x; y)$  з'єднує центри кіл, дані за умовою задачі 1. Тоді його координати можна виразити через відомі величини за правилом віднімання векторів:

$$\vec{a}(x; y) = \vec{c}(x_2; y_2) - \vec{b}(x_1; y_1) = \vec{a}(x_2 - x_1; y_2 - y_1) \quad (2.16).$$

Нехай вектор  $\vec{a}(y; -x)$  – перпендикуляр до вектора  $\vec{a}(x; y)$ . Розглянемо скалярний добуток  $\vec{a}d$ , де  $d(x_0 - x_1; y_0 - y_1)$  – це вектор, що має спільний початок з вектором  $\vec{a}(y; -x)$  і кінець в центрі розглянутого кола  $(x_0, y_0)$ . Тоді:

$$\vec{a}d = y(x_0 - x_1) - x(y_0 - y_1) \quad (2.17)$$

Скалярний добуток визначається рівністю:

$$\vec{a}d = |\vec{a}||d|\cos\varphi \quad (2.19)$$

Таким чином, якщо  $\vec{a}d > 0$ , то розглянуте коло з центром в точці  $(x_0, y_0)$  знаходиться ліворуч від вектора  $\vec{a}(x; y)$ , якщо дивитися уздовж напрямку вектора. Відповідно, якщо  $\vec{a}d < 0$ , то розглянуте коло знаходиться праворуч від вектора  $\vec{a}(x; y)$ , якщо дивитися уздовж напрямку вектора.

2.3 Аналіз застосування генетичних алгоритмів розв'язку задачі розкрою матеріалів.

На даний час широке практичне застосування отримали такі методи оптимізації, як генетичні алгоритми. Проте, генетичні алгоритми входять до ширшого класу алгоритмів - еволюційні обчислення (алгоритми), які являють собою модель біологічного процесу еволюції: у множині об'єктів (популяції) йде боротьба за існування. З плином часу об'єкти вдосконалюються і виживають найсильніші.

Об'єктом може бути структура даних, автомат, програма. Якість об'єкта оцінює єдина для всієї популяції функція придатності. Цю функцію вибирають так, щоб вищу придатність мали об'єкти, що більш відповідають меті еволюції. Часто придатність визначається очевидним способом.

Наприклад, це може бути прибуток, рахунок очок у грі, кількість вчинених помилок, довжина маршруту, відхилення контрольованого параметра від заданого значення та ін.

Нова популяція може створюватися із старої як із збереженням числа її членів, так і зі зростанням популяції до заданої межі.

Вибірка з популяції, оператори обробки об'єктів мають елемент випадковості. За раціонального вибору правил модифікації популяція поступово заповнюється об'єктами з високою придатністю [66].

Широкому розповсюдженню еволюційних обчислень сприяють принаймні дві причини. По-перше, природний відбір усуває необхідність заздалегідь враховувати всі особливості задачі, що розв'язується, та дії, які має виконувати програма при зіткненні з цими особливостями.

Можна створювати програми, що успішно розв'язують поставлену задачу, структуру яких програміст не знає [66, 58].

Є багато прикладів розв'язання за допомогою еволюційних обчислень досить складних практичних задач. Серед них такі відомі як задача комівояжера для великої кількості міст, різні задачі розподілу ресурсів.

Є приклади розв'язання задач у реальному часі, серед них - керування об'єктом, що рухається.

Друга причина – високий паралелізм алгоритмів: кожен член популяції може оброблятися незалежно. Тому такі алгоритми добрі для виконання на мультипроцесорних обчислювальних системах, що дозволяє істотно прискорити їх виконання.

Багато розв'язків задач, отриманих за допомогою еволюційних обчислень, виконані на мультипроцесорних машинах [66].

Властивістю еволюційних алгоритмів є поступове покращення розв'язку від популяції до популяції, і якщо можна задовольнятися грубим розв'язком, то його можна отримати за менший час. Це важлива якість для систем реального часу.

Типові представники еволюційних обчислень – генетичні алгоритми. Поведінку генетичних алгоритмів можна наочно продемонструвати на прикладах функцій від двох змінних. На двовимірній області можна легко простежити динаміку збіжності особин популяції. На рис. 2.24 точками зображено популяцію особин деяких тестових задач на 1-му, 5-му і 10-му поколіннях [58].

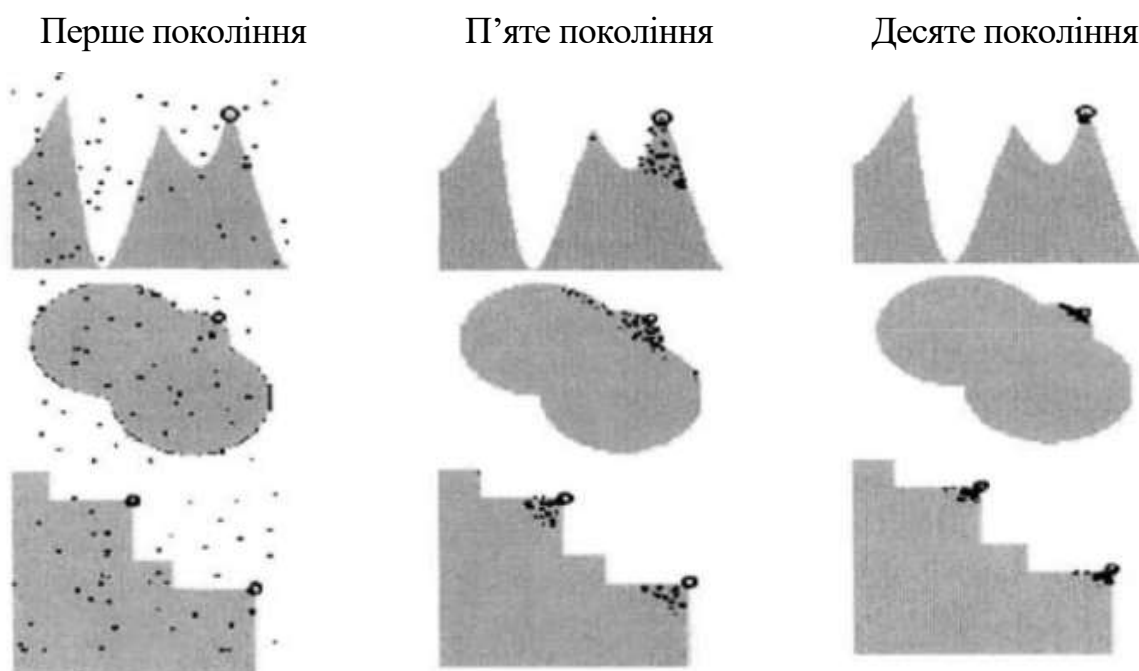


Рисунок 2.24 – Поведінка популяції з прикладу тестових завдань

Узгодженість і ефективність роботи елементів біологічних організмів вже давно наштовхує людей на думку — чи можна використовувати принципи біологічної еволюції для оптимізації практично важливих для людини систем? У різних модифікаціях подібні ідеї виникали в низки авторів. У 1966 р. Л. Фогель, А. Овен, М. Волш написали книгу «Штучний інтелект та еволюційне моделювання», в якій запропонували схему еволюції логічних автоматів, що розв'язують задачі прогнозу [74]. У 1975 р. вийшла фундаментальна книга Дж. Холланда «Адаптація в природних і штучних системах», в якій був запропонований генетичний алгоритм, досліджений надалі учнями і колегами Холланда в Мічиганському університеті [83]. Приблизно в цей же час група німецьких вчених (І. Рехенберг, Г.-П. Швехель та ін.) почала розробку так званої еволюційної стратегії. Ці роботи заклали основи еволюційного моделювання або еволюційних алгоритмів [54]. У загальному вигляді еволюційний алгоритм — це евристичний метод стохастичної оптимізації, що базується на еволюції популяції особин. Кожна особина характеризується придатністю — багатовимірною функцією її генів. Задача оптимізації полягає в максимізації функції придатності. В процесі еволюції в результаті відбору, рекомбінацій і мутацій геномів особин



відбувається пошук особин з високими придатностями. Основні еволюційні алгоритми: генетичний алгоритм, призначений для оптимізації функцій дискретних змінних, що акцентує увагу на рекомбінаціях геномів; еволюційне програмування, орієнтоване на оптимізацію безперервних функцій без використання рекомбінацій; еволюційна стратегія, орієнтована на оптимізацію безперервних функцій із використанням рекомбінацій; генетичне програмування, що використовує еволюційний метод для оптимізації комп'ютерних програм [54].

Порівняно зі звичайними оптимізаційними методами еволюційні алгоритми мають такі особливості: паралельний пошук, випадкові мутації і рекомбінації вже знайдених добрих рішень. Вони добре підходять як простий евристичний метод оптимізації багатовимірних, погано визначених функцій. Найбільшого поширення набув генетичний алгоритм. На його основі здійснюються: оптимізація профілів балок у будівництві, розподіл інструментів у металообробних цехах, обробка рентгенівських зображень у медицині, оптимізація роботи нафтових трубопроводів і багато іншого. Одна з основних областей застосування - вирішення задач комбінаторної оптимізації [54]. На відміну від звичайних методів оптимізації, генетичний алгоритм не висуває якихось особливих вимог до цільової функції (тобто функції придатності); не потрібно отримання інформації про її градієнт; вона може бути багатомодальною, розривною, «голчастою» тощо [6].

Генетичний алгоритм – це комп'ютерна модель еволюції популяції штучних особин. Кожна особина характеризується своєю хромосомою  $S_k$ , хромосома є геномом особини. Хромосома визначає пристосованість {придатність} особини  $f(S_k)$ ;  $k = 1, \dots, n$ ;  $n$  – чисельність популяції. Хромосома є ланцюжком символів  $S_k = (S_{k1}, S_{k2}, \dots, S_{kN})$   $N$  – довжина ланцюжка. Символи інтерпретуються як гени особини, розташовані в хромосомі  $S_k$ . Задача алгоритму полягає в максимізації функції пристосованості  $f(S_k)$ . Еволюція складається з послідовності поколінь. Для кожного покоління відбираються особини з більшими значеннями пристосованостей. Хромосоми пристосованих особин рекомбінуються і піддаються малим мутаціям. Формально, схема генетичного

алгоритму може бути представлена наступним чином (популяція  $t$ -го покоління позначається як  $\{S_k(t)\}$ ):

Крок 0. Створити випадкову початкову популяцію  $\{S_k(0)\}$ .

Крок 1. Обчислити пристосованість  $f(S_k)$  кожної особини  $S_k$  популяції  $\{S_k(t)\}$ .

Крок 2. Здійснюючи відбір особин  $S_k$  відповідно до їх пристосованостей  $f(S_k)$  та застосовуючи генетичні оператори (рекомбінації і точкові мутації) до відібраних особин, сформуванати популяцію наступного покоління  $\{S_k(t + 1)\}$ .

Крок 3. Повторювати кроки 1, 2 для  $t = 0, 1, 2$  допоки не виконається деяка умова завершення еволюційного пошуку (припиняється зростання максимальної пристосованості в популяції, число поколінь  $t$  досягає заданої межі тощо) [54].

Існує ряд конкретних варіантів генетичного алгоритму, які відрізняються за схемами відбору, рекомбінацій, за формою подання хромосом тощо. Традиційний варіант генетичного алгоритму базується на наступній конкретній схемі:

- ланцюжки символів у хромосомах бінарні (символи  $S_k$  приймають значення 0 або 1), довжина ланцюжків постійна ( $N = const$ );
- метод відбору пропорційно-ймовірнісний (див. нижче),
- рекомбінації проводяться за схемою одноточкового кросинговеру.

Пропорційно-ймовірнісний відбір означає, що на кроці 2 відбір здійснюється з ймовірностями, пропорційними пристосованостям  $f_k$  особин ( $f_k = f(S_k)$ ). Цю схему відбору можна уявити як вибір особини за допомогою рулетки, відносні площі секторів якої дорівнюють  $q_k = f_k / [\sum_l f_l]$  [54]  
 рисунок 2.23.

Можливі й інші методи відбору. Наприклад, відбір може бути ранжированим: всі особини ранжуються за пристосованостями і задана частина (наприклад, краща половина) кращих особин відбирається для формування наступного покоління.

Одноточковий кросинговер організовується за аналогією з біологічною рекомбінацією. А саме, якщо є два батьки  $S_1 = (S_{11}, S_{12}, \dots, S_{1N})$  і  $S_2 = (S_{21}, S_{22}, \dots, S_{2N})$ , то їх нащадки є  $(S_{11}, \dots, S_{1m}, S_{2m+1}, \dots, S_{2N})$  і  $(S_{21}, \dots, S_{2m},$

$S_{1m+1}, \dots, S_{1N}$ ); тобто "голова" і "хвіст" хромосоми нащадка беруться від різних батьків. Точка кросинговеру обирається випадковим чином, у наведеному прикладі вона розташовується між  $m$ -им і  $m+1$ -им генами. Аналогічним чином може бути організований двохточковий і «багатоточковий» кросинговер.

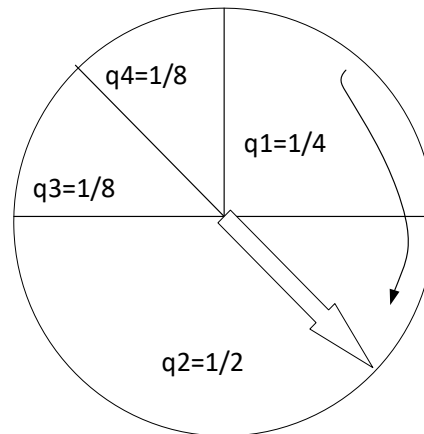


Рисунок 2.25 – Схема відбору, при якій особи вибираються в популяцію нового покоління з ймовірностями  $q_k$  пропорційним їхнім пристосуванням  $f_k$

Тип рекомбінації за схемою кросинговеру часто доповнюється інверсіями, тобто зміною порядку слідування символів в ділянках хромосом; це аргументується як необхідність підібрати суттєві для пристосованості комбінації символів в хромосомі [54].

Деякі схеми генетичних алгоритмів використовують рівномірні рекомбінації. Це означає, що два батьки мають двох нащадків, символи хромосоми одного з нащадків обираються випадково від будь-якого з двох батьків (але із збереженням порядку слідування символів), а другому нащадку дістаються решта символи.

Наприклад, два нащадки батьків  $S_1 = (S_{11}, S_{12}, \dots, S_{1N})$  і  $S_2 = (S_{21}, S_{22}, \dots, S_{2N})$  можуть мати наступні хромосоми:  $(S_{11}, S_{22}, S_{13}, S_{14}, \dots, S_{2N})$  і  $(S_{21}, S_{12}, S_{23}, S_{24}, \dots, S_{1N})$ .

Як вже було зазначено, генетичні алгоритми володіють внутрішнім паралелізмом (implicit parallelism): різні окремі суттєві комбінації генів - їх зазвичай називають «схематами» («schemata») або «схемами» – відшукуються

паралельним чином, одночасно для всіх комбінацій. Можна відзначити, що чим менше комбінація, тим легше вона може бути знайдена [66, 54].

Розглядається найпростіший випадок традиційного варіанту генетичного алгоритму і припускається, що по-перше ланцюжки символів у хромосомах бінарні ( $S_{ki} = 0$  або  $1$ ), по-друге довжина ланцюжків  $N = const$ , по-третє відбір пропорційно-ймовірнісний, по-четверте рекомбінації відсутні, є лише точкові мутації (випадкові рівноймовірні заміни символів), по-п'яте чисельність популяції постійна:  $n = const$ . Тоді справедливі наступні оцінки (2.20), (2.25):

$$T \sim N, \quad (2.20)$$

де  $T$  – число поколінь, необхідних для знаходження оптимуму;

$$n_{zag} \sim N^2, \quad (2.21)$$

де  $n_{zag}$  – загальне число особин, що беруть участь в еволюції [54].

Точність рішення, що отримується генетичним алгоритмом, можна оцінити наступним чином. Точність методу відпалу – приблизно 3-5% [28]. Численні дослідження показали, що рішення, яке отримується за допомогою генетичного алгоритму, принаймні, не гірше [84]. Але генетичний алгоритм автоматично здійснює одночасну (паралельну) оптимізацію за великою кількістю параметрів. Для зміни параметра в тому ж методі відпалу потрібне спеціальне, подекуди трудомістке програмування [45, 46].

Хоча ці оцінки досить грубі, вони важливі з інженерної точки зору - використовуючи їх, розробник конкретного алгоритму може оцінити ту обчислювальну потужність, яка йому знадобиться. З інженерної точки зору також важливо те, що можлива апаратна реалізація багатопроцесорних спеціалізованих обчислювальних мікроелектронних пристроїв, які ефективно реалізують генетичний алгоритм. А саме, кожній особині популяції можна поставити у

відповідність окремий процесор. Тоді розрахунки пристосованостей можна виконувати паралельним чином, що дозволяє прискорити процес оптимізації.

Отже, щоб краще зрозуміти функціонування генетичного алгоритму, використовується поняття схема та формулюється основна теорема, що відноситься до генетичних алгоритмів і називається теоремою про схеми.

Поняття схеми введено для визначення множини хромосом, що мають деякі спільні властивості, тобто подібних одна до одної [14, 59]. Наприклад, якщо алелі приймають значення 0 або 1 (розглядаються хромосоми з двійковим алфавітом), то схема являє собою множину хромосом, що містять нулі та одиниці на деяких заздалегідь визначених позиціях.

При розгляді схем зручно використовувати розширений алфавіт  $\{0, 1, *\}$ , до якого окрім 0 і 1 введено додатковий символ  $*$ , що позначає будь-яке допустиме значення, тобто 0 або 1. Символ  $*$  у конкретній позиції означає "байдуже". Наприклад, схема  $S: 101 = \{1001, 1011\}$ .

Крім того, визначено також поняття порядку та охоплення схеми. Нехай  $L$  позначає довжину хромосом, що відповідають схемі  $S$ . Порядок схеми  $S$  – це кількість постійних позицій у схемі, тобто нулів і одиниць у випадку алфавіту  $\{0, 1, *\}$ . Наприклад, порядок схеми  $S: 101$  дорівнює трьом. Порядок схеми – це завжди ціле число з інтервалу  $[0, L]$ . Охоплення схеми  $S$  – це відстань між першим та останнім постійним символом (тобто різниця між правою та лівою крайніми позиціями, що містять постійні символи). Наприклад, охоплення схеми  $S: 10 * 1$  дорівнює  $4 - 1$ , тобто 3.

На основі цих понять проведено дослідження операторів селекції, схрещування та мутації і сформульовано наступну теорему про схеми [14, 59]: схеми малого порядку, з малим охопленням та пристосованістю вище середньої формують показниково зростаючу кількість своїх представників у наступних поколіннях генетичного алгоритму.

Опосередкованим результатом цієї теореми можна вважати наступну гіпотезу, що називається гіпотезою про цеглинки (або про будівельні блоки) [14,59]: генетичний алгоритм прагне досягти близького до оптимального

результату за рахунок комбінування добрих схем (з пристосованістю вище середньої) малого порядку та малого охоплення. Такі схеми називаються цеглинками (або будівельними блоками).

Гіпотеза про будівельні блоки висунута на підставі теореми про схеми з урахуванням того, що генетичні алгоритми досліджують простір пошуку за допомогою схем малого порядку та малого охоплення, які згодом беруть участь в обміні інформацією під час схрещування.

Генетичні алгоритми дозволяють здійснювати пошук оптимальних рішень для цілого ряду практичних задач. Широта сфери застосування генетичних алгоритмів зумовлена насамперед їх універсальністю, а також здатністю одночасно оптимізувати розв'язок задачі за декількома критеріями. Ця здатність знайшла своє застосування у великій кількості додатків, включаючи проектування літаків, налаштування параметрів алгоритмів, пошук стійких станів систем нелінійних диференціальних рівнянь тощо [23].

Аналіз результатів використання генетичних алгоритмів дозволяє виділити такі ознаки, за наявності яких задача вирішується генетичним алгоритмом ефективніше, ніж багатьма іншими методами:

- великий простір пошуку, ландшафт якого є негладким (містить декілька екстремумів);
- складно формалізована функція оцінки якості розв'язку;
- багатокритеріальність пошуку;
- пошук за заданими критеріями прийняттого, а не єдиного оптимального розв'язку [23];
- багатопараметричність задачі [45, 46].

У особливий підклас виділяються задачі, які вирішуються в режимі реального часу або близькому до нього (пошук оптимальних маршрутів у багатоагентних системах, керування рухомими об'єктами в реальному часі, оптимізація баз знань нечітких контролерів тощо). Час реалізації генетичного алгоритму є дуже істотним критерієм при вирішенні таких задач.

Прийнято вважати, що генетичні алгоритми не гарантують знаходження глобального оптимуму, проте їх сила полягає в тому, що вони дозволяють «досить швидко» знаходити «досить гарні» розв'язки широкого кола задач, в тому числі і тих, які важко вирішуються іншими методами. Ця властивість робить доцільним застосування генетичних алгоритмів для вирішення різних задач у режимі реального часу.

Взагалі, досягнення генетичним алгоритмом компромісу між якістю розв'язку та обчислювальними витратами – одна з головних переваг методу [6, 45, 46, 66].

Велику популярність здобула видана у 1989 р. і стала класичною книга Д. Гольдберга «Генетичні алгоритми в задачах пошуку, оптимізації та машинного навчання». Вона містить велику кількість прикладів і можливих постановок задач з різних галузей застосувань, що вирішуються за допомогою генетичних алгоритмів.

Останніми роками область застосування генетичних алгоритмів значно розширилася. Ці методи виявляються ефективними при вирішенні таких задач:

- ідентифікація складних динамічних об'єктів;
- вибір оптимальної конфігурації багатоагентних робототехнічних систем;
- синтез оптимальних алгоритмів керування багатоланковими роботами-маніпуляторами;
- оптимальне керування стикуванням космічних апаратів;
- планування маршрутів руху транспортних засобів в умовах перешкод;
- структурний синтез проектних рішень, синтез розкладів [45, 46].

Отже, застосування генетичних алгоритмів охоплює не лише клас традиційних задач оптимізації, а й швидко поширюється на задачі керування складними динамічними об'єктами в умовах невизначеності.

Окремо слід сказати про застосування генетичних алгоритмів при вирішенні задач, що виникають у процесі проектування інтелектуальних систем керування.

Оптимізаційні задачі посідають одне з центральних місць при проектуванні різних класів інтелектуальних систем. Причиною цього є природне прагнення вибрати найпростіший варіант побудови системи чи моделі при дотриманні заданих вимог до якості її функціонування (задача структурного синтезу) або знайти оптимальні налаштування параметрів багатокomпонентної системи за заданої її структури (задача параметричного синтезу) [6].

Привабливість генетичних алгоритмів для вирішення завдань, що виникають при проектуванні інтелектуальних систем, і багатьох інших полягає саме в тому, що вони значною мірою позбавлені зазначених недоліків.

Генетичні алгоритми - метод настільки універсальний, що добрий не лише сам по собі, а й у поєднанні з іншими методами. Безпосередньо способи комбінування методів можуть бути різними. Нижче, як приклад, в загальних рисах розглядається один з варіантів, що отримав в літературі назву «метод комбінування евристик».

Багато практично важливих задач планування, розподілу ресурсів, конструювання зводяться до задач дискретної оптимізації, вирішення яких викликає значні труднощі [30].

Суттєве підвищення точності розв'язку можливе, якщо вихідну задачу трансформувати у задачу пошуку та використання оптимальної послідовності локальних цільових функцій (інакше кажучи, оптимальної послідовності евристик), застосовуваних у підзадачах. При цьому передбачається, що трансформована задача вирішуватиметься генетичними методами. Подібний підхід вже успішно застосовується, наприклад, до задач синтезу розкладів, і названий методом комбінування евристик (НСМ - Heuristics Combination Method) [30, 29].

Як показують дослідження, виконані багатьма авторами, ефективність застосування генетичних алгоритмів значно зростає у випадку їх комбінованого використання з іншими методами теорії штучного інтелекту.

До числа найбільш перспективних напрямків в цій галузі належать дослідження в області «м'яких обчислень» (soft computing) [6]. Цей термін було



запропоновано Л. Заде [17, 86]. Він означає інтеграцію ідей нечіткої логіки, нейронних мереж, генетичних алгоритмів та ймовірнісних міркувань з метою побудови інтелектуальних систем обробки інформації та управління.

Досі не існує суворого наукового обґрунтування генетичних алгоритмів. Природно поставити запитання - чи варто користуватися такими алгоритмами? Тут не можна відповісти однозначно.

З одного боку, в математиці існує досить великий клас абсолютно надійних (в сенсі гарантії отримання точного розв'язку) методів вирішення різних задач.

З іншого боку, мова йде про дійсно складні практичні задачі, в яких ці надійні методи часто непридатні. Чимало цих задач виглядають настільки неосяжними, що не робляться навіть спроби їх осмисленого розв'язання.

Генетичні алгоритми – реалізація однієї з найпопулярніших ідей такого роду. Ця популярність викликана, мабуть, виключною красою підходу та його близькістю до природного механізму [66].

Як аргументи на користь генетичних алгоритмів можна навести наступні міркування. Генетичний алгоритм – це «далекий родич» методу імітації відпалу, обтяжений великою кількістю параметрів (в методі відпалу єдиний параметр – «температура») [73].

Результати численних порівняльних досліджень такі, що генетичні алгоритми в середньому розв'язують задачі оптимізації швидше та/або краще за метод відпалу [84]. Причому, чим складніша структура задачі, тим більше програє метод відпалу.

Справедливості заради, слід визнати, що у багатьох задачах геном особини та операції мутації і рекомбінації влаштовані таким чином, що не відомий точно характер наслідків цих операцій. Далеко не завжди, наприклад, з повною впевненістю ми можемо стверджувати, що прямі нащадки двох «високопридатних» батьків будуть «високопридатними». Та сама ситуація і з масштабом мутації: не завжди зрозуміло, чи призведе мала мутація до такого ж малого зміни значення корисності, до малої зміни реальних властивостей реального об'єкта.

Однак є й задачі, в яких значення та зміна геномів особин передбачувані. Наприклад, оптимізація розташування підписів на географічній карті. Очевидно, що якщо одна особина має добре "підписану" західну ділянку карти, а інша - східну, то їх нащадок, що поєднує в собі обидві половини, матиме високе значення придатності.

Приймаючи рішення щодо використання генетичного алгоритму як методу оптимізації, в будь-якому разі слід пам'ятати, що на сьогодні вже не викликає сумнівів, що генетичні алгоритми працюють добре у всіх галузях, де їх взагалі доцільно застосовувати [84, 66].

Часто ставиться і інше питання - а таки доцільна розробка таких "швидких" евристичних алгоритмів оптимізації, як генетичні алгоритми?

По-перше, при великих розмірностях задач, що розв'язуються, поліноміальний алгоритм (наприклад, генетичний) працює на порядки швидше за експоненціальні.

По-друге, обчислювальна техніка поки розвивається істотно повільніше, ніж розробляється алгоритм.

По-третє, багатопроцесорні системи існують і успішно експлуатуються, в тому числі, до речі, і для більш ефективної реалізації генетичних алгоритмів [66, 45, 46].

Генетичні алгоритми, генетичне програмування, еволюційні стратегії та еволюційне програмування є чотирма основними формами моделювання еволюції. Кожна з цих форм має свої відмінні риси та особливості. Цей розділ присвячено лише генетичним алгоритмам. Але загальні для всіх еволюційних алгоритмів переваги й недоліки поширюються і на них. Переваги еволюційних алгоритмів: широка область застосування еволюційних алгоритмів; можливість проблемно-орієнтованого кодування рішень, підбору початкової популяції, комбінування еволюційних алгоритмів з нееволоційними алгоритмами, продовження процесу еволюції доки є необхідні ресурси; придатність для пошуку в складному просторі рішень великої розмірності; відсутність обмежень на вид цільової функції; ясність схеми та базових принципів еволюційних

алгоритмів; інтегрованість еволюційних алгоритмів з іншими неklasичними парадигмами штучного інтелекту, такими як штучні нейронні мережі та нечітка логіка.

Недоліки еволюційних алгоритмів: евристичний характер еволюційних алгоритмів не гарантує оптимальності отриманого рішення (правда, на практиці часто важливо за заданий час отримати одне чи кілька майже оптимальних альтернативних рішень, тим більше що вихідні дані в задачі можуть динамічно змінюватися, бути неточними чи неповними); відносно висока обчислювальна трудомісткість, причинами якої є те, що в ході моделювання еволюції багато рішень відкидаються як неперспективні (тим не менш численні експериментальні дані вказують на те, що часова складність еволюційних алгоритмів в середньому нижче, ніж у кращих конкуруючих алгоритмів, але не більше ніж на один порядок); невисока ефективність еволюційних алгоритмів на заключних фазах моделювання еволюції (особливо це стосується генетичних алгоритмів); це пояснюється тим, що оператори пошуку в еволюційних алгоритмах не орієнтовані на швидке потрапляння у локальний оптимум.

Незважаючи на те, що конструювання хромосом і функцій придатності може вимагати значних зусиль, генетичні алгоритми легко реалізуються навіть з нуля і здатні розв'язувати широке коло задач. Використовуючи аналогію з розвитком живих організмів від простих форм до складніших, генетичні алгоритми наблизилися до того, щоб стати загальним методом розв'язання задач.

### РОЗДІЛ 3

## ПОБУДОВА ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ПОШУКУ СУБОПТИМАЛЬНИХ РОЗВ'ЯЗКІВ ЗАДАЧІ ПЛОСКОГО РОЗКРОЮ СТРІЧКИ НА ПРЕДМЕТИ НЕСКЛАДНОЇ ГЕОМЕТРИЧНОЇ ФОРМИ В МЕТАЛУРГІЇ

3.1 Генетичний алгоритм вирішення задачі плоского розкрою стрічки на предмети нескладної геометричної форми

Серед усіляких задач розкрою можна виділити два класи. Перший – це задачі, в яких задані кількість і параметри розкрійних об'єктів та асортимент вирізуваних предметів, потрібно знайти кількість і місцезнаходження предметів кожного виду, що мінімізує втрати розкрою. Другий клас – це задачі, в яких заданий і асортимент предметів, і кількість предметів кожного виду, і параметри розкрійних об'єктів, а знайти потрібно розташування предметів на об'єктах, мінімізуючи кількість об'єктів або незайняту частину об'єктів. У даній роботі розглядаються тільки задачі другого класу. Ставиться задача двовимірного розкрою напівнескінченної смуги на набір предметів нескладної форми. Предметами нескладної геометричної форми назвемо предмети такого вигляду: квадрати і прямокутники; кола і еліпси; трикутники; складні предмети з декількох (2-х, 3-х) таких, що торкаються сторонами, прямокутників; еліпси і прямокутники, контури яких видозмінені симетричним усіченням частини фігури по прямій або дузі кола.

Змістовна постановка задачі. Є прямокутна смуга певної ширини та необмеженої довжини, а також набір предметів нескладної геометричної форми. Ширина смуги, кількість, форма і розміри предметів задані. Потрібно визначити параметри оптимального розташування предметів у смугі за таких умов:

- предмети не перетинаються один з одним;
- предмети не перетинаються зі сторонами смуги.

Під оптимальним розташуванням розуміється таке, при якому зайнята всіма предметами з набору довжина смуги мінімальна. Поставлена задача розкрою є NP-повною, як і всі задачі розкрою та упакування. Як метод розв'язання обраний генетичний алгоритм. Цей метод примітний не тільки тим, що знаходить відносно гарне рішення за відносно малий час. Крім того, генетичний алгоритм може успішно працювати зі структурою даних, що не містить інформації про геометрію предметів. Якщо потрібно упакувати у смугу предмети деякого списку замовлень, то для роботи алгоритму ці предмети вважаються пронумерованими, наприклад, у порядку їхнього слідування у списку.

Як хромосому, що описує особину, виступатиме будь-яка послідовність номерів фігур без повторів (далі – пріоритетний список, *priority list*, PL). Такий опис відповідає біологічному терміну генотип. Саме по собі воно не дає можливість зрозуміти, як виглядає упаковка, котра відповідає певному порядку переліку предметів. Фенотип особини в даному випадку – це геометричне розміщення предметів на смугі, задане набором координат розташування кожного предмету. Лише за фенотипом особини можна судити про якість упаковки – про частку корисної площі, частку площі втрат, довжину відповідної упаковки.

Щоб генотип перетворити у фенотип, потрібний набір правил, згідно з якими можна розміщувати предмети на стрічці у порядку їх переліку в пріоритетному списку. Ці правила називатимемо процедурою декодування або декодером. Очевидно, що декодер не повинен породжувати оптимальне розміщення предметів, проте ця процедура має розміщувати предмети щільно, без очевидних прогалин і втрат корисної площі. Наприклад, при упакуванні прямокутників кожен наступний предмет з хромосоми можна вважати досить добре укладеним, якщо його не можна зсунути у повздовжньому напрямку до початку смуги і у поперечному напрямку він теж щільно примикає до якогось іншого предмету чи краю стрічки.

Нижче наведено схему розробленого генетичного алгоритму, що використовує декодер як параметр, відповідальний за геометрію предметів, які

упаковуюються, та обчислює значення функції пристосованості для кожної конкретної особини:

1) Створити початкову популяцію з  $N$  особин. Кожна особа задається випадковою перестановкою номерів предметів зі списку замовлень (3.1):

$$S_k = \{S_{k1}, S_{k2}, \dots, S_{km}\}, k = 1/N, \quad (3.1)$$

де  $m$  – число предметів,

$$S_{ki} = 1/N - \text{номер предмета}, S_{ki} \neq S_{kj} \forall k, \forall i \neq j, j = 1/m$$

2) Здійснити схрещування всіх особин з випадковим для кожної особини партнером. Зберегти всіх нащадків ( $2N$  особин).

3) Обчислити значення пристосованості для кожної особи, включно з нащадками, за допомогою процедури декодування (3.2):

$$f(S_k) = \text{decoder}(S_k), k = 1/3N. \quad (3.2)$$

4) Впорядкувати значення пристосованості за спаданням, залишити  $N$  кращих особин.

5) Застосувати оператор мутації до кожної особини з малою ймовірністю  $\mu$ .

6) Повторювати 2 – 5 доти, доки найвище значення функції пристосованості в популяції не перестане змінюватися. Передбачено ще два способи завершення еволюції: якщо виконано  $m$  поколінь і примусове завершення (користувач програмного забезпечення натискає відповідну кнопку).

Процедура схрещування для такого алгоритму також не залежить від геометрії предметів і здійснюється за наступною схемою.

Нехай  $S_k, S_l$  – дві хромосоми, два списки номерів предметів, що відповідають парі батьків  $S_{knew}$ , і  $S_{lnew}$  – два нові списки, створені схрещуванням. Спочатку вибираються два випадкові числа  $g$  і  $q$ ,  $1 \leq g, q \leq m$ . Починаючи з випадкової позиції  $g$ , оператор схрещування копіює  $q$  елементів з  $S_k$

в  $S_{knew}$ . Потім решту позицій у  $S_{knew}$  заповнюються іншими елементами з  $S_l$ , в тій самій послідовності, в якій вони зустрічаються в цьому списку. Аналогічно створюється другий нащадок пари батьків  $S_k, S_l$ . Оператор мутації обмінює два випадкові елементи кожного списку з невеликою ймовірністю  $\mu$ . Для прямокутних предметів доречно ввести додатковий оператор мутації – поворот випадкового предмета на 90 градусів.

Взагалі, генетичний алгоритм і раніше використовувався для вирішення задач розкрою та упакування, а саме розглядалися задачі одновимірною та ортогонального розкрою. Судячи з результатів порівнянь різних методів, генетичний алгоритм добре себе зарекомендував [36]. Однак застосування його для вирішення задач ортогонального розкрою наклало свій відбиток. З одного боку, метод показує чудові результати при упакуванні прямокутників, але з іншого боку, абсолютно непридатний для вирішення будь-яких інших задач розкрою.

У роботі пропонується схема еволюційного процесу, що легко реструктуризується і переноситься на різні, як буде показано нижче, задачі розкрою. Розроблений метод відрізняється від відомих модифікацій в основному наступними характеристиками.

Генетичний алгоритм проводить схрещення доти, доки є покращення в популяції. Згідно експериментів, проведених у цій роботі, це призводить до виродження генофонду в нащадки однієї – двох локально кращих особин. З такою проблемою можна боротися за рахунок виділення в популяції практично не взаємодіючих у процесі еволюції груп. Такий підхід робить алгоритм більш громіздким і повільним. Отже, пропонується після кожного набору схрещень, що відповідає одному поколінню, проводити аналогічний набір мутацій з малою ймовірністю для кожної особини. Це зменшує обчислювальні витрати і перешкоджає виродженню.

Генетичний алгоритм використовувались для вирішення задач, в яких можна обчислити нижню межу для результату розв'язання. Ця можливість активно використовувалась для, наприклад, завершення роботи генетичного алгоритму. У випадку розкрою на предмети круглої форми така можливість

зникає. У зв'язку з чим, знадобилося розробляти набір ознак стану популяції для зупинки генетичного алгоритму. Наприклад, відстежується кількість поколінь, протягом яких значення пристосованості для кращої особини в популяції не збільшується. Знадобилося експериментально встановлювати прийнятне значення цієї кількості, при якому слід завершити процес еволюції.

Генетичний алгоритм використовує оператор мутації, в якому відбувається виклик процедури декодування і обробка фізичної карти розкрою, орієнтована на прямокутну форму предметів і об'єкта. Це не тільки збільшує обчислювальні витрати, а й звужує область застосування такого оператора мутації до єдиної задачі ортогонального розкрою.

Генетичний алгоритм доповнено набором детермінованих процедур, орієнтованих на ортогональність розкрою та інформацію про нижню межу розв'язку. Їх не можна використовувати у випадку вирішення задачі розкрою, наприклад, на круглі предмети.

Таким чином, розроблено модифікацію генетичного алгоритму для вирішення поставленого завдання, що практично не залежить від геометрії предметів.

### 3.2 Програмна реалізація розв'язку задачі плоского розкрою стрічки на предмети нескладної геометричної форми на базі генетичний алгоритму

Задача оптимального розкрою матеріалів є важливою в багатьох галузях промисловості, зокрема у металургійній. Суть її полягає у розміщенні на площині заготовки (листа, стрічки) заданої форми набору деталей певної конфігурації так, щоб відхід матеріалу був мінімальним. Ця задача належить до класу NP-складних оптимізаційних задач комбінаторної оптимізації.

Існує чимало програмних продуктів для її розв'язування. Серед найбільш відомих – OptiNest, JETCAM Expert і SilverNest. Вони дозволяють автоматизувати процес розкрою, прискорити його та підвищити ефективність використання матеріалу. Ці програми реалізують різні евристичні алгоритми



та метаевристики для знаходження наближених розв'язків: жадібні алгоритми, моделювання відпалу, генетичні алгоритми тощо.

Зазвичай такі продукти мають зручний графічний інтерфейс, що дозволяє користувачу створювати бібліотеки деталей та заготовок, задавати параметри і обмеження, візуально контролювати процес розкрою. Крім автоматичного режиму передбачено й ручне розміщення елементів. Після розкладання можна аналізувати якість розкрою, відсоток використання матеріалу, генерувати карту розкрою та виводити керуючі програми для різального обладнання.

Деякі системи, як-от JETCAM, інтегровані з САПР і дозволяють працювати безпосередньо з 3D-моделями деталей. Є можливість оптимізації з врахуванням напрямку волокон та інших особливостей матеріалу.

Перераховані програми цілком доступні для придбання українськими користувачами. Їх може опанувати будь-який технолог, знайомий з комп'ютером та геометрією. Однак при ретельнішому розгляді виявляється, що переважна більшість запропонованих програм орієнтовані на розв'язання задач одномірного або ж двовимірного прямокутного розкрою. Більше того, багато з них передбачають ручне складання карти розкрою на ПК. Звідси можна зробити висновок, що нині в Україні програмне забезпечення, орієнтоване на довільну геометричну форму предметів, тільки починає з'являтися та експлуатуватися. Тим часом у величезній кількості організацій плани розкрою і далі складаються не сучасною ефективною програмою, а зусиллями операторів ПК. Розкроювати вихідний матеріал на деталі різної форми потрібно при розв'язанні різних промислових задач. Майже завжди невелика зміна у геометрії предметів вимагає розробки принципово нових схем, нових методів та математичних моделей для побудови плану розкрою. При заміні обладнання на сучасніше, при перепрофілюванні відділу підприємства, при зміні асортименту продукції, що випускається, при появі замовлень нового типу розроблений загальний підхід до розв'язання різних задач плоского розкрою дозволяє звести до мінімуму фінансові та

інтелектуальні витрати на розробку відповідних математичних і програмних засобів автоматизації процесу раціонального розкрою.

Програмна реалізація задачі плоского розкрою стрічки на нескладні деталі є важливим і нетривіальним завданням, що потребує залучення сучасних методів та інструментів оптимізації.

По-перше, основою для створення програмного забезпечення розкрою є формалізація задачі у вигляді математичної моделі. Зазвичай застосовують моделі цілочислового, нелінійного або динамічного програмування. Вони дозволяють врахувати суттєві параметри задачі: розміри та форму листів і деталей, критерій мінімізації відходів, технологічні обмеження тощо.

По-друге, на базі цих моделей розробляють алгоритмічне забезпечення для розв'язання задачі. Можуть застосовуватися як точні (точні методи оптимізації), так і наближені алгоритми (евристики, генетичні алгоритми). Також ефективним є підхід гібридизації – поєднання різних методів для досягнення кращих результатів.

По-третє, реалізують засоби аналізу вихідних даних для автоматичної оцінки складності конкретної задачі розкрою і вибору доцільного методу розв'язання. Адже на практиці задачі можуть змінюватися в широкому діапазоні параметрів.

По-четверте, розробляють гнучкий інтерфейс користувача для зручного задання вихідних даних і візуалізації результатів розкрою. Це може бути як окремий модуль або система, так і складова загальної ERP чи MES системи підприємства.

По-п'яте, оскільки задачі оптимального розкрою часто є обчислювально складними, значну увагу приділяють аспектам швидкодії та масштабованості програмних рішень. Застосовують підходи розподілених обчислень та паралельних обчислювальних архітектур (кластери, хмари тощо).

Отже, програмна реалізація задачі оптимального розкрою стрічки є комплексним багаторівневим процесом. Він потребує синергії математичного

моделювання, алгоритмізації, програмування та комп'ютерної інженерії для створення ефективних додатків, придатних для промислового застосування.

Для вирішення розв'язку задачі плоского розкрою стрічки на предмети нескладної геометричної форми на базі розробленого генетичного алгоритму був розроблений програмний додаток.

Додаток складається з трьох вікон (рис. 3.1):

- головне («раціональний плоский розкрій»);
- вікно параметрів («Параметри»);
- вікно результатів («Результат»).

Головне вікно містить меню, що складається з пунктів:

- а) «Прямокутний розкрій»;
- б) «Круглий розкрій»;
- в) «Карта розкрою»;
- г) «Про програму».

Перші два пункти дозволяють вибрати вид задачі. Коли задача вибрана, автоматично з'являється вікно «Параметри». В ньому чотири розділи:

- а) «Поточний розкрій»;
- б) «Параметри розкрою»;
- в) «Файл пакета замовлень»;
- г) «Стан еволюції».

У «Поточний розкрій» відображається назва вибраної задачі. Всі параметри задаються з файлу. Щоб вибрати файл, слід натиснути кнопку «Вибрати пакет замовлень» в розділі «Параметри розкрою». Автоматично з'явиться діалогове вікно «Вибір файлу пакета замовлень». Даним для задач прямокутного розкрою відповідають файли з розширенням «*grp*», круглого – «*ср*». Якщо файл пакета замовлень вибрано, слід натиснути кнопку «Ок» у поточному вікні. У розділі «Файл пакета замовлень» відобразиться його назва.

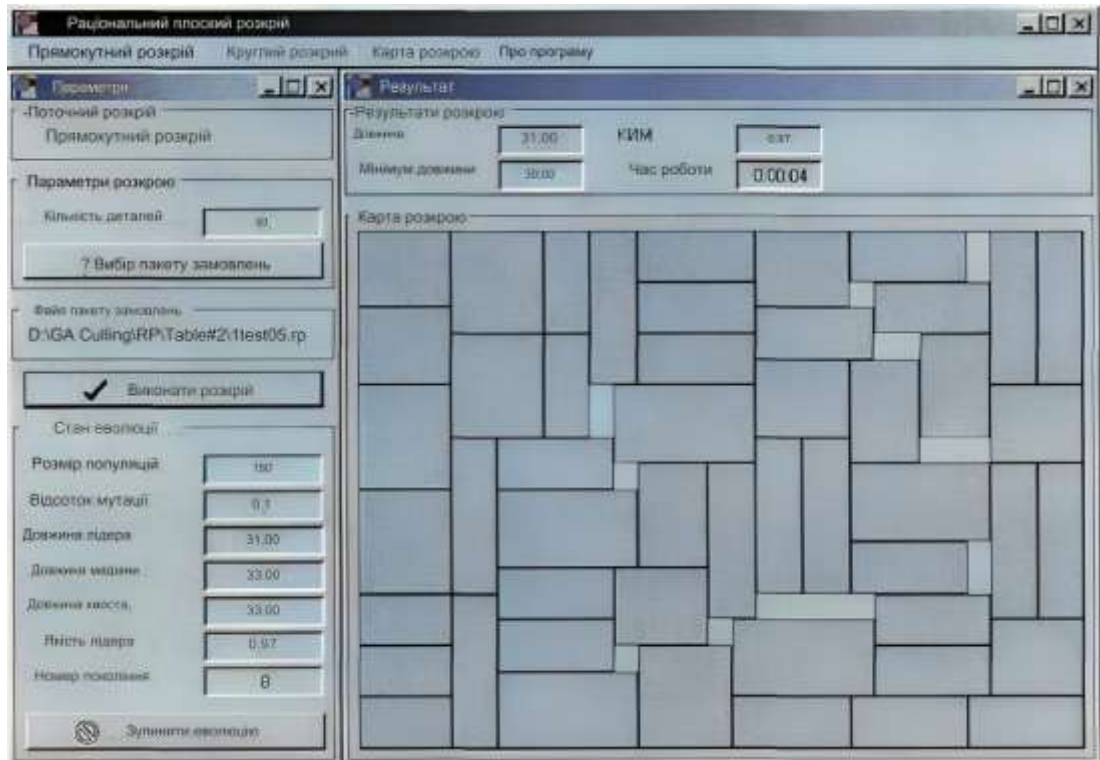


Рисунок 3.1 – Приклади роботи програмного забезпечення

Щоб розв’язати поставлену задачу розкрою, слід натиснути кнопку «Виконати розкрій» у вікні «Параметри». Запуститься генетичний алгоритм. У вікні «Параметри» в розділі «Стан еволюції» з’явиться поточна інформація про процес еволюції. Генетичний алгоритм закінчує роботу в будь-якому з чотирьох випадків:

- а) краще значення функції пристосованості перестало змінюватися;
- б) досягнуто відомий оптимум (для тестових задач);
- в) пройшло задане в програмі число поколінь еволюції (пропорційне кількості предметів у пакеті замовлень);
- г) користувач примусово зупинив еволюцію натисканням кнопки «Зупинити еволюцію» в розділі «Стан еволюції».

Якщо генетичний алгоритм закінчив роботу, автоматично з’являється вікно «Результат», в якому представлені розділи: «Результати розкрою» та «Карта розкрою». У розділі «Результати розкрою» виводяться:

- довжина отриманої упаковки;
- розрахунковий мінімум довжини;

- якість упаковки (частка корисної площі або коефіцієнт використання матеріалу);

- час роботи алгоритму.

«Карта розкрою» – це піксельний малюнок у відповідному масштабі, що зображує фізичне розміщення предметів на смузі.

Розділ меню «Карта розкрою» викликає вікно «Результат», якщо воно було закрите. Вікно «Параметри» з'являється автоматично кожного разу при виборі типу задачі.

Розділ меню «Про програму» містить інструкції з експлуатації програми.

Програма працює з будь-яким пакетом замовлень, заданим у звичайному текстовому файлі за формою (Додаток А):

- [ширина смуги];

- [число деталей першого виду], [довжина деталей першого виду], [ширина деталей першого виду];

- [число деталей другого виду], [довжина деталей другого виду], [ширина деталей другого виду];

- [число деталей останнього виду], [довжина деталей останнього виду], [ширина деталей останнього виду];

Дані про результат роботи та процес еволюції також виводяться у файл, в туж директорію, де знаходиться виконуваний файл програми.

### 3.3 Оцінка ефективності розробленого алгоритму для розв'язання задач плоского розкрою стрічки на предмети нескладної геометричної форми

За допомогою розробленого програмного забезпечення поставлено низку обчислювальних експериментів, мета яких – оцінити швидкодію та якість роботи запрограмованих алгоритмів. Серед різних способів оцінки якості можна відзначити такі:

– візуальний – карта розкрою повинна бути досить щільною і не повинна містити порожнеч, очевидних для заповнення крайовими предметами упаковки. Такий неформалізований підхід дозволяє запобігти марну витрату сил і часу на аналіз результатів роботи «неналагодженої» програми;

– порівняльний (тестування на наборах даних з відомим результатом) – досягнення відомого оптимуму позитивно характеризує розроблений алгоритм;

– числовий – при упаковці довільних наборів предметів алгоритм характеризується певними показниками якості, наприклад, часткою корисної площі в упаковці, чим краще працює алгоритм, тим вище буде такий показник якості;

– часовий – абсолютний час роботи характеризується конкретною тривалістю обчислень для тестів різної розмірності, під відносним часом роботи будемо розуміти вигляд залежності часу обчислень від розмірності задачі;

– спеціальний для генетичних алгоритмів – кількість поколінь, за яке досягається прийнятне рішення, не повинно бути дуже великим і характеризує кожен конкретну модифікацію алгоритму.

Розмірністю задачі плоского розкрою будемо вважати сумарну кількість предметів всіх розмірів  $m$ . Якість рішення будемо оцінювати за допомогою частки корисної площі. Це відношення сумарної площі предметів до площі зайнятої упаковкою прямокутної частини смуги (формула (2.4)).

Будемо вважати погано обумовленими такі задачі та набори предметів, поєднання параметрів в яких несприятливо позначається на якості упаковки. Наприклад, розміри предметів, що упаковуються, можуть бути співмірними із шириною смуги, а можуть бути на порядок менше. Якщо поперек ширини смуги можна покласти лише 2-4 предмети, то можливостей підібрати їх вдале поєднання менше, ніж у ситуації, де поперек смуги вміщується 10-15 предметів. Заздалегідь очевидно, що у другому випадку показники якості

будуть вище, ніж у першому. Першу задачу можна назвати погано обумовленою.

Для прямокутного розкрою довжина і ширина кожного предмета також можуть бути спів мірними, а можуть сильно різнитися. В середньому, «квадратні» предмети упаковуються краще «прямокутних». Набір вузьких довгих предметів теж можна назвати погано обумовленим. Назвемо прийнятним (досить хорошим) таке рішення задачі розкрою, для якого частка корисної площі не менше значень 0,93 (для представленого у додатку набору погано обумовлених тестових задач) і 0,97 (для добре обумовлених) для прямокутного і 0,78 для круглого розкрою. Ці значення отримані в результаті роботи програми на безлічі тестів і являють собою середнє від кращих результатів для кожної задачі. При отриманні найкращого результату для кожного тестового набору предметів інші параметри алгоритму варіювалися наскільки завгодно широко, в тому числі час роботи алгоритму не брався до уваги і міг досягати недоцільно великих значень.

Будемо вважати, що алгоритм знаходить рішення за досить малий час, якщо він досягає відомого оптимуму або прийнятного рішення за число поколінь, що не перевищує число предметів, тобто розмірності задачі.

Задача прямокутного розкрою характеризується певним (можливо, недосяжним) значенням квазіоптимуму, яке можна обчислити як частку загальної площі заготовок та ширини смуги (3.3):

$$L_{opt} = \frac{\sum_{i=1}^m w_i l_i}{W} \quad (3.3)$$

Окрім того, як тестовий набір заготовок у задачі прямокутного розкрою можна використати набір, отриманий внаслідок розбиття прямокутника шириною  $W$  та довжиною  $L_{test}$ . Для отриманого набору заготовок відоме не лише числове значення оптимуму, рівне квазіоптимуму:  $L_{opt} = L_{test}$ , а й за

визначенням існування цього оптимуму з певним конкретним варіантом відповідної йому карти розкрою.

Якість розкрою оцінюватимемо за допомогою вже згаданого вище коефіцієнта корисної площі за формулою (2.4).

Для декількох таких наборів із відомим оптимумом проведено обчислювальний експеримент (табл. 3.1, рис. 3.2).

Таблиця 3.1 – Результати для наборів з відомим оптимумом

| Файл        | $W$ | $L_{opt}$ | $L$ | $m$ | $\eta$ | $t$ (мін: сек) |
|-------------|-----|-----------|-----|-----|--------|----------------|
| 05test05.rp | 10  | 7         | 7   | 5   | 1.00   | 00:00          |
| 10test10.rp | 10  | 13        | 13  | 10  | 1.00   | 00:00          |
| 10test20.rp | 10  | 26        | 26  | 20  | 1.00   | 00:02          |
| 10test30.rp | 10  | 39        | 39  | 30  | 1.00   | 00:11          |
| 10test40.rp | 10  | 52        | 53  | 40  | 0,98   | 00:18          |
| 10test50.rp | 10  | 65        | 66  | 50  | 0,98   | 00:25          |
| 07test07.rp | 20  | 30        | 30  | 7   | 1.00   | 00:00          |
| 23test23.rp | 20  | 30        | 30  | 23  | 1.00   | 00:05          |
| 24test24.rp | 20  | 30        | 31  | 24  | 0.97   | 00:14          |
| 30test30.rp | 20  | 32        | 33  | 30  | 0.97   | 00:16          |
| 37test37.rp | 20  | 30        | 31  | 37  | 0.97   | 00:19          |
| 51test51.rp | 20  | 30        | 31  | 51  | 0.97   | 00:28          |

Він показав, що майже для всіх наборів знаходження оптимуму можливе за малий час, але оскільки алгоритм має елемент випадковості і пошук може розгортатися в різних напрямках, то безпосереднього досягнення оптимуму можна домогтися при другому, третьому, можливо четвертому запуску алгоритму. Отримання оптимуму з першого разу можна забезпечити збільшенням кількості поколінь еволюції, але це трохи складніше технічно і істотно довше за часом. Крім того, якщо все ж оптимум невідомий, то невідомо також, до якої міри чекати. Отже, практично для всіх наборів задачі прямокутного розкрою генетичний алгоритм ефективно знаходить близькі до оптимальних або оптимальні розв'язки за прийнятний час. Подальше поліпшення рішень можливе шляхом повторних запусків алгоритму, але потребує додаткових обчислювальних витрат.



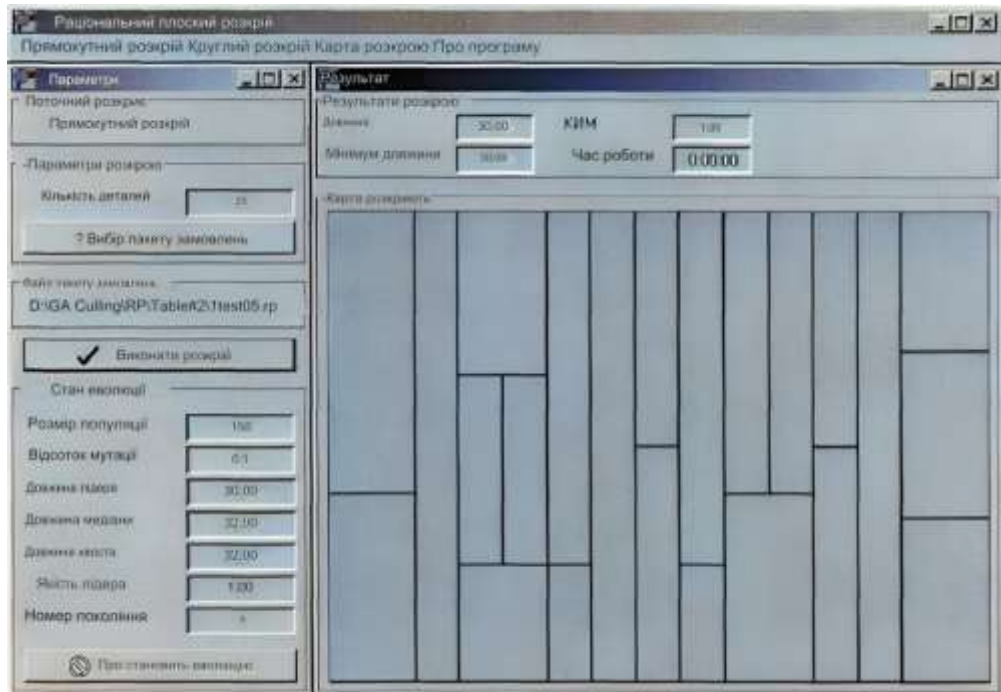


Рисунок 3.2 – Результат роботи генетичного алгоритму з блоковим декодером

Причиною, чому генетичний алгоритм може не досягти відомого оптимуму, формулюється так: не існує шляху до оптимуму зі сформованого генофонду популяції за допомогою сформульованих операторів схрещування та відбору, а оператор мутації в силу своєї випадковості ніяк не дає саме того оновлення генофонду, яке для цього необхідно. У такій ситуації слід пам'ятати, що в загальному вигляді оптимум невідомий, а реалізований алгоритм у всіх випадках досягає досить високої якості упаковки. З табл. 3.1 можна зробити два очевидних висновки: зі зростанням розмірності задачі час роботи алгоритму зростає; зі зростанням розмірності знаходження наявного оптимуму стає менш ймовірним, але якість рішення залишається досить високою. Тестування проводилося для наборів з цілими значеннями параметрів смуги і заготовок та цілим значенням оптимуму. У зв'язку з цим спостерігається дискретність параметра  $\eta$ . Наприклад, якщо  $L_{opt} = 30,00$ , а  $L = 30,00$ , то  $\eta = 0,97$  і в цій задачі ні за яких обставин не може бути  $\eta = 0,98$  або  $\eta = 0,99$ , оскільки довжина упаковки теж має бути цілим числом.

Крім використання тестових задач, в яких оптимум відомий з побудови пакету замовлень, існує ще один спосіб тестувати роботу методу прямокутного розкрою. Якщо всі задані заготовки мають цілі значення довжини і ширини, то їх сумарна площа теж ціле число. Навіть якщо ширина смуги також ціла, то все одно для більшості пакетів замовлень їх частка, тобто «мінімально можлива» зайнята набором заготовок довжина смуги за припущення відсутності пустот, число неціле. Однак, оскільки всі розміри заготовок цілі, то будь-яка їх комбінація займає довжину смуги, що має ціле значення. Тобто, в такій задачі оптимумом можна вважати найближче ціле число зверху до значення «мінімально можливої» довжини. Розроблене програмне забезпечення успішно пройшло тестування і за цим критерієм.

У задачі круглого розкрою, в силу геометрії заготовок, неможливий розрахунок квазіоптимуму, навіть недосяжного, за аналогією із задачею прямокутного розкрою. Однак існують деякі набори заготовок з відомим оптимумом. Перший набір відповідає упаковці кіл однакового радіусу в напівнескінченну стрічку з шириною, кратною цьому радіусу. За достатньо великої різниці радіусів кіл і ширини стрічки (наприклад, на порядок), відоме найбільш раціональне розташування кіл - шахове (або коміркове). Довжина упаковки, що виходить за такого розташування, і є оптимумом. Розроблений алгоритм вирішує таку задачу за одне покоління еволюції і досягає оптимуму. Строго кажучи, в даному випадку сама еволюція не відіграє ніякої ролі. З точки зору використовуваного декодера в розглянутій задачі всі особини ідентичні. Крайовий декодер влаштований таким чином, що викладає кола на стрічку саме у шаховому порядку. На рис. 3.3 проілюстровано результати роботи програми при вирішенні розглянутої задачі.

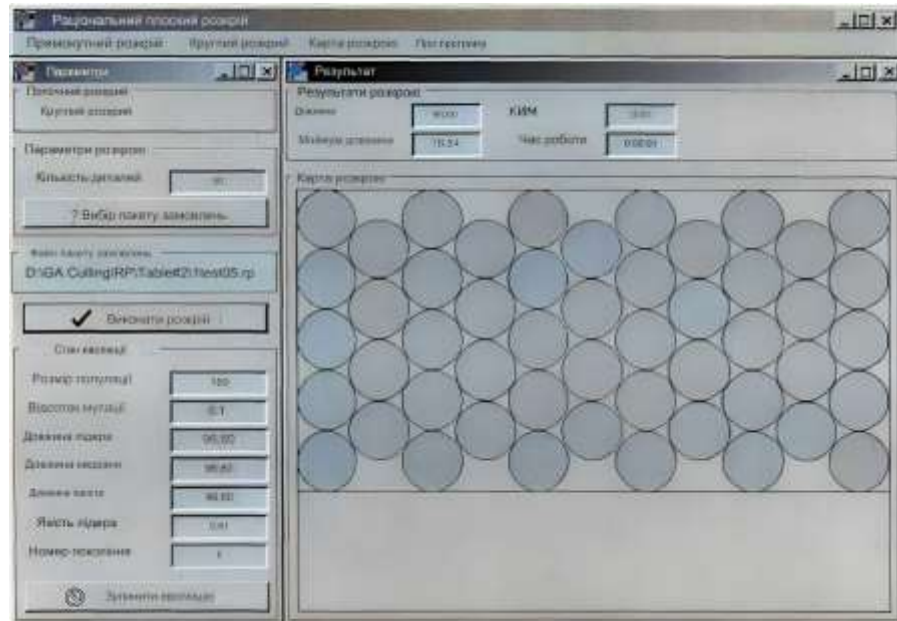


Рисунок 3.3 – Результат роботи крайового декодера

Тестування алгоритму для наборів однакових кіл і шириною смуги, не кратній їхньому радіусу, показало, що генетичний алгоритм (точніше декодер) упакує їх краще, ніж при класичному шаховому розкрії. Підвищення щільності упаковки досягається за рахунок зрушень на початок і на незаповнені краї смуги «трикутних» груп кіл (рис. 3.4).

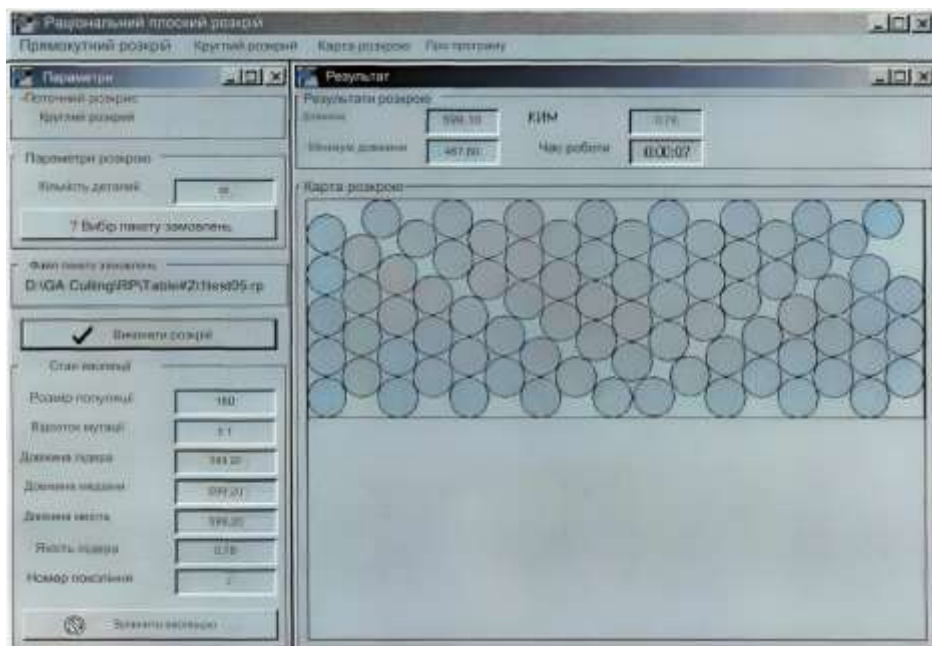


Рисунок 3.4 – Результат роботи крайового декодера

Обчислювальний експеримент показав, що коефіцієнт  $\eta$  залежить від зумовленості задачі та її розмірності. У табл. 3.2-3.3 частково наведені результати тестування для задач різних класів. В середньому прямокутний розкрій генетичним алгоритмом з блоковим декодером для добре обумовлених задач працює не гірше, ніж з показником  $\eta = 0,97$ , для погано обумовлених -  $\eta = 0,93$ .

Таблиця 3.2 – Прямокутний розкрій, погано обумовленої задачі

| №          | $W$ | $L_{opt}$ | $L$ | $m$ | $\eta$ | $t$ (мін: сек) |
|------------|-----|-----------|-----|-----|--------|----------------|
| 0test01.rp | 64  | 47,89     | 50  | 6   | 0,96   | 00:02          |
| 0test02.rp | 64  | 95,78     | 99  | 12  | 0,97   | 00:05          |
| 0test03.rp | 64  | 143,7     | 149 | 18  | 0,96   | 00:09          |
| 0test04.rp | 64  | 191,6     | 199 | 24  | 0,96   | 00:13          |
| 0test05.rp | 64  | 239,5     | 255 | 30  | 0,94   | 00:16          |
| 0test06.rp | 64  | 287,3     | 306 | 36  | 0,94   | 00:17          |
| 0test07.rp | 64  | 335,2     | 358 | 42  | 0,94   | 00:24          |
| 0test08.rp | 64  | 383,1     | 408 | 48  | 0,94   | 00:31          |
| 0test09.rp | 64  | 431,0     | 460 | 54  | 0,94   | 00:35          |
| 0test10.rp | 64  | 478,0     | 510 | 60  | 0,94   | 00:42          |
| 0test11.rp | 64  | 526,8     | 565 | 66  | 0,93   | 00:46          |
| 0test12.rp | 64  | 574,7     | 619 | 72  | 0,93   | 00:52          |
| 0test13.rp | 64  | 622,6     | 657 | 78  | 0,95   | 01:03          |
| 0test14.rp | 64  | 670,5     | 715 | 84  | 0,94   | 01:08          |
| 0test15.rp | 64  | 718,5     | 760 | 90  | 0,95   | 01:14          |

Таблиця 3.3 – Прямокутний розкрій

| №          | $W$ | $L_{opt}$ | $L$ | $m$ | $\eta$ | $t$ (мін: сек) |
|------------|-----|-----------|-----|-----|--------|----------------|
| 1test01.rp | 20  | 6         | 6   | 10  | 1,00   | 00:01          |
| 1test02.rp | 20  | 12        | 12  | 20  | 1,00   | 00:05          |
| 1test03.rp | 20  | 18        | 19  | 30  | 0,95   | 00:12          |
| 1test04.rp | 20  | 24        | 24  | 40  | 1,00   | 00:05          |
| 1test05.rp | 20  | 30        | 31  | 50  | 0,97   | 00:23          |
| 1test06.rp | 20  | 36        | 37  | 60  | 0,97   | 00:25          |
| 1test07.rp | 20  | 42        | 43  | 70  | 0,98   | 00:41          |
| 1test08.rp | 20  | 48        | 49  | 80  | 0,98   | 00:45          |
| 1test09.rp | 20  | 54        | 55  | 90  | 0,98   | 00:55          |
| 1test10.rp | 20  | 60        | 61  | 100 | 0,98   | 00:57          |

Проведено тестування алгоритму для найрізноманітніших наборів заготовок. Різних за кількістю, за асортиментом, за співвідношенням середнього діаметра до ширини смуги. У табл. 3.4-3.5 частково наведені

результати тестування для задач різних класів. Як середнє значення частки корисної площі отримано 0,78. Для задачі круглого розкрою цей результат досить добрий.

Таблиця 3.4 – Круглий розкрий

| №          | $W$ | $L$    | $m$ | $\eta$ | $t$ (мін: сек) |
|------------|-----|--------|-----|--------|----------------|
| 1test01.cp | 100 | 39,59  | 8   | 0,73   | 00:04          |
| 1test02.cp | 100 | 74,07  | 16  | 0,78   | 00:14          |
| 1test03.cp | 100 | 111,17 | 24  | 0,78   | 00:27          |
| 1test04.cp | 100 | 146,60 | 32  | 0,79   | 00:42          |
| 1test05.cp | 100 | 184,27 | 40  | 0,79   | 01:01          |
| 1test06.cp | 100 | 221,08 | 48  | 0,79   | 01:23          |
| 1test07.cp | 100 | 259,01 | 56  | 0,78   | 01:48          |
| 1test08.cp | 100 | 295,66 | 64  | 0,79   | 02:20          |
| 1test09.cp | 100 | 332,31 | 72  | 0,79   | 03:10          |
| 1test10.cp | 100 | 369,30 | 80  | 0,79   | 04:01          |
| 1test11.cp | 100 | 405,26 | 88  | 0,79   | 04:39          |
| 1test12.cp | 100 | 445,3  | 96  | 0,78   | 05:24          |
| 1test13.cp | 100 | 479,68 | 104 | 0,79   | 05:27          |

Таблиця 3.4 – Круглий розкрий

| №          | $W$ | $L$    | $m$ | $\eta$ | $t$ (мін: сек) |
|------------|-----|--------|-----|--------|----------------|
| 0test01.cp | 60  | 27,30  | 5   | 0,63   | 00:02          |
| 0test02.cp | 60  | 45,13  | 10  | 0,77   | 00:06          |
| 0test03.cp | 60  | 66,39  | 15  | 0,78   | 00:11          |
| 0test04.cp | 60  | 89,10  | 20  | 0,78   | 00:17          |
| 0test05.cp | 60  | 111,10 | 25  | 0,78   | 00:24          |
| 0test06.cp | 60  | 132,81 | 30  | 0,78   | 00:31          |
| 0test07.cp | 60  | 156,01 | 35  | 0,78   | 00:49          |
| 0test08.cp | 60  | 177,86 | 40  | 0,78   | 00:56          |
| 0test09.cp | 60  | 199,78 | 45  | 0,78   | 01:12          |
| 0test10.cp | 60  | 222,64 | 50  | 0,78   | 01:26          |
| 0test11.cp | 60  | 245,74 | 55  | 0,77   | 01:35          |
| 0test12.cp | 60  | 267,13 | 60  | 0,78   | 01:45          |
| 0test13.cp | 60  | 290,54 | 65  | 0,77   | 01:53          |
| 0test14.cp | 60  | 313,19 | 70  | 0,77   | 02:08          |
| 0test15.cp | 60  | 337,36 | 75  | 0,77   | 02:34          |
| 0test16.cp | 60  | 357,99 | 80  | 0,77   | 03:13          |
| 0test17.cp | 60  | 380,88 | 85  | 0,77   | 02:59          |
| 0test18.cp | 60  | 404,45 | 90  | 0,77   | 03:18          |
| 0test19.cp | 60  | 425,42 | 95  | 0,77   | 03:37          |
| 0test20.cp | 60  | 449,24 | 100 | 0,77   | 04:01          |

Крім того, коефіцієнт  $\eta$  сильно залежить від асортименту та розмірності задачі, тому для задачі круглого розкрою розкид значень набагато ширший, ніж для прямокутного розкрою. Наприклад, частка корисної площі дуже висока в задачах, де поєднуються два такі розміри кіл, що перші легко заповнюють пустоти між другими. А якщо упаковувати кола великі відносно ширини смуги, то між ними утворюються дуже великі ділянки невикористаної площі, а коефіцієнт  $\eta$  виходить меншим, ніж для інших наборів заготовок.

*Оцінка складності алгоритму.* Оскільки задача розкрою NP-повна, то для будь-якого алгоритму, що реалізує її розв'язання, питання складності – найактуальніше. Для того, щоб на це запитання відповісти, проведена серія обчислювальних експериментів такого змісту. Для задач прямокутного і круглого розкрою різного асортименту збільшувалася кількість заготовок, що упаковуються з рівним кроком. Для кожного набору кілька разів запускався еволюційний процес, результат усереднювався. Для кожної задачі була побудована залежність часу роботи алгоритму від розмірності задачі. Первинний аналіз показав, що час роботи програми зростає повільніше, ніж квадрат розмірності задачі, але швидше, ніж розмірність (табл. 3.5-3.9, рис. 3.5-3.9).

Таблиця 3.5 – Прямокутний розкрій; поперек смуги укладається не більше 3-х предметів; всього 6 видів предметів

| $m$ | $t$ (сек) | $t/m$  | $t/m^2$ |
|-----|-----------|--------|---------|
| 6   | 2         | 0,3333 | 0,08333 |
| 12  | 5         | 0,4167 | 0,01667 |
| 18  | 9         | 0,5    | 0,00617 |
| 24  | 13        | 0,5417 | 0,00321 |
| 30  | 17        | 0,5667 | 0,00196 |
| 36  | 21        | 0,5833 | 0,00132 |
| 42  | 25        | 0,5952 | 0,00095 |
| 48  | 31        | 0,6458 | 0,00067 |
| 54  | 36        | 0,6667 | 0,00051 |
| 60  | 42        | 0,7    | 0,0004  |
| 66  | 48        | 0,7273 | 0,00032 |
| 72  | 55        | 0,7639 | 0,00025 |
| 78  | 62        | 0,7949 | 0,00021 |
| 84  | 68        | 0,8095 | 0,00018 |
| 90  | 74        | 0,8222 | 0,00015 |

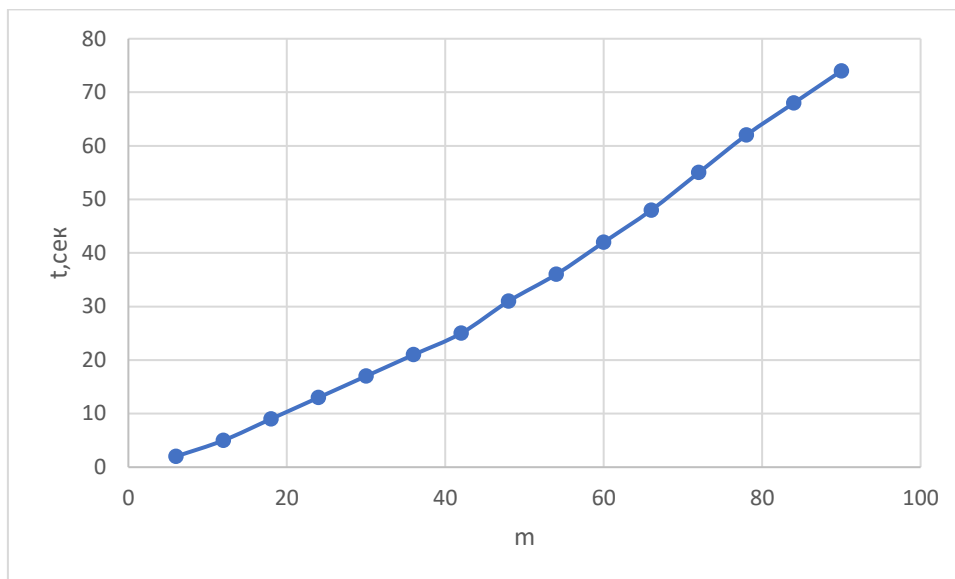


Рисунок 3.5 – Залежність часу роботи від розмірності завдання даних з таблиці 3.5

Таблиця 3.6 – Прямокутний розкрій; поперек смуги укладається 8 – 10 предметів; всього 10 видів предметів

| $m$ | $t$ (сек) | $t/m$  | $t/m^2$ |
|-----|-----------|--------|---------|
| 10  | 2         | 0,2    | 0,05    |
| 20  | 5         | 0,25   | 0,01    |
| 30  | 12        | 0,4    | 0,00278 |
| 40  | 17        | 0,425  | 0,00147 |
| 50  | 23        | 0,46   | 0,00087 |
| 60  | 28        | 0,4667 | 0,0006  |
| 70  | 37        | 0,5286 | 0,00039 |
| 80  | 45        | 0,5625 | 0,00028 |
| 90  | 55        | 0,6111 | 0,0002  |
| 100 | 65        | 0,65   | 0,00015 |

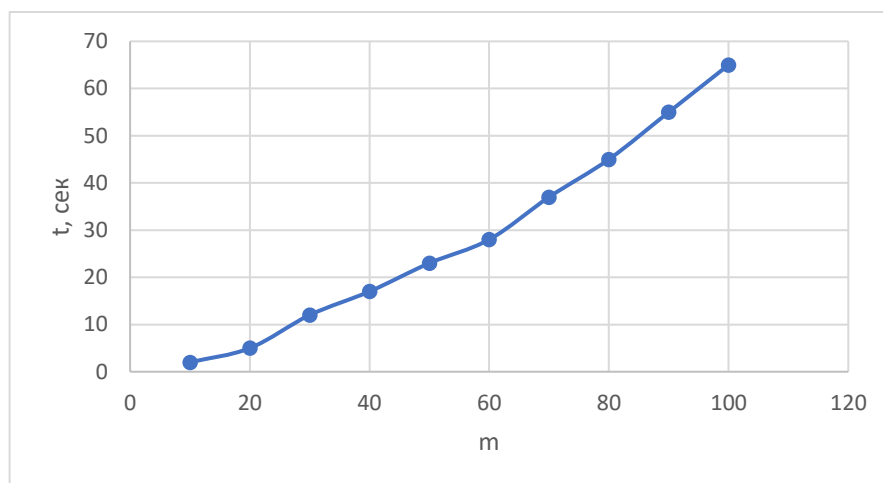


Рисунок 3.6 – Залежність часу роботи від розмірності завдання даних з таблиці 3.6

Таблиця 3.7 – Круглий розкрій; поперек смуги укладається не більше 3-х предметів; всього 5 видів предметів

| $m$ | $t$ (сек) | $t/m$  | $t/m^2$ |
|-----|-----------|--------|---------|
| 5   | 2         | 0,4    | 0,1     |
| 10  | 6         | 0,6    | 0,01667 |
| 15  | 11        | 0,7333 | 0,00606 |
| 20  | 17        | 0,85   | 0,00294 |
| 25  | 24        | 0,96   | 0,00167 |
| 30  | 32        | 1,0667 | 0,00104 |
| 35  | 41        | 1,1714 | 0,0007  |
| 40  | 50        | 1,25   | 0,0005  |
| 45  | 61        | 1,3556 | 0,00036 |
| 50  | 72        | 1,44   | 0,00028 |
| 55  | 85        | 1,5455 | 0,00021 |
| 60  | 98        | 1,6333 | 0,00017 |
| 65  | 116       | 1,7846 | 0,00013 |
| 70  | 130       | 1,8571 | 0,00011 |
| 75  | 147       | 1,96   | 0,00009 |
| 80  | 164       | 2,05   | 0,00008 |
| 85  | 182       | 2,1412 | 0,00006 |
| 90  | 200       | 2,2222 | 0,00006 |
| 95  | 221       | 2,3263 | 0,00005 |
| 100 | 240       | 2,4    | 0,00004 |



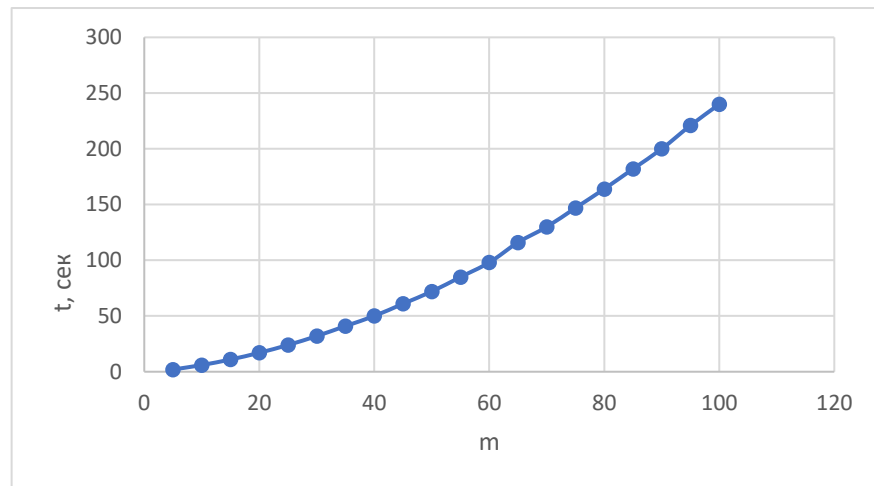


Рисунок 3.7 – Залежність часу роботи від розмірності завдання даних з таблиці 3.7

Таблиця 3.8 – Круглий розкрій; поперек смуги укладається не більше 6-и предметів; всього 8 видів предметів

| $m$ | $t$ (сек) | $t/m$  | $t/m^2$ |
|-----|-----------|--------|---------|
| 8   | 5         | 0,625  | 0,025   |
| 16  | 14        | 0,875  | 0,00446 |
| 24  | 27        | 1,125  | 0,00154 |
| 32  | 42        | 1,3125 | 0,00074 |
| 40  | 62        | 1,55   | 0,0004  |
| 48  | 84        | 1,75   | 0,00025 |
| 56  | 109       | 1,9464 | 0,00016 |
| 64  | 141       | 2,2031 | 0,00011 |
| 72  | 171       | 2,375  | 0,00008 |
| 80  | 207       | 2,5875 | 0,00006 |
| 88  | 244       | 2,7727 | 0,00005 |
| 96  | 288       | 3      | 0,00004 |
| 104 | 331       | 3,1827 | 0,00003 |

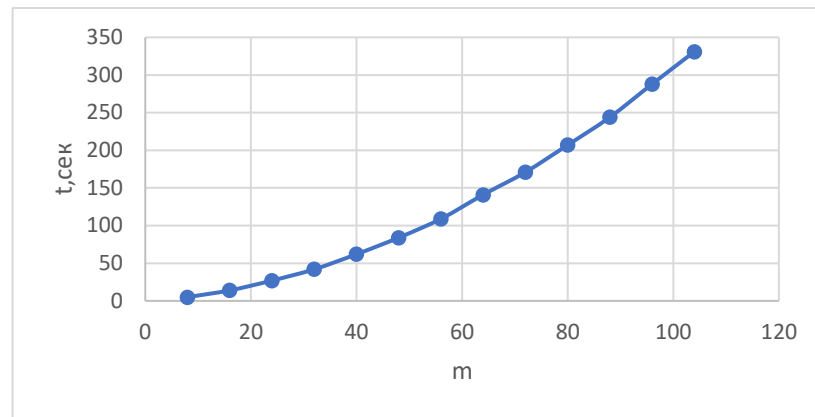


Рисунок 3.8 – Залежність часу роботи від розмірності завдання даних з таблиці 3.8

Для чотирьох наведених нижче різних серій тестів були отримані такі наближені залежності для круглого розкрою:

–  $t = O(m^{1,60})$  для даних з табл. 3.5;

–  $t = O(m^{1,65})$  для даних з табл. 3.6;

та для прямокутного розкрою:

–  $t = O(m^{1,33})$  для даних з табл. 3.7

–  $t = O(m^{1,48})$  для даних з табл. 3.8

Додатковий обчислювальний експеримент показав, що складність запропонованого алгоритму розв'язання задачі розкрою коливається залежно від складності конкретних задач, але ніколи не досягає значення  $O(m^2)$ . З огляду на те, що складності алгоритмів круглого і прямокутного розкрою коливаються в різних діапазонах, загальний висновок можна сформулювати так: NP-повна задача оптимізації плоского регулярного розкрою вирішена евристичним методом, який отримує прийнятне рішення за час, що зростає зі збільшенням розмірності задачі повільніше, ніж квадрат розмірності.

Для оцінки кількості поколінь, за яке алгоритм знаходить достатньо добре рішення, проведено обчислювальний експеримент: для декількох задач різної геометрії, різної складності, різної розмірності алгоритм запускався 10 разів для кожної, а результат усереднювався. Отримані дані дозволяють зробити висновок, що частка корисної площі максимально наближається до

свого граничного значення в середньому за  $n/2$  поколінь. Коефіцієнт  $\eta$  перестав змінюватися в середньому через  $m$  поколінь, де  $m$  – число заготовок, що упаковуються. На рис. 3.9–3.10 проілюстровано дві серії дослідів для прямокутного і круглого розкрою в задачах розмірності  $n = 100$ . Для добре обумовлених задач такої розмірності алгоритм знаходить прийнятне рішення швидше, за 10-20 поколінь.

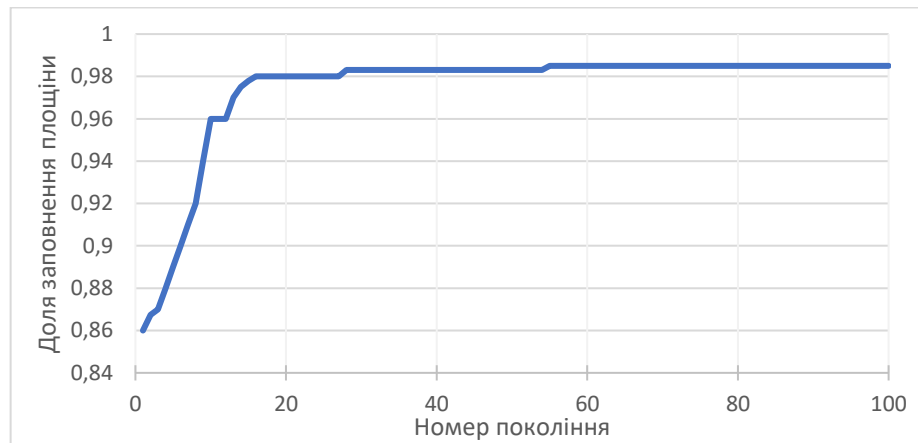


Рисунок 3.9 – Приклад роботи генетичного алгоритму для прямокутного розкрою

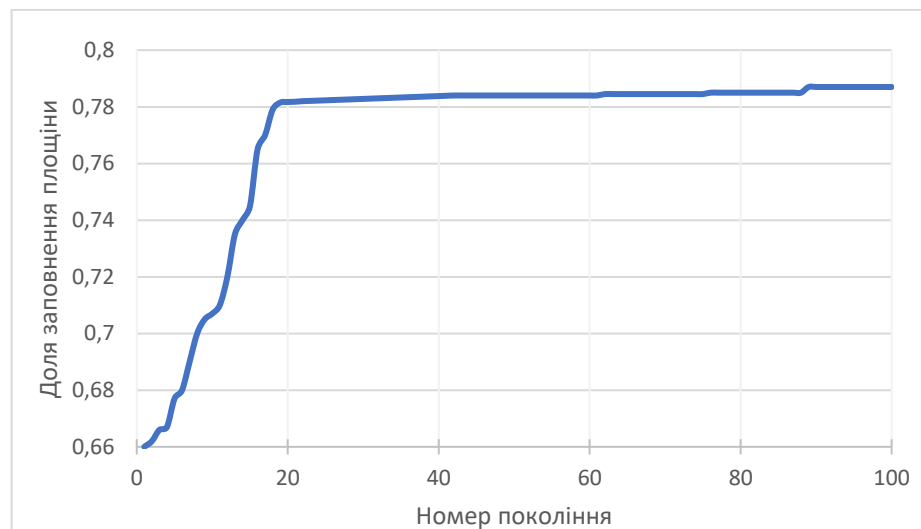


Рисунок 3.10 – Приклад роботи генетичного алгоритму для кругового розкрою

Оцінюючи розроблений генетичний алгоритм, слід пам'ятати, що цей алгоритм злитий з процедурою декодування, в яку вже закладено максимально можливе для неї поліпшення якості упаковки. У добре обумовлених задачах декодер, по суті справи, бере на себе половину роботи. Але в усіх інших випадках якість рішення більше залежить від порядку слідування заготовок, який оптимізується саме генетичним алгоритмом.

## ВИСНОВКИ

Задача оптимального розкрою матеріалів є надзвичайно актуальною для металургійної галузі. Адже від її ефективного вирішення безпосередньо залежать такі ключові техніко-економічні показники підприємств, як ресурсоемність, продуктивність, собівартість, рентабельність виробництва.

В умовах постійного зростання цін на металопрокат та загострення конкуренції на світовому ринку, питання раціонального розкрою листових та рулонних матеріалів набуває особливої гостроти для вітчизняних меткомбінатів. Адже оптимізація витрат металу на початковому етапі дозволяє істотно знизити собівартість готової продукції та підвищити її конкурентоспроможність.

Як показує аналіз, задача розкрою має глибоке теоретичне підґрунтя та численні прикладні застосування в матеріаломістких галузях промисловості. Широке коло міжнародних та вітчизняних публікацій присвячені розробці математичних моделей та обчислювальних методів для її вирішення.

Разом з тим, застосування класичних підходів у металургійній галузі ускладнюється низкою специфічних факторів. Зокрема, необхідно враховувати анізотропні властивості металопрокату, наявність дефектних зон, вимоги до групування за якісними ознаками тощо.

Тому існує нагальна потреба в розробці спеціалізованих моделей та методів розв'язання задачі розкрою саме для умов металургійного виробництва. Перспективним вбачається створення генетичних алгоритмів, для вирішення задачі оптимального розкрою матеріалів у металургійній промисловості.

Генетичні алгоритми є одним з найпоширеніших методів еволюційних обчислень, які ґрунтуються на моделюванні процесу біологічної еволюції для пошуку оптимальних або близьких до оптимальних рішень складних задач. Генетичні алгоритми можуть ефективно застосовуватися для оптимізації багато-параметричних функцій у великих просторах пошуку, якими є задачі

розкрою матеріалів. Показано, що генетичні алгоритми відрізняються від класичних методів оптимізації тим, що не вимагають гладкості та диференційованості цільової функції, а отже, придатні для вирішення задач розкрою матеріалів, які часто є багато-екстремальними та не мають аналітичних моделей. Ще однією перевагою генетичних алгоритмів є можливість їх паралельної реалізації за рахунок того, що кожну особу популяції можна оцінювати незалежно. Це сприяє значному прискоренню процесу пошуку рішення.

Таким чином, генетичні алгоритми є універсальним, гнучким та ефективним методом для розв'язання задачі розкрою матеріалів у металургійній промисловості. Вони дозволяють швидко знаходити близькі до оптимальних розв'язки у великих просторах пошуку за рахунок моделювання процесів природної еволюції.

У роботі розроблено генетичний алгоритм для вирішення задачі плоского розкрою стрічки на предмети нескладної геометричної форми. Побудований генетичний алгоритм працює з перестановками абстрактних предметів і не залежить від їхньої геометричної форми параметром такого алгоритму є декодер, що трансформує перестановку в карту розкрою об'єктів. Запропонований підхід дозволяє вирішувати як задачі прямокутного, так і круглого розкрою.

Також розроблено програмне забезпечення на основі цього алгоритму, що дає можливість користувачеві задавати вхідні дані, візуалізувати результати розкрою у вигляді карти та аналізувати показники якості отриманого розв'язку. Програма продемонструвала здатність ефективно працювати з задачами різної складності та розмірності.

Проведено дослідження ефективності запропонованого методу на основі серії обчислювальних експериментів. Встановлено, що для задач середньої складності генетичний алгоритм знаходить близьке до оптимального розв'язання за кількість поколінь еволюції, що не перевищує кількості предметів у задачі.

Показано, що час роботи алгоритму зростає повільніше, ніж квадрат розмірності задачі. Це доводить, що розроблений метод належить до класу поліноміальних евристичних алгоритмів. Він дозволяє отримувати наближені розв'язки задачі оптимального розкрою за прийнятний час.

Отже, проведені дослідження підтвердили ефективність застосування генетичного алгоритму та розробленого на його основі програмного забезпечення для вирішення практичних задач раціонального розкрою матеріалів у металургійній галузі.

## ПЕРЕЛІК ДЖЕРЕЛ ТА ПОСИЛАННЯ

1. Абдурахманов Ш. Способи аксонометричного зображення точок об'єктів багатовимірних просторів. *Прикладна геометрія та інженерна графіка*. Вип. 81. Київ: КНУБА, 2009. С. 130–133.
2. Асанов М.О., Баранський В.А., Расін В.В. Дискретна математика: графи, матроїди, алгоритми. Київ: «Регулярна і хаотична динаміка», 2011. 288 с.
3. Балан С.О. Інтелектуальні інформаційні технології в машинобудуванні. Одеса: Астропринт, 2002. 360 с.
4. Балицька Т.Ю., Нефьодов Ю.М. Методи он і імітації в прикладах і задачах: Навчальний посібник. Київ: Кондор, 2011. 324 с.
5. Бідюк П.І., Корбутяк М.В. Моделювання процесу розкрою листового прокату. Вісник НТУ «ХП». Серія: *Математичне моделювання в техніці та технологіях*. 2015. № 60 (1169). С. 3–8.
6. Бідюк П.І., Ткачук М.М., Стадник Б.І. Моделювання розкрою листового прокату в середовищі MATLAB. *Системні технології*. 2015. № 1 (98). С. 75–82.
7. Білик О.М. Методика моделювання раціональних схем розкрою шкіри. *Вісник КНУТД*. 2010. № 1. С. 78–84.
8. Бондаренко М.Ф., Білоус Н.В., Руткас А.Г. Комп'ютерна дискретна математика. Харків: «Компанія СМІТ», 2004. 480 с.
9. Богдан Ю.М., Кушнір Р.М. Проектування раціонального використання шкіри. *Вісник ХНУ*. 2002. № 573. С. 70–73.
10. Богун Т.В. Методи оцінювання ефективності використання шкіри на етапах технологічної підготовки виробництва. *Вісник КНУТД*. 2019. № 1. С. 95–104.
11. Богун Т.В., Горбенко Н.І., Гамола О.Є. Оптимізація процесу розкрою шкіряної сировини. *Вісник КНУТД*. 2017. № 1 (105). С. 41–47.
12. Ванін В.В., Вірченко Г.А., Ванін І.В. Деякі питання розробки обчислювальних алгоритмів структурно-параметричного моделювання складних



геометричних об'єктів. *Прикладна геометрія та інженерна графіка*. Київ: КНУБА, 2006. Вип. 76. С. 17–23.

13. Васильков П.В. Визначення раціональних параметрів лекал жіночих спідниць. *Вісник ХНТУСГ*. Технічні науки. 2021. № 205. С. 153–163.

14. Вітлінський В.В., Наконечний С.І., Терещенко Т.О. Математичне програмування: навч.-метод. посіб. для сам. вивчення дисципліни. Київ: КНЕУ, 2001. 248 с.

15. Герасимчук О.А. Дослідження продуктивності алгоритмів моделювання при вирішенні задач раціонального розкрою. *Вісник КНУТД*. 2013. № 3. С. 115–122.

16. Герасимчук О.А., Коваленко Л.М. Дослідження методів моделювання при вирішенні задач раціонального розкрою текстильних матеріалів. *Вісник КНУТД*. 2010. № 5. С. 98–104.

17. Герман Н.В. Побудова моделей раціонального розміщення лекал на шкіряній тканині. *Вісник ХНУ*. 2002. № 573. С. 218–221.

18. Горбатюк Р.М., Піпа Б.Ф. Метод розв'язання задачі розкрою з урахуванням анізотропних властивостей матеріалів. *Науковий вісник НЛТУ України*. 2009. Вип. 19.9. С. 167–172.

19. Григорак М.Ю. Моделювання задачі розміщення прямокутників на основі еволюційного алгоритму. *Наукові записки*. 2006. Т. 59. С. 69–75.

20. Григорак М.Ю. Методи вирішення задачі про розміщення прямокутників. *Науковий вісник Волинського національного університету ім. Лесі Українки*. 2010. № 5. С. 62–68.

21. Григорак М.Ю., Мікульонок І.О. Моделі та методи вирішення задачі оптимального розміщення лекал. *Наукові нотатки*. 2013. Вип. 43. С. 58–65.

22. Григоренко Г.М., Зінь М.М., Ковалевська О.А. Математичне моделювання процесів обробки металів тиском. *Вісник Приазовського державного технічного університету*. 2017. № 34. С. 158–165.

23. Грицишин Я.М., Корпильов Д.В., Кривий Р.З., Свірідова Т.В., Ткаченко С.П. Генетичні алгоритми для розв'язання задач розміщення. *Вісник НУ*

«Львівська політехніка». *Комп'ютерні науки та інформаційні технології*. Львів: Видавництво НУ «Львівська політехніка», 2009. № 638. С. 271–276.

24. Давиденко І.В. Методи автоматизованого проектування конструкцій швейних виробів. *Вісник ХНТУСГ*. Технічні науки. 2021. № 211. С. 109–121.

25. Давиденко І.В. Модифікація методу послідовного аналізу варіантів розкрою для виробів складних форм. *Вісник ХНТУСГ*. Технічні науки. 2022. № 216. С. 137–145.

26. Дубіна А. В. Алгоритми локального пошуку та табу-пошуку для задач розміщення прямокутників забезпечення технологічного прогресу ХХІ сторіччя: матеріали Міжнародної наукової конференції (Т. 2) (м. Чернівці, 1 травня, 2020 р.). Чернівці: МЦНД, 2020. С. 48–53.

27. Згуровський М.З., Панкратова Н.Д. Основи системного аналізу. Київ: Видавнича група ВНУ. 2007. 544 с.

28. Ієвлева С.М. Основи системного аналізу: навч.-метод. комплекс. Харків: ХНУРЕ, 2017. 240 с.

29. Іващук О.Т. Економіко-математичне моделювання: Навчальний посібник. Тернопіль: ТНЕУ «Економічна думка», 2008. 704 с.

30. Квіта Г.М., Шіковець К.О., Безсмертна Ю.С. Моделювання виробничих процесів автоматизованого виробництва. *Економіка та управління підприємствами*. 2018. № 15. С. 316–321.

31. Кібець А.П., Хом'як О.М. Процес формоутворення одягу як задача оптимального розміщення плоских фігур. *Вісник ХНТУСГ*. Технічні науки. 2019. № 203. С. 86–96.

32. Колиско О.З. Модифікація генетичного алгоритму для генерації секцій розкрійних схем. *Вісник КНУТД*. 2009. № 1. С. 14–17.

33. Колиско О.З. Застосування еволюційного моделювання при побудові раціональних розкрійних схем на рулонних матеріалах. *Вісник КНУТД*. 2010. № 5. С. 62–65.

34. Колиско О.З. Структура та зміст процедур генетичного алгоритму для складання розкрійних схем. *Вісник Київського національного університету технологій та дизайну*. 2014. № 4 (78). С. 227–233.
35. Косолап А.І., Кодола Г.М. Ефективний метод оптимізації в задачах лінійного розкрою матеріалів. *Математичне моделювання*. 2018. № 1. С. 12–21.
36. Косолап, А.І., Кодола Г.М. Оптимізація в задачах лінійного розкрою матеріалів *Вісник Національного технічного університету «ХПІ»*. 2016. № 44 (1216) С. 56–66.
37. Кодола Г.М., Рогоза Б.Є. Побудова технологічної матриці варіантів лінійного розкрою. *Вісник Національного технічного університету «ХПІ»* 2017. – № 23(1245). Харків, НТУ «ХПІ», 2017. С. 111–116
38. Кривий Р.З., Лобур М.М., Ткаченко С.П. Застосування генетичного алгоритму прямокутного розміщення для гільйотинного розкрою. *Вісник Національного університету «Львівська політехніка». Комп'ютерні системи проектування. Теорія і практика*. 2010. № 685. С. 138–142.
39. Кузьмініх В.О., Молодід О.К., Тараненко Р.А. Методи дослідження операцій: навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки». Київ: КПІ ім. Ігоря Сікорського, 2020. 117 с.
40. Кузьмін О.Є. Системний аналіз і прийняття інноваційних рішень: Навч. посіб. Львів: Видавництво «Новий Світ -2000». 2021. 226 с.
41. Кутянська В.І. Розв'язання однієї задачі розкрою алгоритмом оптимізації бджолою колонією. *Науковий вісник НЛТУ України*. 2010. № 8. С. 290–294.
42. Лавров Є.А., Перхун Л.П., Шендрик В.В. Математичні методи дослідження операцій: підруч. Суми: Сумський державний університет, 2017. 212 с.
43. Ладанюк А.П. Основи системного аналізу: навчальний посібник. Вінниця: Нова книга, 2004. 176 с.

44. Ліпницький А.А. Застосування генетичних алгоритмів до задачі про розміщення прямокутників. *Кібернетика та системний аналіз*. 2002. № 6. С. 180–184.
45. Михайленко В.Є. Сучасний стан методів геометричного та комп'ютерного моделювання та напрямки їх розвитку. *Прикладна геометрія та інженерна графіка*. Київ. 2001. Вип. 68. С. 3–6.
46. Новожилова М.В., Астахов Є.А Комп'ютерне моделювання умови неперетину в задачах розміщення в анізотропній області. *Системи обробки інформації*. 2003. № 6. С. 138–143.
47. Панченко С.В., Юхимчук В.Д. Планування та моделювання раціонального розкрою шкіри. *Вісник КНУТД*. 2012. № 5 (65). С. 151–157.
48. Підлазова А.В. Генетичні алгоритми: походження, сутність, застосування. *Економіка, інформаційні технології та управління в металургії: збірник наукових праць*. Київ: Вид-во НТУУ «КПІ» «Освіта України», 2004. С. 122–128.
49. Прокопович І.В., Сенько В.І. Математичне моделювання задачі розкрою прокату. *Вісник Житомирського державного технологічного університету. Серія: Технічні науки*. 2017. № 1 (77). С. 130–135.
50. Романова Т.Є. Засоби побудови математичних моделей оптимізаційних задач розміщення геометричних об'єктів та їх застосування: автореф. дис. ... докт. техн. наук: 01.05.02 / Ін-т кібернетики ім. В.М. Глушкова НАН України. Київ: ПП Овчаренка О.Г., 2003. 36 с.
51. Семак Б.Д. Моделі і методи прийняття проектних рішень в задачах раціонального використання текстильних матеріалів. *Вісник ХНУ*. 2002. № 573. С. 215–217.
52. Сергієнко І.В. Математичні моделі і методи розв'язання задач дискретної оптимізації. Київ: Наукова думка, 1985. 384 с.
53. Сергієнко І.В., Лебедева Т.Т., Роцин В.А. Наближені методи розв'язання дискретних задач оптимізації. Київ: Наукова думка, 1980. 276 с.

54. Соболь О.М., Собина В.О., Тур О.М. Побудова  $\omega$ -функцій в задачах покриття заданої області геометричними об'єктами зі змінними метричними характеристиками *Прикладна геометрія та інженерна графіка. Міжвідомчий науково-технічний збірник*. Київ: КНУБА, 2010. Вип. 86. С. 118–122.
55. Стецюра Г.Г. Еволюційні методи в задачах управління, вибору, оптимізації. *Прилади і системи управління*. 1998. № 3. С. 54–62.
56. Стоян Ю.Г., Шайтхауер Г., Яськов Г.М. Пакування нерівних куль у різні контейнери. *Кібернетика і системний аналіз*. 2016, № 3 (52). С. 97–105.
57. Стоян Ю.Г., Гіль М.І. Методи і алгоритми розміщення геометричних об'єктів. Київ: Наукова думка, 2016. 247 с.
58. Сьомка С.В., Куц Ю.В., Харчук В.Є., Сердюк Г.Б. Раціональний розкрій сталевого прокату. *Металургійна та гірничорудна промисловість*. 2001. № 1. С. 41–43.
59. Ткачук М.М., Корбутяк М.В. Застосування математичного і комп'ютерного моделювання в задачах оптимізації розкрою матеріалів. *Наукові праці ВНТУ*. 2010. № 2. С. 1–8.
60. Чорнобай К.Ю., Сєдих О.Л. Математична модель та алгоритм рішення задачі про оптимальний лінійний розкрій. *Інформаційні технології в соціокультурній сфері, освіті та економіці*: матеріали Міжнародної науково-практичної конференції (м. Київ, 18–19 квітня 2019 р.). Київ: Видавничий центр КНУКіМ, 2019. С. 185–188.
61. Чупринка В.І., Волошин О.Т., Піпа Т.А. Підготовка інформації для автоматичного розкрою. Київ: Вісник ДАЛПУ. 2000, № 1. С. 91–93.
62. Чуб І.А., Новожилова М.В., Андронов В.А. Моделювання прикладних оптимізаційних задач розміщення об'єктів з метричними характеристиками, що змінюються: монографія. Харків: НУЦЗ України, 2017. 167 с.
63. Щербань В.Ю., Хомик Н.І. Модель і метод вирішення задачі розкрою матеріалу. *Науковий вісник НЛТУ України*. 2010. Вип. 20.5. С. 167–173.

64. Щербань В.Ю., Колиско О.З., Мельник, Г.В., Шолудько М.І., Калашник В.Ю. Комп'ютерне проектування систем: програмні та алгоритмічні компоненти. Київ: Освіта України. 2019. 902 с.
65. Alvarez-Valdes R., Parajon A., Tamarit J.M. Reactive GRASP for the strip-packing problem. *Computers & Operations Research*. 2008. No. 35 (4). Pp. 1065–1083.
66. Bennell J.A., Oliveira J.F., Song X. A tutorial in irregular shape packing problems. *Journal of the Operational Research Society*, 2010 No. 61 (10). Pp. 1369–1379.
67. Burke E., Kendall G., Whitwell G. A simulated annealing enhancement of the best-fit heuristic for the orthogonal stock cutting problem. *INFORMS journal on Computing*. 2004. No. 16 (3). Pp. 345–355.
68. Caggiano A., Seccia A., Di Sciascio F. A mixed integer linear programming model for optimising stock cutting in steel mills. *International Journal of Production Research*. 2020. No. 58 (16). Pp. 5063–5078.
69. Dyckoff H. Topology of cutting and packing problems. *European Journal of Operational Research*. 1990. No. 44. Pp. 145–159.
70. Fuentes A. Solution to travelling salesman problem by clusters and a modified multi-restart iterated local search metaheuristic PloSone. 2018. No 13. Pp. 63–84.
71. Goldberg D.E. Genetic Algorithms in Search. Optimization and Machine Learning. 1989. 412 p.
72. Gonçalves J. F. A hybrid genetic algorithm-heuristic for a two-dimensional orthogonal packing problem. *European Journal of Operational Research*. 2007. Vol. 183 (3) P. 1212–1229.
73. Heckmann R., Lengauer T. A simulated annealing approach to the nesting problem in the textile manufacturing industry. *Annals of Operations Research*. 1995. No. 57. Pp. 103–133.
74. Hinxman A.I. The trim loss and assortment problems. *European Journal of Operational Research*. 1980. No. 5. Pp. 8–18.

75. Hopper H., Turton B. A review of the application of meta-heuristic algorithms to 2D strip packing problems. *Artificial Intelligence Review*. 2001. No. 16. Pp. 257–285.
76. Jakobs S. On genetic algorithms for the packing of polygons. *European journal of operational research*. 1996. No. 88 (1). Pp. 165–181.
77. Karaboga D. A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review*. 2014. No. 42 (1). Pp. 21–57.
78. Kierkosz I., Luczak M. A hybrid evolutionary algorithm for the twodimensional packing problem. *Central European Journal of Operations Research*. Berlin: Springer, 2014. Vol. 22. P. 729–753.
79. Martello S., Toth P. Knapsack problems: Algorithms and Computer Implementation. *Operation Research*. 1990. No. 19. Pp. 983–998.
80. Oliveira J.F., Gomes A.M., Ferreira J.S. Enhanced pattern matching algorithms for wood veneer layout. *Expert Systems with Applications*. 2014. No. 41 (2), Pp. 681–689.
81. Wang J., Liu J., Zhong Y.F., Wang G.G. Intelligent roll-shifting optimization for hot strip mills. *Journal of Materials Processing Technology*. 2017. No. 247. Pp. 39–53.
82. Zadeh L.A. Soft computing and Fuzzy Logic. *Software Engineering Journal*. 1994. No. 4. Pp. 21–25.
83. Zhang S., Liu Q., Lai X., Cheng H. Strip layout planning model of hot rolling production based on flexible constraints. *Journal of Manufacturing Systems*. 2015. No. 37. Pp. 320–332.

## ДОДАТКИ

## Додаток А. Приклад задання пакета замовлень

|                |             |               |             |
|----------------|-------------|---------------|-------------|
| 05test05.rp    | 23test23.rp | 30test30.rp . | 51test51.rp |
| 10             | 20          | 20            | 20          |
| 1 7 3          | 3 2 20      | 1 2 12        | 4 3 2       |
| 1 3 4          | 4 2 10      | 1 2 5         | 1 5 2       |
| 1 4 4          | 2 2 15      | 1 6 6         | 1 4 3       |
| 1 2 3          | 2 2 5       | 1 2 8         | 1 4 4       |
| 1 3 5          | 3 4 7       | 1 6 4         | 1 2 6       |
|                | 2 4 8       | 1 8 1         | 1 5 2       |
| 10test10.rp    | 2 2 12      | 1 4 6         | 2 2 2       |
| 10             | 2 2 8       | 1 6 2         | 1 5 2       |
| 1 3 5          | 1 4 5       | 1 3 2         | 1 7 3       |
| 1 3 4          | 1 4 6       | 1 4 8         | 1 2 2       |
| 1 4 4          | 1 4 12      | 1 1 8         | 1 3 3       |
| 1 4 3          |             | 2 4 3         | 1 2 4       |
| 1 2 3          | 24test24.rp | 1 8 4         | 1 3 5       |
| 1 2 6          | 20          | 1 14 6        | 1 4 3       |
| 1 3 4          | 1 3 8       | 1 2 4         | 1 4 2       |
| 1 4 3          | 1 3 7       | 1 6 3         | 1 2 4       |
| 1 7 3          | 1 16 2      | 1 7 1         | 1 5 3       |
| 1 3 4          | 1 3 6       | 1 3 6         | 1 3 3       |
|                | 1 2 23      | 1 4 7         | 1 5 4       |
| 10test20.rp    | 1 3 2       | 1 6 4         | 1 4 4       |
| 10             | 1 18 2      | 1 2 5         | 1 3 4       |
| 2 3 5          | 1 3 8       | 1 8 5         | 1 4 2       |
| 2 3 4          | 1 2 19      | 1 4 6         | 1 4 4       |
| 2 4 4          | 1 6 5       | 1 6 4         | 1 2 7       |
| 2 4 3          | 1 14 2      | 1 1 8         | 1 5 3       |
| 2 2 3          | 1 2 10      | 1 6 3         | 1 5 3       |
| 2 2 6          | 1 11 2      | 1 11 3        | 1 4 4       |
| 2 3 4          | 1 2 21      | 2 1 2         | 1 2 3       |
| 2 4 3          | 1 11 3      |               | 1 2 4       |
| 2 7 3          | 1 2 10      | 37test37.r p  | 1 3 6       |
| 2 3 4          | 1 3 7       | 20            | 1 3 2       |
|                | 1 2 14      | 3 6 4         | 1 8 2       |
| 10test50.txt . | 1 3 5       | 7 5 3         | 1 2 7       |
| 10 65          | 1 6 2       | 2 8 3         | 1 5 2       |
| 5 3 5          | 1 8 3       | 8 3 6         | 1 3 7       |
| 5 3 4          | 1 7 2       | 2 5 5         | 1 5 2       |
| 5 4 4          | 1 11 2      | 2 8 2         | 1 8 3       |
| 5 4 3          | 1 2 12      | 2 7 2         | 1 2 6       |
| 5 2 3          |             | 3 3 3         | 1 3 3       |
| 5 2 6          |             | 1 4 5         | 2 5 3       |
| 5 3 4          |             | 3 5 2         | 1 4 3       |
| 5 4 3          |             | 1 3 4         | 1 3 5       |
| 5 7 3          |             | 2 2 6         | 1 2 4       |
| 5 3 4          |             | 1 2 4         |             |



**Декларація академічної доброчесності  
здобувача вищої освіти ЗНУ**

Я Семілетов Деніс Володимирович, студент другого курсу, форми навчання денна, факультету економічного, спеціальності 051 – Економіка, адреса електронної пошти semeletov@gmail.com,  
- підтверджую, що написана мною кваліфікаційна робота на тему «Моделювання раціонального розкрою матеріалів у металургійній промисловості» відповідає вимогам академічної доброчесності та не містить порушень, що визначені у ст. 42 Закону України «Про освіту», зі змістом яких ознайомлений/ознайомлена;  
- заявляю, що надана мною для перевірки електронна версія роботи є ідентичною її друкованій версії;  
- згоден/згодна на перевірку моєї роботи на відповідність критеріям академічної доброчесності у будь-який спосіб, у тому числі за допомогою Інтернет-системи, а також на архівування роботи в базі даних цієї системи.

Дата \_\_\_\_\_ Підпис \_\_\_\_\_ ПІБ (студент) Семілетов Д. В.

Дата \_\_\_\_\_ Підпис \_\_\_\_\_ ПІБ (наук. керівника) Козін І.В.