НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ім. ІГОРЯ СІКОРСЬКОГО»
«ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ»
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

Алгоритмізація та порівняльний аналіз двовимірної упаковки

Виконав: студент группи КА-41 Мнівець О.О.

Науковий керівник: доц. Подколзін Г.Б.

Київ 2018

Постановка задачі

У задачах двовимірної прямокутної упаковки потрібно без перетинів розташувати на площині скінченну множину різних предметів прямокутної форми так, щоб значення деякої цільової функції досягало мінімуму. В якості цільової функції можуть розглядатися площа оточуючого упаковку прямокутника, довжина займаної предметами смуги, і т. д.

Математична постановка задачі

I = {1, 2,. . . , N} - множина прямокутників; w_i і h_i – ширина і висота i-го прямокутника. x_i і y_i – координати лівого нижнього кута i-го прямокутника; W і H – ширина і висота оточуючого прямокутника; змінна $l_{ij} \in \{0, 1\}$ дорівнює одиниці, якщо i-й прямокутник знаходиться лівіше j-го, і нулю в іншому випадку; змінна $b_{ij} \in \{0, 1\}$ дорівнює одиниці, якщо i-й прямокутник знаходиться нижче j-го, і нулю в іншому випадку.

знайти minWH при обмеженнях:

$$
\begin{aligned}
&x_i \ge 0, && i \in I,\\
&y_i \ge 0, && i \in I,\\
&x_i + w_i \le W, && i \in I,\\
&y_i + h_i \le H, && i \in I.
\end{aligned}
$$

Математична постановка задачі

Для того щоб два прямокутника не перетинались, досить щоб один з них був лівіше, правіше, нижче, або вище іншого.

$$x_i + w_i \le x_j + (1 - l_{ij})\sum_{k=1}^{n} w_k, \qquad i, j \in I,$$

$$y_i + h_i \le y_j + (1 - b_{ij})\sum_{k=1}^{n} h_k, \qquad i, j \in I,$$

$$l_{ij} + l_{ji} + b_{ij} + b_{ji} = 1, \qquad i, j \in I.$$

Евристики

Мають низький час обчислень, але невелику точність.

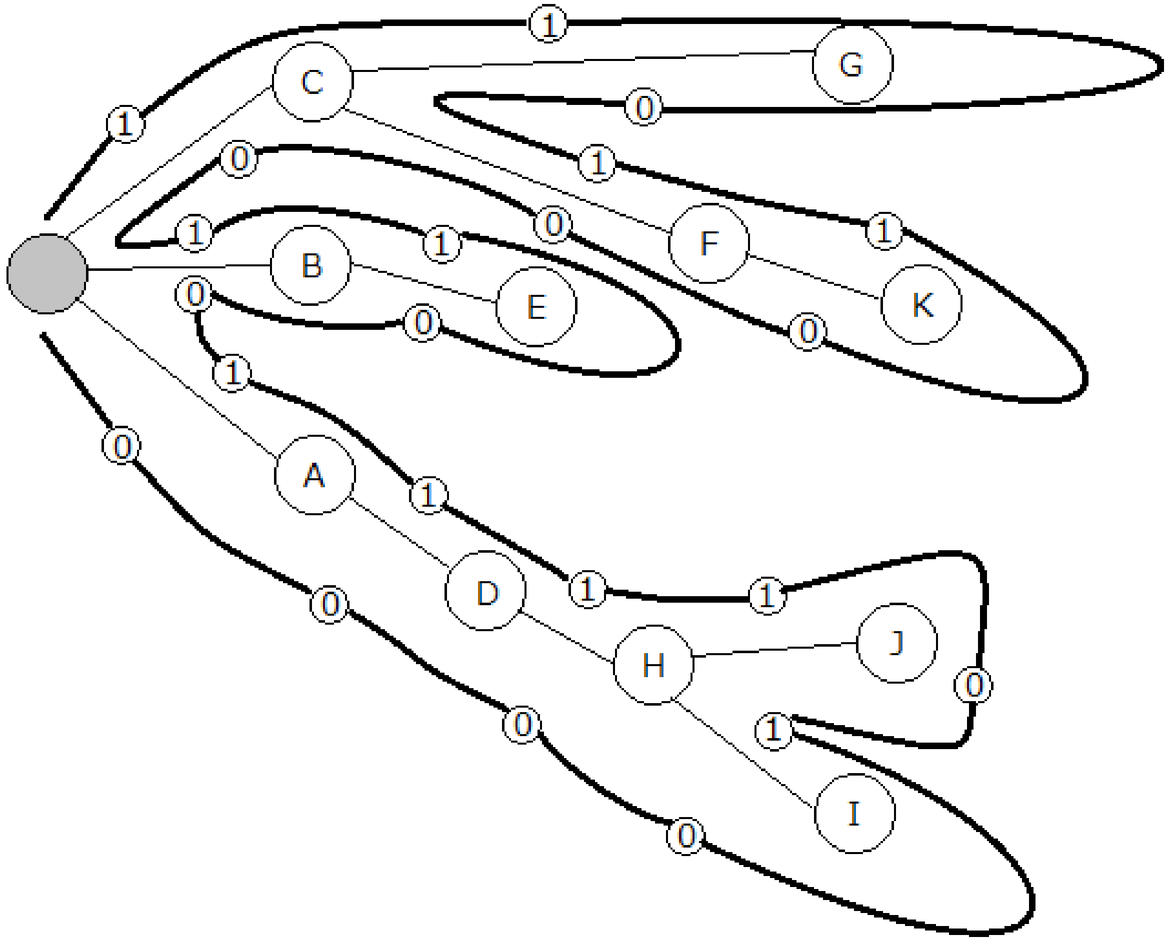
Евристичні алгоритми, як правило, поділяються на конструктивні евристики та евристики покращення.

- Конструктивні евристики будують розв'язок шляхом додавання елементів один за одним.
- Евристики покращення починають з вихідного розв'язку, тобто повного розв'язку. Потім цей початковий розв'язок покращується шляхом застосуванням невеликих послідовних змін, доки не буде виконаний критерій зупинки.

Алгоритми локального пошуку

Мають високу точність та велику складність обчислення. Усі методи локального пошуку починають з вже існуючого розв'язку і намагаються його покращити. В роботі ці алгоритми були реалізовані за допомогою схеми «Орієнтовані дерева»

Кодування орієнтованих дерев

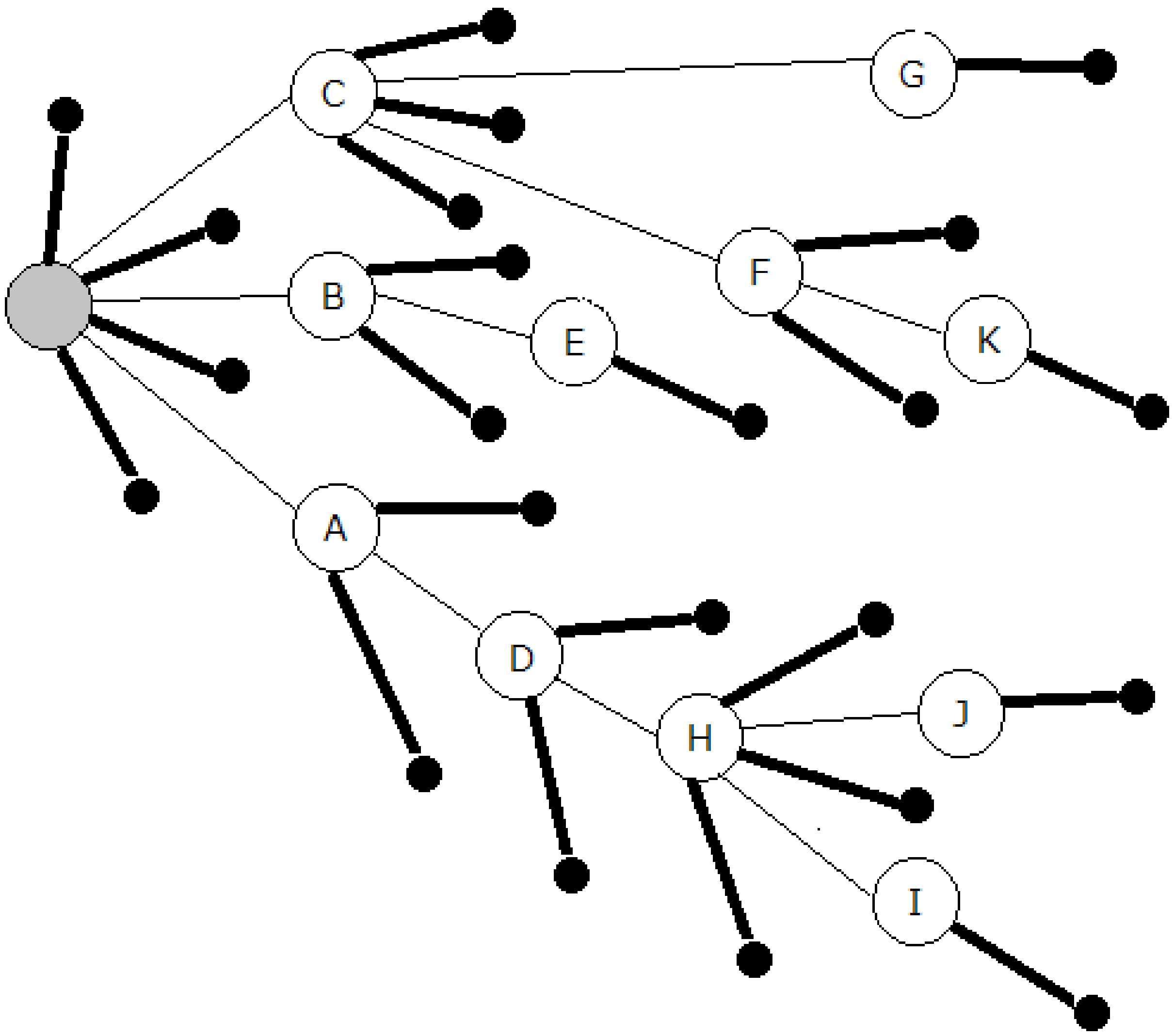


Окіл

Околом поточного розв’язку називається множина всіляких розв'язків, отриманих шляхом одноразового застосування деякої операції до поточного розв’язку. Розглянемо наступні операції над орієнтованими деревами:

- (O1) Перестановка двох вершин дерева і відповідних їм прямокутників. При цьому структура дерева не змінюється.
- (O2) Переміщення листа дерева і відповідного йому прямокутника, в нову позицію листа.

Окіл



Алгоритм локального спуску

АЛГОРИТМ ЛОКАЛЬНОГО СПУСКУ

1. Побудувати початковий розв'язок S.

2. Повторювати, поки в околі N(S) є нерозглянуті розв'язки.

   2.1. Вибрати нерозглянутий розв'язок S' з околу N(S).

   2.2. Якщо F(S') < F(S), то S: = S'.

3. Видати результат S.

Отриманий в результаті виконання алгоритму розв'язок S буде локально-оптимальним.

Алгоритм імітації випалювання

1. Побудувати початковий розв'язок S.

2. Вибрати початкову температуру T> 0.

3. Повторювати, поки не виконаний критерій зупинки.

   3.1. Виконати цикл L раз.

      3.1.1. Вибрати випадковим чином розв'язок S' з околу N(S).

      3.1.2. Обчислити Δ = F(S') - F(S).

      3.1.3. Якщо Δ ≤ 0, то покласти S: = S'.

      3.1.4. Якщо Δ > 0, то покласти S: = S' з ймовірністю e^{-T}.

   3.2. Знизити температуру T = rT.

4. Видати кращий знайдений розв'язок.

Розширення класу задач

I = {1, 2,. . . , N} - множина предметів; w_{i1} і h_{i1} – ширина і висота першого прямокутника і-го предмета, w_{i2} і h_{i2} – ширина і висота другого прямокутника і-го предмета.

знайти minWH при обмеженнях:

$x_i \geq 0$, i ∈ I,

$y_i \geq 0$, i ∈ I,

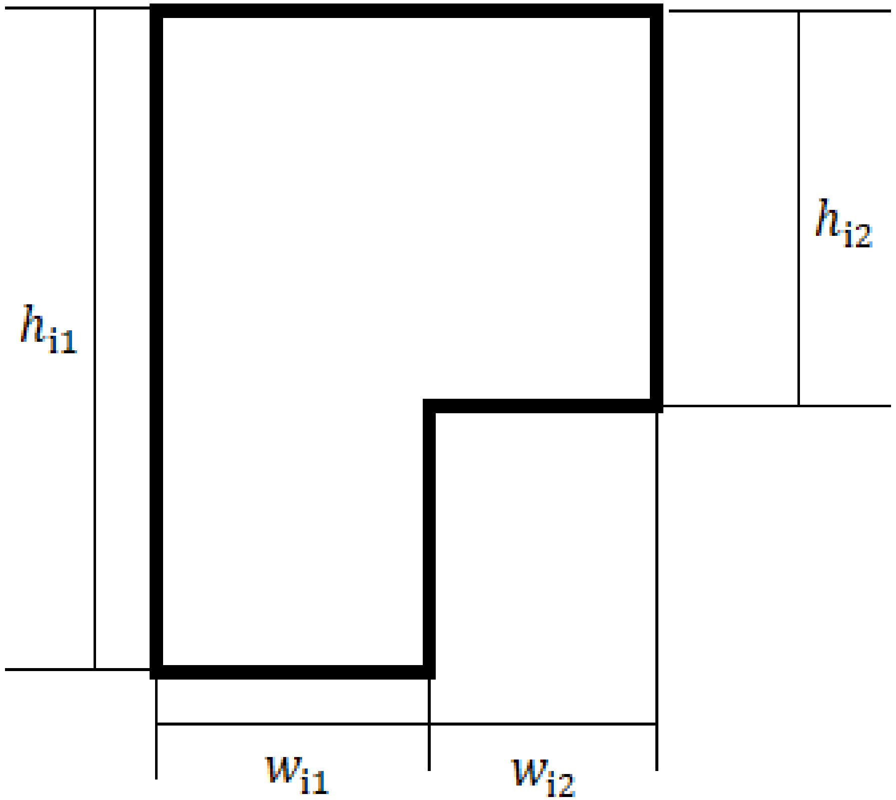
$x_i + w_{i1} + w_{i2} \leq W$, i ∈ I,

$y_i + h_{i1} \leq H$, i ∈ I.

$x_i + w_{i1} + w_{i2} \leq x_j + (1 - l_{ij})\sum_{k=1}^{n}(w_{k1} + w_{k2})$, i, j ∈ I,

$y_i + h_{i1} \leq y_j + (1 - b_{ij})\sum_{k=1}^{n} h_{k1}$, i, j ∈ I,

$l_{ij} + l_{ji} + b_{ij} + b_{ji} = 1$, i, j ∈ I.



Результати

Метод	Пакування прямокутників	Пакування складних предметів
Локальний пошук	13,44 сек, 94,7%	18,44 сек, 93,1%
Імітація випалювання	24,2 сек, 96,6%	27,21 сек, 95,4%
Гібридний алгоритм	38,8 сек, 98,1%	40,82 сек, 97,7%
Рівневий алгоритм	1,89 сек, 57,3%	4,33 сек, 77,6%
«найбільш доречний(BF)»	3,21 сек, 78,1%	6,71 сек, 83,6%

Висновки

- Розглянуто задачу упаковки предметів в контейнер. Описані деякі евристичні алгоритми та алгоритми локального пошуку. Розроблено модифікації існуючих методів для нового класу предметів. Зроблено аналіз особливостей запропонованих алгоритмів. Приведено опис їх реалізації, досліджено їх можливості на тестовому прикладі. В результаті бачимо високу ефективність методів локального пошуку та високу шкидкість евристичних алгоритмів.

Дякую за увагу!