ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1998
Seria: AUTOMATYKA z. 125 Nr kol. 1391

Igor KIERKOSZ
Politechnika Koszalińska

ZASTOSOWANIE ALGORYTMÓW GENETYCZNYCH DO OPTYMALIZACJI ROZKROJU ELEMENTÓW PŁASKICH

Streszczenie. W pracy omówiono możliwość zastosowania algorytmów genetycznych do optymalizacji rozkroju materiałów w dwuwymiarowym problemie rozkroju. Problem polega na rozkroju prostokątnych płyt na szereg prostokątnych części przy założeniu gilotynowego cięcia dwustopniowego. Przedstawiono sposób kodowania rozwiązań oraz operatory genetyczne. Zamieszczono również wyniki obliczeń dla przykładowych zadań rozkroju dwuwymiarowego i jednowymiarowego.

OPIMIZATION OF CUTTING MATERIALS IN THE 2D CUTTING STOCK PROBLEM USING GENETIC ALGORITHMS

Summary. The paper presents the application of genetic algorithm for optimization of the cutting in the 2D Cutting Stock Problem. The problem of cutting rectangular pieces from rectangular sheets with minimum loss is analysed. Moreover, we assume that the cutting machine is just able to cut the rectangles in two phases with a single straight cut. The coding method of this problem and the genetic operators are shown. The numerical examples to illustrate the proposed algorithm are solved.

1. Wstęp

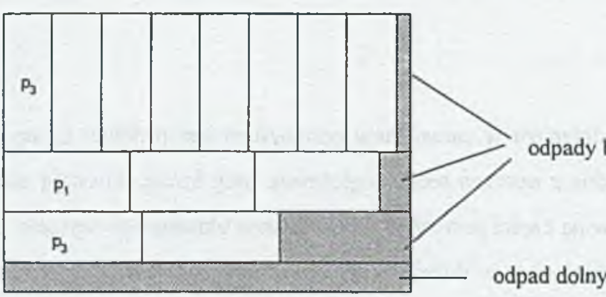
Ważnym zagadnieniem w zarządzaniu przemysłem jest problem oszczędności surowców i materiałów. Jedną z operacji technologicznych, przy której powstają znaczne odpady, jest rozkrój materiałów na części potrzebne do wykonania określonego wyrobu. Sposób podejścia do problemu rozkroju jest w dużej mierze uzależniony od dziedziny produkcji, w której proces cięcia jest realizowany. Wynika to z różnorodności rozkrawanych materiałów wyjściowych, technik rozkroju i organizacji procesów produkcyjnych [5]. W pracy rozważany jest problem optymalnego rozkroju prostokątnych płyt na szereg prostokątnych elementów. Rozkrawanym materiałem mogą być: drewniane płyty, arkusze papieru czy też tafle szkła.

Optymalizacja rozkroju materiałów dokonywana jest często z rozróżnieniem dwóch faz: generowania wzorów rozkroju oraz rozwiązywania zadania optymalizacyjnego, polegającego na określeniu, ile płyt należy pociąć zgodnie z każdym wzorem, aby wykonać zamówienie produkcyjne. W opisanym w pracy podejściu do problemu obie te fazy przebiegają jednocześnie. W każdym kroku zaproponowanego algorytmu przetwarzany jest zbiór dopuszczalnych rozwiązań danego zadania, przy czym pojedyncze rozwiązanie uwzględnia od razu sposoby rozkroju wszystkich płyt wykorzystanych w procesie produkcyjnym. Aby przetwarzanie tak dużej ilości informacji przebiegało efektywnie, zdecydowano się na zastosowanie algorytmów genetycznych.

Ważnym celem pracy było zaprojektowanie takiego sposobu kodowania oraz dobranie takich operatorów genetycznych, aby każde przetwarzane rozwiązanie było rozwiązaniem dopuszczalnym.

2. Opis problemu

Problem polega na rozkroju prostokątnych płyt o jednakowych wymiarach na szereg prostokątnych elementów z uwzględnieniem ograniczeń co do liczby części poszczególnych typów. Cięcie odbywa się jedynie po liniach prostych, równolegle do krawędzi arkusza przez całą jego długość. W fazie pierwszej cięcia otrzymujemy pasy, natomiast w fazie drugiej - żądane części (cięcie dwustopniowe [5]). Jako kryterium oceny rozwiązań przyjęto *wydajność rozkroju*, tj. stosunek sumy pól powierzchni wycinanych części do sumy pól powierzchni rozkrojonych materiałów wyjściowych (w %). Kryterium to podlega maksymalizacji, co jest równoważne minimalizacji odpadu powstałego w procesie cięcia.



Rys. 1. Przykładowy wzór rozkroju
Fig. 1. Some pattern of cutting

Każdy sposób rozkroju materiału będziemy nazywać *wzorem cięcia*. Rysunek 1 przedstawia przykładowy wzór rozkroju płyt z zadania 1, rozpatrywanego w dalszej części pracy.

Wprowadźmy oznaczenia i pojęcia. Załóżmy, że dysponujemy jednym rodzajem arkuszy wyjściowych o wymiarach W i S. Przyjmijmy również założenie, które będzie obowiązywać w całej pracy, że pierwsza faza cięcia odbywać się będzie prostopadle do wymiaru W, nazywanego umownie wysokością arkusza. Indeksami $i = 1,\ldots,n$ oznaczmy poszczególne rodzaje części p, które chcemy uzyskać w procesie rozkroju. Niech w_i i s_i będą wymiarami elementu p_i. Każdy element może wystąpić w dwóch orientacjach: (w_i, s_i) lub (s_i, w_i). Zbiór wszystkich rodzajów elementów z uwzględnieniem ich orientacji możemy zapisać w postaci:

$$P = \{p_1^1, p_2^1, \ldots, p_n^1, p_1^2, p_2^2, \ldots, p_n^2\}$$

Górny indeks oznacza orientację elementu, a dolny jego rodzaj. Z każdym elementem zbioru P związany jest inny rodzaj pasa, jaki można uzyskać w pierwszej fazie rozkroju. Zbiór pasów oznaczmy przez $K = \{K_1^1, K_2^1, \ldots, K_n^1, K_1^2, K_2^2, \ldots, K_n^2\}$.

W pierwszym etapie działania algorytmu dla każdego elementu p_i wyznaczana jest taka liczba pasów typu K_i^1 i K_i^2, aby wykonać zamówienie produkcyjne przy jednoczesnej minimalizacji odpadu bocznego związanego z pasem K_i^j, gdzie $i = 1,2,\ldots,n$, $j = 1,2$. Warunki te możemy zapisać w postaci:

$$\begin{cases} \min(k_i^1 y_i^1 + k_i^2 y_i^2) \text{ dla } k_i^1 y_i^1 + k_i^2 y_i^2 \geq x_i \\ \min\left[k_i^1 (Sv_i^1 - y_i^1 w_i s_i) + k_i^2 (Sv_i^2 - y_i^2 w_i s_i)\right] \end{cases}, \tag{1}$$

gdzie:

- x_i - zapotrzebowana liczba elementów i-tego typu,
- k_i^j - liczba pasów typu K_i^j,
- y_i^j - liczba elementów typu p_i zawartych w pasie K_i^j,
- v_i^j - wysokość pasa typu K_i^j, przy czym $v_i^1 = w_i$, a $v_i^2 = s_i$.

Wartość wyrażenia $k_i^1 y_i^1 + k_i^2 y_i^2$ określa uzyskaną liczbę elementów i-tego typu, a wyrażenie $S v_i^j - y_i^j w_i s_i$ - wielkość odpadu dla pojedynczego pasa K_i^j. Przyjmując założenie, że każdy nadmiar ponad zapotrzebowaną liczbę elementów i-tego typu również traktujemy jako odpad, drugi warunek w układzie (1) należałoby zapisać w postaci:

$$\min\left[S(k_i^1 w_i + k_i^2 s_i) - x_i w_i s_i\right] \tag{2}$$

Przy ustalonej wartości k_i^1, minimalną liczbę pasów typu K_i^2, potrzebną do wykonania zamówienia, znajdujemy ze wzoru:

$$k_i^2 = \text{ceil}\left(\frac{x_i - k_i^1 y_i^1}{y_i^2}\right), \tag{3}$$

gdzie **ceil** jest funkcją zaokrąglającą daną liczbę w górę do najbliższej liczby całkowitej. Zatem, aby wyznaczyć takie k_i^1 i k_i^2, dla których odpad boczny jest najmniejszy, można posłużyć się procedurą, która dla ustalonego i oraz zmieniających się w pętli wartości k_i^1 od 0 do

$\mathrm{ceil}\left(\frac{x_i}{y_i^1}\right)$ oblicza ze wzoru (3) wartości k_i^2 oraz wielkość odpadu, zapamiętując najlepsze rozwiązanie. Przy wyznaczonych już dla każdego i wartościach k_i^1 i k_i^2 sformułowany w pracy dwuwymiarowy problem rozkroju z cięciem dwustopniowym sprowadza się do rozkroju z cięciem jednostopniowym, który z kolei można utożsamiać z rozkrojem jednowymiarowym. Nasze zadanie możemy teraz sformułować następująco: pewną liczbę arkuszy wyjściowych o wysokości W należy pociąć na określoną liczbę (k_i^j) pasów o wysokościach y_i^j. Problem optymalizacji rozkroju polegać będzie na znalezieniu takich sposobów rozkroju płyt wyjściowych, dla których łączny odpad powstający przy cięciu będzie minimalny. Kryterium minimalizacji odpadu można zamienić na kryterium minimalizacji liczby rozcinanych płyt wyjściowych potrzebnych do uzyskania zamówionej liczby części. Fakt ten wynika ze wzoru na rzeczywistą wydajność rozkroju

$$Fcel = \frac{\sum_{i=1}^{n} x_i w_i s_i}{mWS}, \tag{4}$$

gdzie m jest liczbą zużytych podczas rozkroju płyt wyjściowych. Im mniej płyt zużyjemy, tym wydajność będzie większa, a co za tym idzie - mniejszy powstanie odpad.

Przyjmując stuprocentowe wypełnienie arkuszy wyjściowych ze wzoru (4) wyznaczyć można minimalną liczbę arkuszy potrzebną do wykonania zamówienia:

$$m_{\min} = ceil\left(\frac{\sum_{i=1}^{n} x_i w_i s_i}{WS}\right) \tag{5}$$

Po uwzględnieniu założeń dla sformułowanego w pracy problemu możemy ze wzoru (4) wyznaczyć kres górny możliwej do uzyskania wydajności rozkroju (w %):

$$F_{\sup} = \frac{\sum_{i=1}^{n} x_i w_i s_i}{m_{\min} WS} \cdot 100\% \tag{6}$$

Zastosowana metoda będzie omówiona dla przykładowego zadania optymalizacji rozkroju elementów płaskich. Przykład ten ma charakter ilustracyjny.

Przykład: Firma posiada surowiec w postaci płyt o jednakowym wymiarze. Otrzymuje zamówienie produkcyjne na wykrojenie czterech typów elementów, po 10 sztuk każdego rodzaju. Przyjmijmy:

- wymiary płyt (arkuszy) – 140×208,
- wymiary elementów: $p_1 - 64 \times 30$, $p_2 - 45 \times 36$, $p_3 - 70 \times 25$, $p_4 - 55 \times 50$.

Wyliczmy dla tego zadania minimalną liczbę płyt potrzebną do wykonania zamówienia oraz maksymalną możliwą do uzyskania wydajność rozkroju. Ze wzorów (5) i (6) otrzymujemy: $m_{\min} = \mathrm{ceil}(2.76) = 3$; $F_{\sup} = 92.03\%$.

3. Reprezentacja rozwiązania

Bardzo istotnym aspektem badań nad możliwością zastosowania algorytmów genetycznych w różnych zagadnieniach optymalizacji jest zaprojektowanie odpowiedniej do zadania struktury reprezentacji rozwiązania. Jednym z celów podjętych prac było opracowanie takiego sposobu kodowania rozwiązania oraz zastosowanie takich operatorów genetycznych, aby każde otrzymane i przetwarzane rozwiązanie było rozwiązaniem dopuszczalnym. Chodziło w dużej mierze o to, aby uchronić się przed koniecznością stosowania różnych metod naprawy rozwiązania lub nakładania kary na rozwiązania naruszające określone ograniczenia. W zastosowanym algorytmie każde rozwiązanie kodowane jest w postaci permutacji $lpas$ liczb naturalnych, gdzie $lpas = \sum_{i=1}^{n} (k_i^1 + k_i^2)$ - liczba pasów. Permutacje te określają kolejność, z jaką w rozkroju uwzględniane są poszczególne pasy. Zilustrujmy to na przykładzie.

W pierwszym etapie działania algorytmu, zgodnie z opisaną metodą, znajdowana jest liczba pasów każdego typu,potrzebna do wykonania zamówienia. Dla omawianego zadania otrzymano następującą macierz:

$$k_i^j = \begin{bmatrix} 0 & 4 \\ 2 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \quad \text{dla} \quad A_i^j = \begin{bmatrix} 64 & 30 \\ 45 & 36 \\ 70 & 25 \\ 55 & 50 \end{bmatrix} \text{, gdzie } i = 1, \ldots, 4, \ j = 1,2$$

Dla przykładu $k_2^1 = 2$ oznacza, że w rozkroju należy uwzględnić 2 pasy o wysokości 45 ($A_2^1 = 45$). Ponieważ $lpas = 1$, zatem rozwiązania będą kodowane w postaci 11-elementowych permutacji.

Na podstawie macierzy k_i^j i A_i^j tworzony jest wzorzec, względem którego dekodowane będą rozwiązania:

$$wzór = (70\ 55\ 50\ 50\ 45\ 45\ 30\ 30\ 30\ 30\ 25)$$

Na poszczególnych pozycjach tego wzorca umieszczono wysokości pasów (posortowane), jakie będą uwzględniane w rozkroju. Założono ponadto, że wzorzec ten odpowiada permutacji:

$$(1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11)$$

Dla losowo wygenerowanej permutacji można określić teraz wzory rozkroju poszczególnych płyt. Permutacji

$$T = (6\ 2\ 9\ 11\ 3\ 8\ 7\ 4\ 5\ 1\ 10)$$

odpowiada następujący ciąg pasów:

(45 55 30 | 25 50 30 30 | 50 45 | 70 30)

Po uwzględnieniu ograniczenia, że suma wysokości pasów umieszczonych na jednym arkuszu nie może przekroczyć wysokości arkusza ($W = 140$), otrzymujemy sposoby rozkroju poszczególnych płyt oddzielone separatorem " | ". Oczywiście, kolejność pasów w poszczególnych wzorach, jak również kolejność samych wzorów rozkroju nie są istotne, stąd też dla różnych permutacji możemy otrzymać taką samą wydajność rozkroju.

Obliczona ze wzoru (4) wydajność rozkroju dla permutacji T wyniosła 69.02%.

4. Opis algorytmu

Jako metodę przetwarzania rozwiązań zaproponowano algorytmy genetyczne. W swojej budowie zastosowany algorytm bazuje na elementarnym algorytmie genetycznym [1], [4].

Kroki algorytmu:

° *Generowanie populacji początkowej*

Populacja początkowa składa się z losowo wygenerowanych permutacji, które stanowią zakodowany zbiór rozwiązań danego zadania. Każdy osobnik populacji początkowej podlega rozkodowaniu i ocenie.

Główna pętla programu

2° *Selekcja osobników*

Selekcja do puli rodzicielskiej odbywa się metodą wyboru losowego według reszt bez powtórzeń [1], [4].

3° *Krzyżowanie*

W celu dobrania odpowiednich metod przetwarzania rozwiązań eksperymentowano z różnymi operatorami genetycznymi, opisanymi w literaturze, a stosowanymi do zadań z reprezentacją porządkową (zadanie komiwojażera, harmonogramowanie). Przebadano trzy operatory krzyżowania [4]: krzyżowanie z częściowym odwzorowaniem (PMX), krzyżowanie bazujące na porządku oraz krzyżowanie cykliczne (CX). Najlepsze efekty uzyskano dla krzyżowania cyklicznego. Zachowuje ono bezwzględne pozycje elementów u rodziców. Mechanizm krzyżowania cyklicznego zilustrujmy dla przykładowych dwóch rodziców:

$$R1 = (1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6\ \ 7\ \ 8\ \ 9\ \ 10\ \ 11)$$

$$R2 = (6\ \ 9\ \ 2\ \ 11\ \ 3\ \ 8\ \ 7\ \ 4\ \ 5\ \ 1\ \ 10).$$

Startując od pierwszego elementu pierwszego rodzica tworzymy cykl odwzorowań:

$$\leftrightarrow 6 \leftrightarrow 8 \leftrightarrow 4 \leftrightarrow 11 \leftrightarrow 10 \leftrightarrow .$$

Potomków tworzymy pozostawiając na tych samych pozycjach elementy uwzględnione w wyróżnionym cyklu, natomiast pozostałe elementy wymieniamy miejscami między rodzicami. Otrzymujemy w ten sposób dwóch potomków:

$$D1 = (1\ \ \mathbf{9}\ \ \mathbf{2}\ \ 4\ \ \mathbf{3}\ \ 6\ \ \mathbf{7}\ \ 8\ \ \mathbf{5}\ \ 10\ \ 11)$$

$$D2 = (6\ \ \mathbf{2}\ \ \mathbf{3}\ \ 11\ \ \mathbf{5}\ \ 8\ \ \mathbf{7}\ \ 4\ \ \mathbf{9}\ \ 1\ \ 10).$$

4° *Mutacja*

Mutacja jest operacją przeprowadzaną na pojedynczych osobnikach. Zastosowano operator polegający na zamianie miejscami dwóch losowo wybranych elementów w ciągu. Osobnik

$$R = (1\ \ 9\ \ 2\ \ 4\ \ 3\ \ 6\ \ 7\ \ 8\ \ 5\ \ 10\ \ 11)$$

poddany mutacji dałby w wyniku ciąg:

$$D = (1\ \ 9\ \ 7\ \ 4\ \ 3\ \ 6\ \ 2\ \ 8\ \ 5\ \ 10\ \ 11)$$

5° *Ocena populacji*

Każdy osobnik w populacji podlega procedurze dekodowania i oceny. Ocena osobnika polega na obliczaniu zgodnie ze wzorem (4) jego przystosowania (wydajności rozkroju).

Koniec pętli

Opuszczenie pętli następuje po przekroczeniu ustalonej liczby iteracji lub w momencie uzyskania wydajności równej F_{sup}.

6° *Wyprowadzenie wyników*

5. Wyniki obliczeń

W przeprowadzonych obliczeniach przyjęto następujące wartości parametrów algorytmu genetycznego:

- rozmiar populacji = 50,
- liczba pokoleń = 80,
- prawdopodobieństwo krzyżowania = 0.6,
- prawdopodobieństwo mutacji = 0.1.

Poniżej przytoczono wyniki obliczeń dla przykładowych zadań rozkroju. Ze względu na dużą liczbę rozkrawanych płyt ograniczono się jedynie do podania uzyskanych wydajności rozkroju. Wydajność rozkroju wyliczana jest zgodnie ze wzorem (4), a co za tym idzie - każdy nadmiar ponad zamówioną liczbę elementów danego rodzaju również traktowany jest jako odpad.

Zadanie 1. Wymiary arkusza - 40×208. Wymiary i wymagana liczba elementów:

wysokość	64	45	70	55
szerokość	30	36	25	50
liczba elem.	100	100	100	100

Uzyskana wydajność rozkroju: 86.28 % .

Zadanie 2. Wymiary arkusza - 80×360.

wysokość	60	57	54	51	48	45	42	39	36	33	30
szerokość	15	16	18	19	21	22	24	25	27	28	30
liczba elem.	80	80	90	90	100	100	110	110	120	120	130

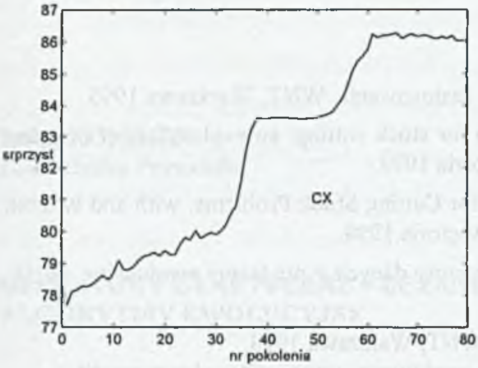
Uzyskana wydajność rozkroju: 92.77% .

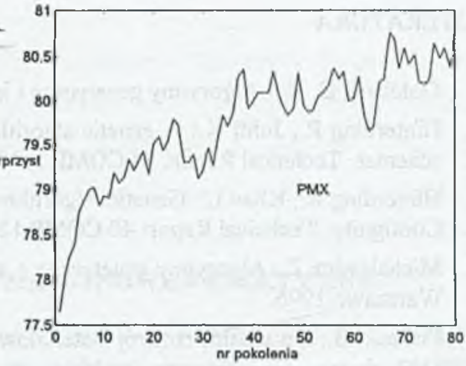
Zadanie 3. Rozkrój jednowymiarowy [H] [mag]. Wymiary arkusza - 4×1.

wysokość	3	4	5	6	7	8	9	10
szerokość	1	1	1	1	1	1	1	1
liczba elem.	5	2	1	2	4	2	1	3

Uzyskana wydajność rozkroju 97.62% jest równa $F_{\sup}$ dla tego zadania i została znaleziona w 23 pokoleniu.

Przebieg procesu poszukiwania rozwiązania można zilustrować wykresami średniego przystosowania populacji, wyrażonego przez wydajność rozkroju (w %) w kolejnych pokoleniach. Na rysunku 2 przedstawiono takie wykresy dla zadania 1 oraz dla dwóch typów krzyżowania: CX i PMX.





Rys.2. Wykresy średniego przystosowania populacji w kolejnych krokach algorytmu genetycznego dla krzyżowania CX i PMX

Fig.2. Diagrams of the average fitness of population in the next steps of genetic algorithm for CX and PMX crossover

6. Uwagi końcowe i wnioski

Różnorodność zagadnień rozkroju materiałów nie daje podstaw do formułowania ogólnych wniosków na temat budowy modeli zagadnień rozkroju i metod ich rozwiązywania. Co prawda wiele zadań rozkroju daje się sformułować w postaci modeli programowania liniowego, wiąże się to jednak z koniecznością uwzględniania różnorodnych ograniczeń związanych z danym zagadnieniem oraz z ogromną liczbą wzorów cięcia, które powinny być brane pod uwagę w obliczeniach. Składa się to na dużą złożoność obliczeniową problemu rozkroju. Jednak ze względu na komercyjne znaczenie rozwiązania tego zagadnienia celowe wydaje się poszukiwanie efektywnych algorytmów obliczeniowych. Wśród alternatywnych metod poszukiwania optymalnego (bądź bliskiego optymalnemu) rozwiązania coraz większą popularnością cieszą się algorytmy genetyczne. Podjęte badania miały na celu zbadanie efektywności algorytmów genetycznych w poszukiwaniu rozwiązań pewnych typów zadań optymalizacji rozkroju. Zastosowana metoda nie gwarantuje, co prawda, uzyskania optymalnego rozwiązania, jednak zamieszczone w pracy wyniki wydają się obiecujące.

Przewiduje się kontynuację podjętych prac. Będą one zmierzały w kierunku poszukiwania bardziej efektywnych, heurystycznych operatorów genetycznych, jak również innych sposobów kodowania rozwiązania, wierniej oddających charakter problemu. W celu poszerzenia obszaru zastosowań zaproponowanej metody należy również uwzględnić możliwość określania różnych rozmiarów arkuszy wyjściowych.

LITERATURA

1. Goldberg D. E.: Algorytmy genetyczne i ich zastosowania. WNT, Warszawa 1995.
2. Hinterding R., Juliff K.: A genetic algorithm for stock cutting: an exploration of mapping schemes. Technical Report 24 COMP 3, Victoria 1993.
3. Hinterding R., Khan L.: Genetic Algorithms for Cutting Stock Problems: with and without Contiguity. Technical Report 40 COMP 12, Victoria 1994.
4. Michalewicz Z.: Algorytmy genetyczne + struktury danych = programy ewolucyjne. WNT, Warszawa 1996.
5. Piasecki B.: Optymalny rozkrój materiałów. WNT, Warszawa 1978.
6. Zając J.: Dwuetapowy algorytm generowania wzorów rozkroju elementów płaskich. Materiały IX Konferencji nt "Metody i środki projektowania wspomaganego komputerowo", Warszawa 1993.

Recenzent: Dr hab.inż. Konrad Wala, prof. AGH Kraków

Abstract

The paper presents the possibilities of applying genetic algorithms to optimize the cutting of materials in the 2D Cutting Stock Problem. The problem consists in cutting many rectangular elements (items) of different size from rectangular sheets with the minimum waste of material. The cutting machine is able to cut rectangles in two steps with a single straight cut (two-step cut).

Two different types of strips, which can be obtained in the first phase of cutting, are connected with each element. At the beginning we determine such a set of strips that the lateral wastage is minimised and the production order is completed. Thus our problem is reduced to the cutting problem with one-step cut. Now each potential solution to a problem may be represented as a permutation of the strips from this set. Strips are cut in the order in which they are represented in the permuted list (chromosome). An initial population is built by generating random permutations. The reproduction operators have been borrowed from the Travelling Salesman Problem. Three types of crossover operators: partially mapped recombination (PMX), cycle recombination (CX) and order-based recombination have been analysed. The best results have been obtained for the CX crossover. The Second genetic operator - mutation - replaces two randomly chosen numbers in a sequence.

The results of the research show that genetic algorithms can be useful in searching for the optimal (near optimal) solution of the Cutting Stock Problem. It would be advisable to indicate the heuristic operators, specific to this problem.