



Damian Kurzydym

ORCID <https://orcid.org/0000-0003-2857-405X>

Akademia Nauk Stosowanych w Raciborzu, Instytut Techniki

ZASTOSOWANIE METOD HEURYSTYCZNYCH DO OPTYMALIZACJI CIĘCIA ARKUSZY ROBOTEM KAWASAKI

Streszczenie (abstrakt): W artykule przedstawiono zrobotyzowane stanowisko do optymalnego cięcia laserem. Proces cięcia laserem wiąże się z tematyką optymalizacji, która jest trudna i stanowi niejednokrotnie problem dla wielu pracowników. W związku z tym pomocy należy szukać w różnych algorytmach oraz metodach programowania robotów. Problem optymalizacji przedstawiony w pracy polega na optymalnym rozkładzie elementów prostokątnych na arkuszu, przy zachowaniu minimalnej powierzchni odpadu. Przeprowadzono symulacje off-line oraz rzeczywiste testy zrobotyzowanego wycinania dla robota Kawasaki z zastosowanym laserem edukacyjnym. Zastosowane w pracy algorytmy heurystyczne dają zawsze dobre (niekoniecznie najbardziej optymalne) rozwiązania, o zagwarantowanej jakości i niewiele odbiegające od rozwiązania optymalnego, przy niewielkich nakładach obliczeniowych. Ostateczna robotyzacja cięcia laserowego zapewnia znacznie zwiększenie wydajności oraz uzyskanie lepszej jakości ciętych powierzchni. Dodatkowo zwiększa ona powtarzalność wykonywanych elementów oraz pozwala na masową produkcję elementów o bardzo skomplikowanych kształtach.

Słowa kluczowe: cięcie laserem, optymalizacja, programowanie robota Kawasaki, symulacja off-line, algorytmy heurystyczne

APPLICATION OF HEURISTIC METHODS FOR OPTIMIZING SHEET CUTTING WITH A KAWASAKI ROBOT

Abstract: The article presents a robotic workstation for optimal laser cutting. The laser cutting process is associated with the issue of optimization, which is difficult and often a problem for many workers. Therefore, it is essential to seek help in various algorithms and robot programming methods. The optimization problem presented in the work involves the optimal arrangement of rectangular elements on a sheet, while minimizing the waste area. Off-line simulations and real tests of robotic cutting were carried out for the Kawasaki robot with the educational laser used. The heuristic algorithms used in the work always provide good solutions (not necessarily the most optimal), with guaranteed quality and not far from the optimal solution, with low computational costs. Ultimately, the robotization of laser cutting significantly increases productivity and improves the quality of the cut surfaces. Additionally, it enhances the repeatability of manufactured components and enables mass production of parts with very complex shapes.

Keywords: laser cutting, optimization, Kawasaki robot programming, off-line simulation, heuristic algorithms

Wstęp do zrobotyzowanego cięcia laserowego

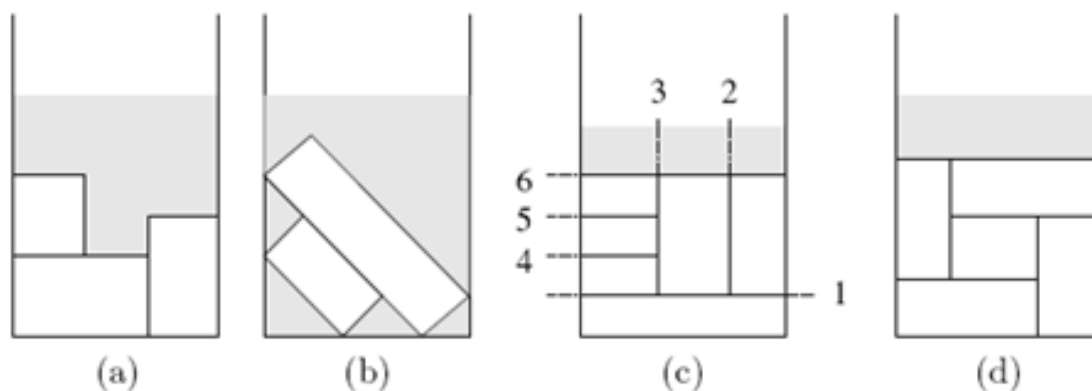
Początki badań nad problemem rozkroju datuje się na lata 40-te dwudziestego wieku. Wówczas rozwiązywano problem rozkroju prostokąta na kwadraty. Związek między problemem rozkroju a programowaniem liniowym dostrzeżono w latach 50. W latach 60-tych rozwinęło się zastosowanie metod programowania liniowego do poszukiwania optymalnego rozwiązania jedno- i dwuwymiarowego problemu rozkroju. Matematyczne modele rozkroju, w których udowodniono, że problem rozkroju jest NP-zupełny (trudnym zagadnieniem teoretycznym algorytmiki), zostały przedstawione w latach 70. Prace opisujące różne metody heurystyczne, w tym algorytm rozmieszczania prostokątnych elementów według określonego porządku na prostokątnym arkuszu, zaczynając od dolnego lewego rogu arkusza, przedstawiono w latach 80-tych. W 1985 r. po raz pierwszy zastosowano algorytm genetyczny do rozwiązania problemu pakowania pudełek. We wczesnych latach 90-tych, najczęściej stosowanymi i opisywanymi algorytmami poszukiwania optymalnych rozwiązań zadań rozkroju są metody heurystyczne oraz metody ewolucyjne. W 1995 r. zaczęto także stosować algorytm genetyczny do dwuwymiarowego problemu pakowania pudełek. W 1997 r. opisano dwa podejścia do problemu rozkroju materiałów, bazujące na sztucznych sieciach neuronowych. Analizując przegląd wykorzystywanych metod optymalnego rozkroju materiałów, właściwym kierunkiem badań jest rozwijanie technik heurystycznych oraz metod sztucznej inteligencji. Spowodowane jest to bardzo dużą złożonością obliczeniową problemu rozkroju¹.

Problemy cięcia i pakowania są optymalizacyjnymi problemami kombinatorycznymi stosowanymi w przemyśle. W wyniku procesu cięcia następuje odpad, dlatego należy zmniejszyć odpad najbardziej jak to jest możliwe dla osiągnięcia większych dochodów przedsiębiorstwa. Problemy tego typu dotyczą przydziału z kompletu zbioru małych rzeczy do dużego obszaru z ograniczeniami². Podczas problemu pakowania i cięcia można wprowadzić od algorytmów dodatkowe ograniczenia (Rys. 1)³.

¹ I. Kierkosz, *Zastosowanie algorytmów genetycznych do optymalizacji rozkroju gilotynowego z ograniczeniami*, [w]: *Polioptrymalizacja i Komputerowe Wspomaganie Projektowania*, tom 2, (red.) W. Tarnowskiego, T. Kiczkowiaka, WNT, Mielno 2003.

² H. Dyckhoff, *A Typology of Cutting and Packing Problems*, „European Journal of Operational Research”, 1990, nr 44(2).

³ B. Piasecki, *Optymalny rozkrój materiałów*, WNT, Warszawa 1978.



Rys. 1. Ograniczenia i typy cięcia: (a) pakowanie prostokątne, (b) pakowanie dowolne, (c) cięcie-gilotynowe, (d) cięcie dowolne

Proces cięcia laserowego polega na skierowaniu energii wiązki promieniowania laserowego w miejsce linii cięcia materiału, co prowadzi do miejscowego stopienia i odparowania ciętego materiału. Razem z przepływem współosiowym wiązki laserowej jest zastosowany przepływ gazu reaktywnego lub obojętnego, który wydmuchuje ze szczeliny cięcia ciekły materiał i pary materiału⁴.

Typowe stanowisko do zrobotyzowanego cięcia laserem jest ogrodzone dwoma pomieszczeniami, w jednym pomieszczeniu jest człowiek a w drugim robot, i dodatkowo oddzielone przegrodą nieprzepuszczającą światła lasera. W pomieszczeniach tych jest umiejscowiony stół obrotowy (pozycjoner) z urządzeniami, które są potrzebne do mocowania wyrobu przeznaczonego do cięcia (Rys. 2).



Rys. 2. Stanowisko do cięcia laserowego firmy Fanuc⁵

Cięcie przestrzenne wymaga zastosowania robota przemysłowego, który posiada sześć stopni swobody. Aby było możliwe cięcie kształtów złożonych, przegub ramienia robota powinien mieć zwiększony zakres ruchu. Robot zastosowany do cięcia powinien

⁴ A. Klimpel, *Spawanie, zgrzewanie i cięcie metali*, Technologie, WNT 2009.

⁵ J. Barczyk, M. Kłosowiak, *Robotyzacja cięcia laserowego*, „Pomiary Automatyka Robotyka” 2005, nr 6, s. 14-17.

charakteryzować się dużymi prędkościami oraz dynamiką ruchu, aby była możliwość minimalizacji czasów produkcji.

Algorytmy heurystyczne

Heurystyka to strategia rozwiązywania problemów optymalizacyjnych, które dają „dobre”, ale niekoniecznie optymalne rozwiązanie, lecz możliwe do akceptacji. Problem pakowania pasa jest problemem NP-trudnym, a jego zastosowanie prowadzi do dużej skali problemów. Praktykujący zwykle chętniej stosują metody heurystyczne niż metody dokładne.

Pierwsza klasa heurystyk to algorytmy poziomu. Każdy prostokąt jest pakowany wg określonego porządku do zbioru poziomych poziomów, utworzonych w pasie. Pakując elementy do pasa, pierwszy poziom jest dnem pasa. Natomiast wysokość każdego poziomu jest określana najwyższą wysokością z prostokątów poprzedniego poziomu (pozioma linia jest narysowana przez szczyt najwyższego prostokąta w poprzednim poziomie).

Algorytm NFDH (z ang. *next fit decreasing height*), tak zwany „następny pasujący ze zmniejszającej się wysokości”, został rozwinięty w 1980 r. Lista prostokątów do upakowania jest posortowana, zgodnie z malejącą wysokością. Pierwszy prostokąt, umieszczony na pierwszym poziomie (dnie pasa), określa wysokość następnego poziomu. Prostokąt jest umieszczany na aktualnym poziomie do dolnej lewej strony, jeśli oczywiście pasuje. W przypadku, gdy prostokąt nie pasuje na aktualnym poziomie, utworzony zostaje nowy aktualny poziom i prostokąt zostaje umieszczony na nowym poziomie. Proces pakowania odbywa się od lewej do prawej i z dołu pasa w górę. Poziomy niższe od aktualnego poziomu, nigdy nie są przeszukiwane. Prostokąty o równej wysokości, zachowują oryginalny porządek względem pozostałych w liście pakowania. Kroki algorytmu opisuje pseudokod przedstawiony poniżej.

Opis: Pakowanie listy prostokątów do pasa o ustalonej szerokości i nieskończonej wysokości. Lista prostokątów jest wcześniej znana, przed rozpoczęciem procesu pakowania.

Wejście: Liczba prostokątów do spakowania n , wymiary prostokątów $\{w(L_i), h(L_i)\}$ oraz szerokość pasa W .

Wyjście: Wysokość z pakowania pasa.

- 1: poziom $\leftarrow 0$; $h(\text{poziom}) \leftarrow 0$; $w(\text{poziom}) \leftarrow 0$; $i \leftarrow 1$
- 2: Posortuj prostokąty od największej wysokości do najmniejszej:

$$h(L_1) \geq h(L_2) \geq \dots \geq h(L_n)$$
- 3: Pakuj prostokąt L_i do lewej dolnej części pasa
- 4: $h(\text{poziom}) \leftarrow h(L_i)$; $w(\text{poziom}) \leftarrow w(L_i)$
- 5: **for** $i=2, \dots, n$ **do**
- 6: **if** $W-w(\text{poziom}) \geq w(L_{i+1})$ **then**
- 7: pakuj prostokąt L_{i+1} do prawej prostokąta L_i
- 8: $w(\text{poziom}) \leftarrow w(\text{poziom}) + w(L_{i+1})$
- 9: **else** $[W-w(\text{poziom}) < w(L_{i+1})]$

```

10: utwórz nowy poziom powyżej poprzedniego i pakuj prostokąt  $L_{i+1}$  na nowym poziomie
11:  $\text{poziom} \leftarrow \text{poziom} + 1$ ;  $w(\text{poziom}) \leftarrow w(L_{i+1})$ ;  $h(\text{poziom}) \leftarrow h(\text{poziom} - 1) + h(L_{i+1})$ 
12: end if
13: end for
14: drukuj  $H = h(\text{poziom})$ 

```

Algorytm FFDH (z ang. *first fit decreasing height*), tak zwany „pierwszy pasujący ze zmniejszającej się wysokości”, też jest datowany na 1980 r. Prostokąty tej samej wysokości zachowują oryginalny porządek jeden do drugiego w liście pakowania. Prostokąt jest umieszczany do lewej na najniższym poziomie, gdzie znajduje się wolne miejsce. Pas jest przeszukiwany od najniższego poziomu do najwyższego w celu przeszkupienia wolnego miejsca dla wstawienia aktualnego prostokąta. Jeśli aktualny prostokąt nie pasuje do któregoś z istniejących poziomów, wtedy nowy poziom zostaje utworzony nad aktualnym poziomem szczytu (który staje się nowym poziomem szczytu) i aktualny prostokąt zostaje umieszczony na nowym poziomie. Różnica między algorytmem NFDH i FFDH jest taka, że w algorytmie FFDH podczas pakowania prostokąta następuje dodatkowe przeszukiwanie poprzednich poziomów w celu znalezienia wolnego miejsca dla aktualnie wymagającego spakowania prostokąta⁶. Kroki algorytmu opisuje pseudokod przedstawiony poniżej.

Opis: Pakowanie listy prostokątów do pasa o ustalonej szerokości i nieskończonej wysokości. Lista prostokątów jest znana wcześniej, przed rozpoczęciem procesu pakowania.

Wejście: Liczba prostokątów do spakowania n , wymiary prostokątów $\{w(L_i), h(L_i)\}$ oraz szerokość pasa W .

Wyjście: Wysokość z pakowania pasa.

```

1:  $\text{poziom} \leftarrow 0$ ;  $h(\text{poziom}) \leftarrow 0$ ;  $i \leftarrow 1$ ;  $\text{PoziomNr} \leftarrow 1$ 
2: Posortuj prostokąty od największej wysokości do najmniejszej:
   
$$h(L_1) \geq h(L_2) \geq \dots \geq h(L_n)$$

3: Pakuj prostokąt  $L_i$  do lewej dolnej części pasa;  $h(\text{poziom} + 1) \leftarrow h(L_i)$ 
4: for  $i = 2, \dots, n$  do
5: przeszukaj wszystkie poziomy w poszukiwaniu wolnego miejsca (zaczynając od dołu)
6: if taki poziom istnieje then
7: pakuj prostokąt  $L_i$  do lewej na tym poziomie
8: else [we wszystkich istniejących poziomach brak wolnego miejsca]
9:  $\text{PoziomNr} \leftarrow \text{PoziomNr} + 1$ ;  $\text{poziom} \leftarrow \text{PoziomNr}$ ;  $h(\text{poziom}) \leftarrow h(\text{poziom} - 1) + h(L_i)$ 
10: pakuj prostokąt na nowym poziomie
11: end if
12: end for
13: drukuj  $H = h(\text{poziom})$ 

```

⁶ E. G. Coffman Jr., D. S. Garey, R. E. Tarjan, *Performance Bounds for Level Oriented Two-Dimensional Packing Algorithms*, „SIAM Journal on Computing” 1980, nr 4.

Na stronie internetowej⁷ można skorzystać z symulacji programu z zastosowaniem algorytmu NFDH i FFDH. Na początku algorytm losuje listę 20 prostokątów. W kolejnym kroku program sortuje listę prostokątów, kolejno od największej do najmniejszej szerokości. Następną czynnością jest pakowanie prostokątów wg wybranego algorytmu do pasa o szerokości $W=100$ i nieskończonej wysokości. Wynikiem końcowym jest wysokość pasa (szczyt) oraz ilość zmarnowanej przestrzeni wewnątrz pasa w %.

Problem optymalizacyjny

Problem w pracy można opisać w następujący sposób. Dysponując materiałem wyjściowym, tj. prostokątnymi płytami o szerokości S i wysokości W oraz kompletem znanych n elementów L , prostokątnych z indeksami $i=1, 2, \dots, n$, należy rozmieścić elementy prostokątne w sposób optymalny na arkuszach przy minimalnej powierzchni odpadu. Wymiar elementu prostokątnego to s_i i d_i (szerokość i długość). Rozcinana płyta umieszczana jest w prostokątnym układzie współrzędnych Oxy w taki sposób, aby lewy dolny wierzchołek płyty pokrył się z początkiem układu współrzędnych.

Założenia dodatkowe to:

- rozkrój niegilotynowy,
- każde cięcie przebiega równoległe do krawędzi arkusza (rozkroj ortogonalny),
- wycinane elementy mają ustaloną orientację, tzn. nie mogą być obracane o 90° ,
- wszystkie dane wejściowe są liczbami całkowitymi,
- spełniony jest warunek ($s_i \leq W$ oraz $d_i \leq S$), co oznacza możliwość umieszczenia przynajmniej jednego elementu prostokątnego na płycie,
- jeśli w arkuszu nie można umieścić wszystkich elementów prostokątnych, jest możliwość kontynuowania rozmieszczania reszty po wymianie arkusza na nowy.

Zadanie optymalizacji polega na maksymalizacji stopnia wykorzystania arkusza wyjściowego, a co za tym idzie minimalizacji odpadu i arkuszy wyjściowych.

Symulacje off-line oraz rzeczywiste testy przeprowadzono na podstawie programów napisanych wg algorytmów NFDH i FFDH, osobno dla pisaka i lasera edukacyjnego, wykorzystując listę prostokątów L (Tabela 1) do rozmieszczania w arkuszach.

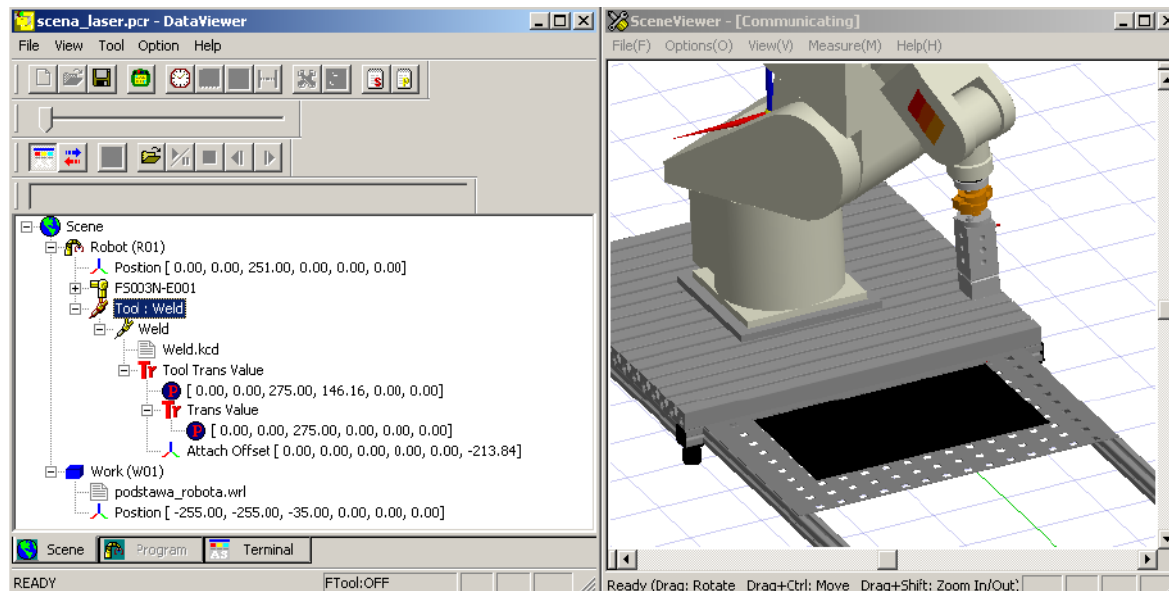
Tabela 1. Lista wylosowanych elementów prostokątnych

Lista przypadkowo wylosowanych prostokątów L ($s[mm]$, $d[mm]$)																				
Lp.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
s	15	10	65	20	47	79	35	72	18	100	47	47	22	87	65	52	69	95	13	87
d	150	10	64	100	50	110	150	106	53	100	57	57	60	29	100	25	60	160	44	92

⁷ H. Smith, *Bin Packing*, <http://users.cs.cf.ac.uk/C.L.Mumford/heidi/Approaches.html> [dostęp: 18.09.2024].

Wykonanie modeli pracy CAD

Aby wirtualny robot w programie symulacyjnym PC-ROSET nie odbiegał od rzeczywistego robota, w programie NX została zamodelowana scena i narzędzia, osobno dla rysowania przez pisak oraz dla zrobotyzowanego cięcia wykorzystując edukacyjny laser (Rys. 3).



Rys. 3. Model sceny robota Kawasaki FS003N z narzędziem lasera

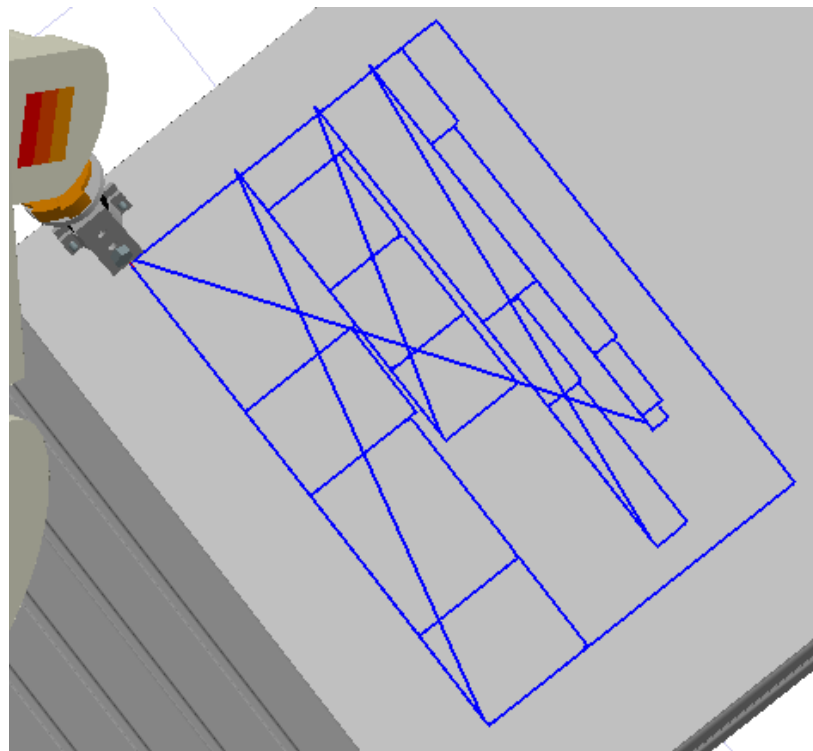
Symulacje off-line oraz rzeczywiste testy

W środowisku PC-ROSET na początku dokonano symulacji programów NFDH i FFDH z wykorzystaniem narzędzia pisak. Po wykonaniu symulacji z pisakiem, przedstawiono symulację zrobotyzowanego cięcia laserem edukacyjnym.

Po uruchomieniu programu w PC-ROSET z odpowiednią sceną, należało wprowadzić dane wejściowe arkusza oraz prostokątów przygotowanych wcześniej (Tabela 1). Wyniki symulacji (trajektorii końcówki pisaka dla programu *NFDH_pisak*) już po wymianie arkusza na arkusz nowy nr 2 (wszystkie prostokąty nie mogły zostać rozmieszczone w jednym arkuszu) przedstawia Rys. 4.

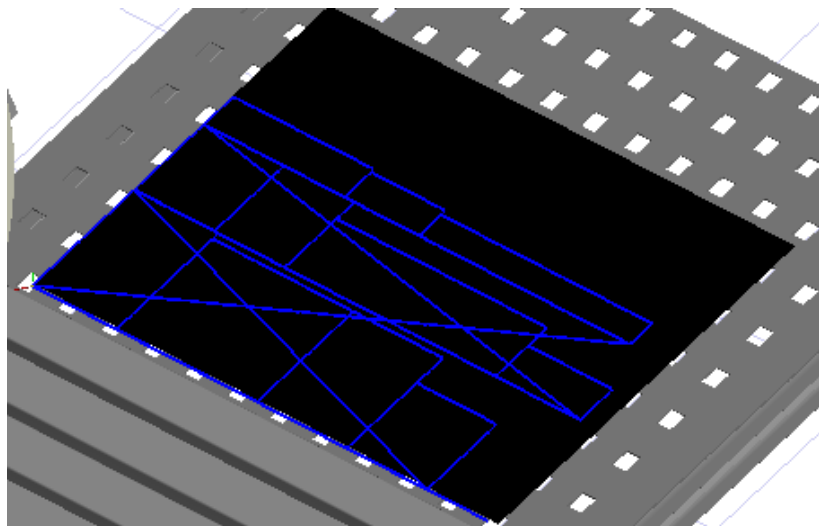
Dla sceny wraz z narzędziem lasera edukacyjnego wyniki symulacji (trajektorii wiązki ogniska lasera dla programu *FFDH_laser*) już po wymianie arkusza na arkusz nowy nr 2 (wszystkie prostokąty nie zmieściły się w jednym arkuszu) przedstawia Rys. 5

Z kolei do rzeczywistej symulacji napisanych wcześniej programów z wykorzystaniem robota Kawasaki FS003N użyto programu komputerowego *Terminal KC Win TCP* za pomocą połączenia *TCP IP*.



Wyniki danych dla arkusza nr 2 :
 Wysokosc arkusza zajeta przez prostokaty wynosi: 177 mm
 Pole arkusza zajete przez prostokaty i straty wynosi: 58410 mm² - 100 %
 Suma pol prostokatow wynosi: 43886 mm² - 75.1 %
 Suma pol strat wynosi: 14524 mm² - 24.9 %

Rys. 4. Wyniki symulacji programu *NFDH_pisak* dla arkusza nr 2

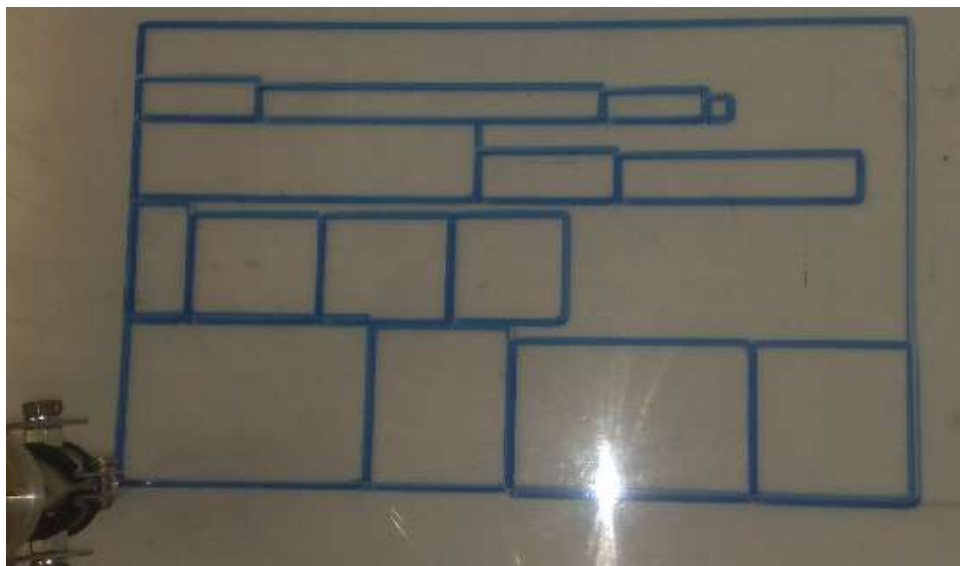


Wyniki danych dla arkuszaz nr 2 :
 Wysokosc arkusza zajeta przez prostokaty wynosi: 136 mm
 Pole arkusza zajete przez prostokaty i straty wynosi: 44880 mm² - 100 %
 Suma pol prostokatow wynosi: 34282 mm² - 76.4 %
 Suma pol strat wynosi: 10598 mm² - 23.6 %

Rys. 5. Wyniki symulacji programu *FFDH_laser* dla arkusza nr 2

Po uruchomieniu programu (*NFDH_pisak*) i wprowadzeniu danych wejściowych arkusza oraz prostokątów (Tabela 1), wyniki rzeczywistej symulacji (rysowania pisaka)

przedstawia Rys. 6 (po wymianie arkusza na arkusz nowy nr 2, gdyż wszystkie prostokąty nie mogły zostać rozmieszczone w jednym arkuszu).



Wyniki danych dla arkusza nr 2 :
Wysokość arkusza zajęta przez prostokąty wynosi: 177 mm
Pole arkusza zajęte przez prostokąty i straty wynosi: 58410 mm² - 100 %
Suma pól prostokątów wynosi: 43886 mm² - 75.1 %
Suma pól strat wynosi: 14524 mm² - 24.9 %

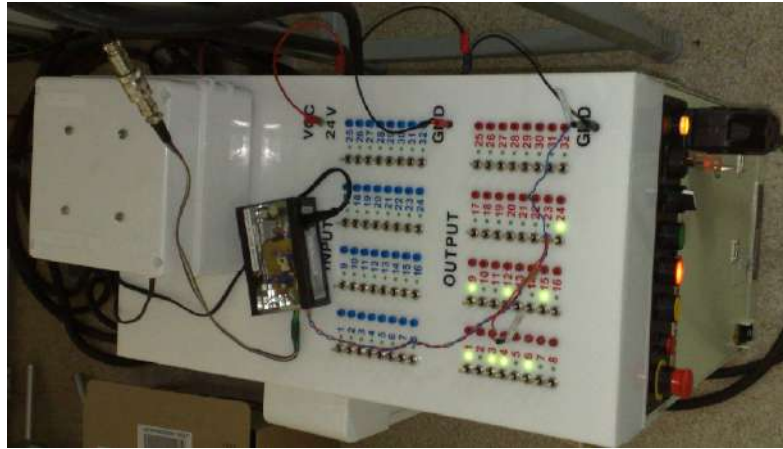
Rys. 6. Wyniki rzeczywistej symulacji programu *NFDH_pisak* dla arkusza nr 2

Do rzeczywistej symulacji cięcia laserem edukacyjnym (Rys. 7) zastosowano moduł lasera o mocy wyjściowej 1W i długości fali 455nm.



Rys. 7. Robot podczas cięcia laserem edukacyjnym

Rys. 8 przedstawia kontroler D70 wraz z podłączonym sterownikiem lasera do układu wyjść podczas pracy zrobotyzowanego cięcia laserem edukacyjnym.



Rys. 8. Sterownik lasera podłączony do kontrolera D70

Rys. 9 przedstawia wyniki rzeczywistej symulacji (dla programu *FFDH_laser*), w wyniku której otrzymano końcowe wycięte prostokąty i odpad po wymianie arkusza na nowy arkusz nr 2 (wszystkie prostokąty nie mogły zostać rozmieszczone w jednym arkuszu).



Wyniki danych dla arkusza nr 2 :
 Wysokość arkusza zajęta przez prostokąty wynosi: 136 mm
 Pole arkusza zajęte przez prostokąty i straty wynosi: 44880 mm² - 100 %
 Suma pól prostokątów wynosi: 34282 mm² - 76.4 %
 Suma pól strat wynosi: 10598 mm² - 23.6 %

Rys. 9. Wyniki rzeczywistej symulacji programu *FFDH_laser* dla arkusza nr 2

Wnioski

W artykule przedstawiono rozwiązanie zadania polegającego na optymalizacji wycinania laserem elementów prostokątnych z arkusza tak, aby powierzchnia odpadu była minimalna. Porównano dwa programy wykorzystujące algorytmy heurystyczne NFDH i FFDH. Programy te zostały rozszerzone o możliwość wymiany arkusza na nowy w momencie, gdy z aktualnego arkusza nie da się wyciąć wszystkich elementów prostokątnych. Algorytmy optymalizacyjne są trudne, czasochłonne i niejednokrotnie nie dają dokładnych rozwiązań. Jednak zastosowane w pracy algorytmy heurystyczne dają zawsze dobre (nie-

koniecznie optymalne) rozwiązania, niewiele odbiegające od rozwiązania optymalnego przy niewielkich nakładach obliczeniowych.

Wyniki symulacji komputerowych oraz rzeczywistych testów zarówno dla programów NFDH oraz FFDH są identyczne. Dlatego w pierwszej kolejności lepiej jest wykonać symulację komputerową, a następnie po przeprowadzonych pozytywnych testach przenieść badany program do rzeczywistego robota. Porównując badane programy, które działają na podstawie algorytmów heurystycznych NFDH i FFDH, można stwierdzić jednoznacznie, iż programy oparte na algorytmach FFDH dają zdecydowanie lepsze wyniki. Doskonale to można zauważyć w symulacji komputerowej (wg trajektorii robota), gdzie trajektoria wiązki ogniska lasera wg programu FFDH wykonuje powroty do wcześniejszych poziomów, aby jeszcze zmieścić dodatkowy prostokąt, natomiast w programie NFDH takich powrotów nie ma. Różnice w wynikach obliczeń dla badanych programów nie są znaczące, gdy w obydwu algorytmach trzeba użyć dwóch arkuszy. Pomimo tego algorytm FFDH rozmieszcza lepiej prostokąty, w wyniku czego arkusz pierwszy jest wykorzystany maksymalnie a drugi jest wykorzystany w niecałych trzech czwartych jego powierzchni, co daje możliwość wykorzystania odpadu w przyszłości. Różnica w wynikach badanych programów byłaby znacząca wówczas, gdyby trzeba byłoby rozmieścić kilkadziesiąt elementów prostokątnych, które zajęłyby kilkanaście arkuszy wyjściowych. Wówczas z pewnością stosując program FFDH zużyto by mniej arkuszy niż podczas wykorzystania programu NFDH.

Program FFDH charakteryzuje się:

- stosowaniem w konkretnych warunkach produkcyjnych,
- dużą prostotą obsługi (wprowadzenie tylko danych wejściowych),
- danymi wyjściowymi w postaci planów rozkroju,
- dużą szybkością obliczeniową w czasie rzeczywistym.

Zastosowanie w praktyce przemysłowej programu FFDH przynosi następujące efekty:

- oszczędności materiału względem metod ręcznych,
- skrócenie cyklu realizacji planów rozkroju oraz ich pracochłonności,
- zmniejszenie zapasów i powierzchni magazynowych.

Wybierając metody rozwiązania zadań z zakresu optymalizacji, należy kierować się rachunkiem ekonomicznym, którego wynik będzie różny w zależności od warunków, w jakich funkcjonuje przedsiębiorstwo. W większości przypadków zastosowanie metod matematycznych i komputerowych pozwala na znaczne oszczędności materiałów.

Podsumowując, robotyzacja cięcia laserowego zapewnia znacznie zwiększenie wydajności oraz uzyskanie lepszej jakości ciętych powierzchni. Dodatkowo zwiększa powtarzalność wykonywanych elementów oraz pozwala na masową produkcję elementów o bardzo skomplikowanych kształtach.

Bibliografia

1. Barczyk J., Kłosowiak M., *Robotyzacja cięcia laserowego*, „Pomiary Automatyka Robotyka” 2005, nr 6.
2. Coffman Jr. E. G., Garey D. S., Tarjan R. E., *Performance Bounds for Level Oriented Two-Dimensional Packing Algorithms*, „SIAM Journal on Computing” 1980, nr 4.
3. Dyckhoff H., *A Typology of Cutting and Packing Problems*, „European Journal of Operational Research” 1990, nr 44(2).
4. Kierkosz I., *Zastosowanie algorytmów genetycznych do optymalizacji rozkroju gilotynowego z ograniczeniami*, [w]: *Poliptymalizacja i Komputerowe Wspomaganie Projektowania*, tom 2, (red.) W. Tarnowskiego, T. Kiczowski, WNT, Mielno 2003.
5. Klimpel A., *Spawanie, zgrzewanie i cięcie metali. Technologie*, WNT, Warszawa 2009.
6. Piasecki B., *Optymalny rozkrój materiałów*, WNT, Warszawa 1978.
7. Smith H., *Bin Packing*, <http://users.cs.cf.ac.uk/C.L.Mumford/heidi/Approaches.html> [dostęp: 18.09.2024].

Dane kontaktowe

Damian Kurzydym, damian.kurzydym@akademiarac.edu.pl