

Damian KRENCZYK, Bożena SKOŁUD
Politechnika Śląska

SEKWENCJONOWANIE PRODUKTÓW W MIESZANYCH LINIACH MONTAŻOWYCH Z ZASTOSOWANIEM ALGORYTMÓW HEURYSTYCZNYCH

Streszczenie. Wzrastająca konkurencja i oczekiwania klientów silnie wpływają na przedsiębiorstwa, które poszukują sprawnych metod planowania. W artykule przedstawiono porównanie zachłannego algorytmu heurystycznego oraz heurystyki przeszukiwania lokalnego w postaci algorytmu symulowanego wyżarzania w rozwiązywaniu problemu wyznaczonej sekwencji pojazdów dla wybranej klasy wielowersyjnych linii montażowych. Dzięki odpowiednio dobranym parametrom, m.in. resetu temperatury, oraz generowaniu rozwiązania wejściowego o wysokim poziomie jakości, wpłynęło na zmniejszenie prawdopodobieństwa wybrania rozwiązania gorszego niż bieżące i spowodowało większą skuteczność algorytmu symulowanego wyżarzania.

SEQUENCING MIXED-MODEL ASSEMBLY LINE USING HEURISTIC ALGORITHMS

Summary. Increasing competition and customer expectations have a strong impact on companies that are looking for efficient planning methods. The paper presents a comparison of the greedy heuristic algorithm and the local search heuristic in the form of a Simulated Annealing algorithm for the sequencing selected class of mixed-model vehicle assembly lines problem. Properly selected parameters, incl. temperature reset, due to the generation of an input solution with a high level of quality, reduce the probability of selecting a solution worse than the current one and cause greater efficiency of the Simulated Annealing algorithm.

1. Wprowadzenie

Problem balansowania i sekwencjonowania w liniach montażowych jest żywy w literaturze, co wynika z rzeczywistej potrzeby poszukiwania balansu i odpowiedniej sekwencji, przy zmianie modeli produktów, jako odpowiedzi na oczekiwania klientów. Takie trendy dotyczą produkcji w różnych branżach, począwszy od branży motoryzacyjnej, produkcję sprzętu AGD, sprzętu elektronicznego i innych.

W niektórych pracach (np. [2]) rozważana jest produkcja (montaż) partiami, pomiędzy którymi przewidywane jest przebrojenie zużywające zasoby i czas. W takim

przypadku oprócz problemu sekwencjonowania dodatkowo analizowany jest jeszcze problem wyznaczania wielkości partii produkcyjnych.

Częste zmiany asortymentu podnoszą znacznie koszty re-konfiguracji linii oraz wpływają na nierównomierne zużycie materiałów i czas realizacji zamówienia. Z tego powodu obserwuje się coraz częstsze odchodzenia od produkcji w partiach na rzecz wprowadzenia do produkcji sekwencji wszystkich typów produktów na liniach montażowych modeli mieszanych (ang.: mix model assembly line - MMAL). Różniące się produkty są montowane w tym samym systemie produkcyjnym, ale procesy są na tyle podobne, że czas przebrojenia nie występuje lub jest pomijalnie mały. Podejście to jest szczególnie ważne i stosowane w montażu ostatecznym samochodów. Linie montażowe modeli mieszanych są powszechnie stosowane ze względu na ich elastyczność w odniesieniu do zmian modelu. Celem stosowania tego typu linii jest wpływ na zmniejszenie zapasów produktu końcowego i zapewnienie ciągłego przepływu materiałów. Jednak nieefektywne wykorzystanie zasobów i z tym związane obniżenie produktywności skutkuje wysokimi kosztami jednostkowymi ze względu na wyższe koszty początkowe inwestycji [7]. Celem sekwencjonowania mieszanych modeli produktów na liniach montażowych jest więc utrzymanie stałego wykorzystania stanowisk na każdej linii montażowej przy najniższych kosztach dostosowania linii. Autorzy [7] zaproponowali inne od tradycyjnego MMAL, gdzie stacje robocze na liniach montażowych połączone w sposób szeregowo-równoległy. Poszukiwano rozwiązania sekwencji produktów w multi-MMAL (MMMAL) montażu samochodów zorientowanej na zamówienie w oparciu o metody wyszukiwania zmiennych sąsiedztwa (VNS). Wyniki pokazują, że proponowane podejście jest wykonalne dla optymalizacji sekwencjonowania produktów zorientowanej na zamówienie i daje rozwiązania dopuszczalne.

Potrzeba optymalizacji sekwencji składnia naukowców do poszukiwania metod i algorytmów poprawiających rozwiązania. W [13] przedstawiono zastosowanie algorytmów mrówkowych. Autorzy [12] zaproponowali poszukiwanie rozwiązań z wykorzystaniem algorytmu pszczołowego (ang. *artificial Bees Algorithm-ABA*). Z kolei w [14] zaproponowano dwupoziomowy model problemu programowania liniowego mieszane całkowitoliczbowego (*bi-level programming*) do balansowania i sekwencjonowania w procesie zarządzania produkcją montażu samochodowego. Autorzy [3] zajęli się koordynacją pomiędzy poszczególnymi etapami planowania. Zaproponowali wprowadzenie centralnych wzorcowych modeli harmonogramowania, które mogą służyć jako interfejs między długo- i średnioterminowym równoważeniem linii montażowych a problemem krótkoterminowego sekwencjonowania oraz ponownego sekwencjonowania w odpowiedzi na nieprzewidziane zakłócenia w toku produkcji.

Podsumowując wstępną analizę można stwierdzić, że problem sekwencjonowania linii montażowych jest ciągle otwarty. Każde nowe, skuteczniejsze rozwiązanie przyczynia się do wzrostu produktywności i poziomu wykorzystania zasobów. W niniejszym artykule przedstawiono kolejną próbę rozwiązania tego problemu. Analizie poddano porównanie wyników uzyskanych z wykorzystaniem dedykowanego algorytmu zachłannego (sekwencja bazowa) oraz uzyskanych przez zastosowanie algorytmu symulowanego wyżarzania.

2. Problem sekwencjonowania pojazdów na linii montażowej

W artykule rozpatrywany jest problem sekwencjonowania pojazdów na linii końcowego montażu, w literaturze określany jako *Car Sequencing Problem* (CSP) [8, 11]. Proponowane formalne modele matematyczne CSP ewoluowały od lat 80-tych XX wieku wraz z ewolucją organizacji linii montażowych związanych z koniecznością wytwarzania wielu modeli i ich odmian (wersji) w różnych wolumenach na tej samej linii (*mixed-model* i *multi-model*). Problem CSP został po raz pierwszy opisany w [10]. W oryginalnym ujęciu zestaw uwzględnianych ograniczeń dotyczy możliwych opcji dla danej wersji produktu oraz definicji tzw. stosunku p/q , w którym w dowolnej pod-sekwencji kolejno zaplanowanych do montażu na linii pojazdów q może występować co najwyżej p wersji pojazdów, w których występuje dana opcja. CSP definiowany jest jako krotka [11]:

$$(V, O, p, q, r), \quad (1)$$

gdzie:

$V = \{v_1, \dots, v_n\}$ – zbiór pojazdów do wykonania,

$O = \{o_1, \dots, o_m\}$ – zbiór różnych opcji mogących występować w danej wersji produktu,

$p: O \Rightarrow \mathbb{N}$ i $q: O \Rightarrow \mathbb{N}$ – ograniczenie wydajności związane z każdą z opcji $o_i \in O$
– dla dowolnej pod-sekwencji q_i produktów może występować co najwyżej p_i pojazdów wymagających opcji o_i ,

$r: V \times O \Rightarrow \{0, 1\}$ – wymagania dotyczące opcji,

– dla każdego pojazdu $v_j \in V$ i dla każdej opcji $o_i \in O$, $r_{ij} = 1$, jeżeli o_i jest wymagana w pojeździe v_j oraz $r_{ij} = 0$, jeżeli o_i nie występuje.

Rozwiązanie problemu CSP sprowadza się do znalezienia kolejności elementów zbioru V , spełniającej zdefiniowane ograniczenia, określając w ten sposób kolejność, w jakiej pojazdy (a ściślej ich wersje powiązane z przynależnym zestawem opcji) będą montowane na danej linii. W szczególności wymagane jest udzielenie odpowiedzi na pytanie, czy można znaleźć taką sekwencję, która spełni wszystkie ograniczenia p/q . Ograniczenia p/q , pierwotnie związane były z przyjętym sposobem montażu, w którym projektowano stacje dedykowane do montażu danej opcji dla określonego maksymalnego odsetka pojazdów przemieszczających się przez linię [10, 11]. W takim przypadku, aby móc zainstalować daną w określonym procencie pojazdów (przy założeniu równomiernego obciążenia pracowników) na wybranych stacjach montażowych należało zaplanować wielo-ekipową obsadę operatorów oraz projektować ich długość jako wielokrotności odległości przejeżdżanej przez pojazd w czasie taktu linii [10].

W obecnie projektowanych liniach montażowych, szczególnie w branży automotive, opisany tryb pracy jest już bardzo rzadko spotykany. Nie stosuje się praktycznie przemieszczania się operatorów w trakcie pracy wzdłuż kilku stacji, a wymagania związane z dodatkową pracochłonnością dla pojazdu, w którym należy zainstalować daną opcję zostają uwzględnione w wyznaczonej uśrednionej długości stacji dla linii, w których pojazd przemieszcza się ze stałą prędkością wzdłuż linii (a co za tym idzie w uśrednionym takcie pracy linii). W takim podejściu zakłada się

możliwość przekroczenia strefy taktu dla tzw. ciężkich wersji (wersji zawierających wymaganą do instalacji opcję powodującą zwiększenie prędkości powyżej przyjętemu średniemu dla danej stacji), które jest niwelowane w trakcie realizacji operacji montażowych dla (zazwyczaj kilku) następujących w sekwencji tzw. lekkich wersji pojazdów. Skrajnie niepożądaną sytuacją jest zatem wystąpienie w sekwencji pojazdów pod rząd dwóch ciężkich wersji dla danej stacji. W praktyce, dla określonego zbioru opcji (i odpowiadających im wersji pojazdów, już na etapie projektowania linii, producenci definiują minimalną liczbę wersji niezawierających danej opcji, które powinny wystąpić po wersji ją zawierającej. Gwarantuje to osiągnięcie zaplanowanej wydajności linii. Wymagane zatem było zmodyfikowanie ograniczeń w modelu (1) w taki sposób, aby ograniczenie p przyjmowało zawsze wartość 1. Odpowiada to zaokrągleniu zestawu ograniczeń p/q , do $1/q$, co odpowiada opisanej powyżej sytuacji, w której wymagania odnośnie do częstości występowania opcji w sekwencji dotyczą minimalnej liczby pojazdów bez opcji pomiędzy każdymi dwoma ją posiadającymi [9].

W zastosowaniu do rozwiązania opisanego problemu sekwencjonowania, sekwencja wersji produktów została zdefiniowana jako uporządkowana macierz S , o wymiarach $m \times n$, której kolumny odpowiadają wektorom poszczególnych wersji produktu r [9]:

$$S = [s_{ij}], s_j \in r_j \quad (2)$$

gdzie:

i – numer opcji mogącej występować w danej wersji pojazdu, $i = 1 \dots m$,

m – liczba opcji,

j – j -ta pozycja w sekwencji realizacji wersji w cyklu produkcyjnym, $j=1 \dots n$,

n – liczba pojazdów do wykonania.

3. Algorytmy heurystyczne

W ogólności w CSP przestrzeń możliwych rozwiązań obejmuje wszystkie możliwe permutacje pojazdów V , wymagających różnych konfiguracji opcji O [11]. CSP jest problemem NP-trudnym, co zostało wykazane w [6] oraz [8]. Proponowane metody rozwiązywania CSP bazują na metodach dokładnych bądź przybliżonych [1, 4]. Pierwsza grupa opiera się na systematycznej eksploracji przestrzeni poszukiwań, aż do znalezienia rozwiązania lub udowodnienia, że brak jest rozwiązania. W zastosowaniach praktycznych mogą być one skuteczne w rozwiązywaniu przypadków o niskim n i nie są konkurencyjne do metod heurystycznych [11]. Szczególnie, że w problemach praktycznych, w wielu przypadkach nie istnieje rozwiązanie spełniające wszystkie ograniczenia, a problem decyzyjny sprowadza się do poszukiwania rozwiązania w jak najmniejszym stopniu je przekraczającego [9]. W grupie metod heurystycznych najczęstszymi są algorytmy zachłanne, przeszukiwanie lokalne, algorytmy genetyczne i algorytmy mrówkowe [11]. W opisywanych w tym artykule wynikach badań zastosowano podejścia oparte na dedykowanym algorytmie zachłannym oraz algorytmie symulowanego wyżarzania z różnymi parametrami.

Algorytm zachłanny

Rozwiązaniem CSP jest macierz S , której kolumny odpowiadają wektorom poszczególnych wersji pojazdu. Danymi wejściowymi są: zbiór V produktów do wykonania, zbiór O opcji mogących występować w danej wersji produktu oraz wymagania dotyczące opcji $r: V \times O \Rightarrow \{0,1\}$. Rozwiązanie tworzone jest zgodnie z podanymi poniżej krokami dedykowanego algorytmu [9]:

```

1: input:  $V, O, q, r, S = \mathbf{0}_{m \times n}, m, n$ 
2: for  $i = (\text{sort}(O, q_{oi}, \text{ascend}))$  do
3:    $V' = \{v_j: v_j \in V \wedge r_{ij}=1\}$ 
4:   while  $V' \neq \emptyset$  do
5:     select  $v_k \in V'$ 
6:     if  $\text{sum}_j(s_{ij}) > 0$  then
7:        $\text{gaps}() = \text{find\_gaps}(S, i)$ 
8:       find  $l$  s.t.  $\text{gaps}(l) \geq q_i$ 
9:       if  $\sim l$  then find  $l$  s.t.  $\text{gaps}(l) = \max(\text{gaps}())$ 
10:    else
11:      find  $l$  s.t.  $\text{sum}_l(s_{il}) = 0$ 
12:    end
13:    for  $1 < j < n$  do
14:       $s_{jl} = r_{jk}$ 
15:    end
16:     $V' = V' \setminus \{v_k\}$ 
17:  end
18: end

```

Algorytm ten został opracowany na potrzeby działu planowania jednego z przedsiębiorstw z branży automotive i został wykorzystany jako baza porównawcza w prezentowanych wynikach badań. Zastosowana w nim funkcja pomocnicza $\text{find_gaps}()$ wyznacza, dla każdej pozycji w aktualnie tworzonej sekwencji S , odległość do najbliższego produktu posiadającego opcję o_i . Algorytm faworyzuje ograniczenia dla wersji, dla których wzrost pracochłonności jest największy, tzn. takich, dla których wartość q_j ograniczenia wśród jeszcze nieprzydzielonych pojazdów jest największa. W przypadku, gdy nie jest możliwe znalezienie pozycji spełniającej narzucone warunki, produkt zostaje przydzielony w wolnym miejscu najbardziej oddalonych od już przydzielonego produktu posiadającego tę samą opcję.

Algorytm symulowanego wyżarzania

Zastosowany algorytm jest zmodyfikowaną wersją standardowego algorytmu symulowanego wyżarzania (Simulated Annealing Optimization SAO) [5]. Modyfikacje dotyczą zabezpieczenia przed zbyt długim pozostawaniem w obrębie lokalnego minimum. Określona została liczba iteracji (70), w których przyjęto nowo wygenerowane rozwiązanie, i po których zostaje wykonany reset temperatury. Obniżenie temperatury następujące po każdej iteracji k algorytmu realizowana jest zgodnie z funkcją wykładniczą $t(k) = t_i 0,95^k$. Natomiast prawdopodobieństwo $P(A)$ zaakceptowania gorszego niż bieżący punkt $f(S_{cm})$ kandydata sekwencji $f(S_{cnd})$ zdefiniowano w dwóch wariantach, jako:

$$P(A) = \frac{1}{1 + \exp\left(100 \left(\frac{f(S_{cnd})}{f(S_{cur})} - 1\right) / t\right)} \quad (3)$$

oraz:

$$P(A) = \frac{1}{1 + \exp(100(f(S_{cnd}) - f(S_{cur})) / t)} \quad (4)$$

Generowanie rozwiązania kandydata (sekwencji kandydującej) odbywa się poprzez wielokrotną zamianę dwóch losowo wybranych kolumn macierzy S . Wielokrotność ta jest zmienna i odpowiada iloczynowi n i stosunku aktualnej temperatury t do temperatury początkowej t_i [9]:

```

1: input: S, m, n, t
2: swaps = ceil(m*t/ti)
3: for 1<i<swaps do
4:   col_1 = randi(m)
5:   do
6:     col_2 = randi(m)
7:   loop until col_1==col_2
8:   for 1<i<n do
9:     temp = Si col_1
10:    Si col_1 = Si col_2
11:    Si col_2 = temp
12:   end
13: end

```

Sekwencje wyznaczone w wyniku wykonania algorytmu (dla obu funkcji prawdopodobieństwa) porównywane są z sekwencją losową oraz sekwencją bazową – sekwencją uzyskaną w wyniku wykonania algorytmu zachłannego.

Ocena rozwiązania

Ocena wyznaczonej sekwencji musi odzwierciedlać stopień spełnienia określonych ograniczeń wydajnościowych (1). W praktyce przemysłowej są one wynikiem założeń przyjętych podczas projektowania linii, jak i aktualnych parametrów (i zakłóceń) pracy linii, m.in. wolumenu produkcji, dostępności komponentów, obsady, itp. Przyjęta funkcja oceny jakości sekwencji odzwierciedlająca stopień spełnienia ograniczeń powinna uwzględniać wpływ stopnia spełnienia poszczególnych ograniczeń na regularność przepływu pojazdów przez kolejne etapy procesu, a co za tym idzie wydajność całej linii. Wpływ ten jest największy dla niespełnienia ograniczenia wersji trudnych, a więc opcji, dla których q_i jest największa. Biorąc powyższe pod uwagę zaproponowani wskaźniki jednostkowe opcji μ_i , rosnące wykładniczo tym szybciej, im przekroczenie jest większe, inaczej niż w podejściach tradycyjnych (biorących pod uwagę liczbę naruszeń ograniczeń w sekwencji S) [9]:

$$\mu_i^S = -(De)^{F\gamma_i^S - F} + 1, \quad (5)$$

gdzie:

D, F – stałe odpowiadające za nachylenie funkcji (przyjęto $D=3$ oraz $F=2$),

γ_i^S – średnia znormalizowana odległość występowania opcji o_i w sekwencji $\langle 0; 1 \rangle$:

$$\gamma_i^S = \frac{\sum_{j:(s_{ij}=1)} gap(ij)}{\sum_{j:(s_{ij}=1)} q_i}, \quad (6)$$

gdzie:

$gap(ij)$ – odstęp w sekwencji S pomiędzy produktem v_j oraz kolejnym posiadającym opcję o_i ,

Na ich podstawie wyznaczany jest ważony sumarycznego wskaźnik naruszenia:

$$\mu^S = \frac{\sum_i(\mu_i^S q_i)}{\sum_i(q_i)}, \quad (7)$$

podlegający minimalizacji podczas pracy algorytmu SA, w którym wagom przyporządkowano wartości q_i , przyjmującym wartości wyższe dla opcji trudnych.

4. Wyniki obliczeń

Zaproponowane w poprzednim rozdziale algorytmy zaimplementowano w środowisku obliczeniowym MATLAB. Dane wejściowe obejmują zbiór pojazdów do wykonania r w trakcie zmiany roboczej, zbiór opcji dla danej wersji oraz ograniczeniach l/q . Zbiór pojazdów do wykonania w trakcie zmiany roboczej zawiera informacje o wymaganych opcjach dla każdego pojazdu i określany jest przez dział koordynacji zamówień. Przykład takich wymagań podano w tabeli 1.

Tabela 1

Wymagania dotyczące opcji dla każdego wyrobu r_{ij} ($n=4$)

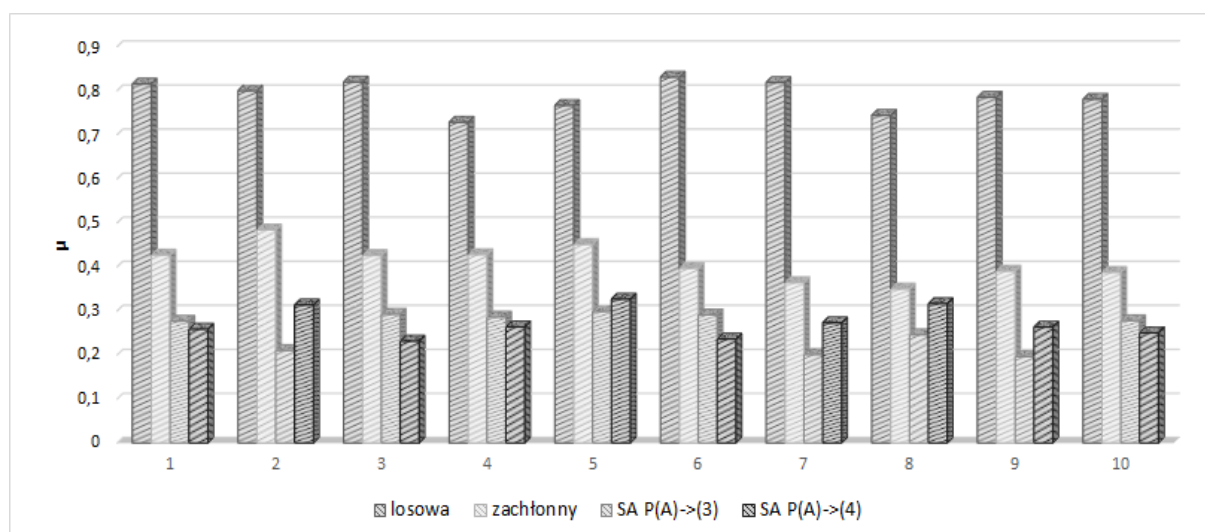
		Pojazd															
		i/j	1	2	3	4	5	6	7	8	9	10	11	12	13	14	
Opcja	1	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	...
	2	1	1	1	1	0	0	1	1	0	1	0	0	0	0	0	
	3	1	0	0	0	0	1	0	0	0	1	1	0	0	0	0	
	4	1	1	0	1	0	0	1	1	0	0	1	0	0	0	0	
		

Następnie wyznaczana zostaje sekwencja z zastosowaniem heurystyki zachłannej stanowiąca bazę porównawczą, i zostaje uruchomiony jest algorytm symulowanego wyżarzania dla obu funkcji prawdopodobieństwa (3) (4). Wybrane zbiorcze informacje, zawierające dane wejściowe oraz rozwiązania uzyskane z zastosowaniem poszczególnych algorytmów, przedstawiono w tabeli 2.

Tabela 2

Dane wejściowe oraz wyniki dla wybranych algorytmów

Lp.	Parametry wejściowe			Uzyskane wskaźniki jakości rozwiązania			
				rozw. losowe	alg. zachłanny	SA P(A) ->(3)	SA P(A) ->(4)
	m	n	q	μ	μ	μ	μ
1	40	9	[8 5 6 3 4 4 7 4 8]	0.8150	0.4267	0.2766	0.2583
2	40	9	[3 3 5 6 8 8 9 9 6]	0.7997	0.4839	0.2101	0.3133
3	40	9	[6 6 9 5 8 3 9 3 4]	0.8195	0.4258	0.2921	0.2314
4	40	9	[6 7 7 5 7 8 7 6 3]	0.7283	0.4277	0.2857	0.2636
5	40	9	[9 7 3 9 3 9 9 4 9]	0.7663	0.4511	0.2986	0.3267
6	40	9	[5 7 4 4 4 8 9 3 9]	0.8305	0.3966	0.2911	0.2358
7	40	9	[9 6 8 3 8 4 8 7 8]	0.8183	0.3641	0.2008	0.2727
8	40	9	[7 5 8 6 6 4 3 8 8]	0.7439	0.3498	0.2456	0.3162
9	40	9	[3 6 4 4 3 8 4 6 6]	0.7848	0.3904	0.1962	0.2632
10	40	9	[3 4 7 7 7 4 6 5 8]	0.7801	0.3874	0.2779	0.2501



Rys. 1. Porównanie wyników dla wybranych algorytmów. Źródło: opracowanie własne

We wszystkich przypadkach nastąpiła poprawa rozwiązania w porównaniu z wynikami uzyskanymi w wyniku zastosowania algorytmu zachłannego (rys. 1). Wariant funkcji prawdopodobieństwa $P(A)$ zaakceptowania kandydata sekwencji, gorszego niż wyznaczony dla bieżącego rozwiązania nie wykazywał stałego trendu i średnio dawał większą poprawę uzyskiwanego rozwiązania dla wariantu (3). Poprawa wyniku dla algorytmu SA w stosunku do algorytmu zachłannego wynosiła od 26% do 56% w przypadku zastosowania funkcji (3) oraz od 10% do 45% w przypadku zastosowania funkcji (4).

5. Podsumowanie

W artykule podjęto się kolejnej próby rozwiązania problemu sekwencjonowania pojazdów w linii montażowej. Zadanie jest odpowiedzią na realne zapotrzebowanie wynikające ze złożoności zagadnienia, czego są świadomi autorzy. Poszukiwali więc algorytmu, który pozwoli na uzyskanie rozwiązania bardziej korzystnego. W związku z tym porównali dwa algorytmy: algorytm zachłanny oraz zmodyfikowany algorytm symulowanego wyżarzania. Modyfikacja polega na zabezpieczeniu przed zbyt długim pozostawaniem w obrębie lokalnego minimum. Przedstawiono dwie funkcje prawdopodobieństwa zaakceptowania kandydata sekwencji. W obu przypadkach algorytm SA dał znacząco lepsze wyniki.

LITERATURA

1. Boysen N., Flidner M., Scholl A.: Sequencing mixed-model assembly lines: Survey, classification and model critique. *European Journal of Operational Research*, 2009, 192, s. 349–373.
2. Boysen N., Flidner M., Scholl A.: A classification of assembly line balancing problems. *European Journal of Operational Research*, 2007, Vol. 183, s. 674–693.
3. Boysen N., Flidner M., Scholl A.: Production planning of mixed-model assembly lines: overview and extensions. *Production Planning and Control*, 2009, Vol. 20:5, s. 455-471.
4. Chutima P. Olarnviwatchai S.: A multi-objective car sequencing problem on two-sided assembly lines. *Journal of Intelligent Manufacturing*, 2018, Vol. 29, s. 1617–1636.
5. Eglese R.W.: Simulated annealing: a tool for operational research. *European Journal of Operational Research*, 1990, Vol. 46, 27 s. 1-28.
6. Gent I.P.: Two Results on Car Sequencing Problems. Technical Report APES APES-02-1998.
7. Jiang Z., Li L., Li Z., Li Z.: Order-oriented cooperative sequencing optimisation in multi-mix-model assembly lines. *International Journal of Production Research*, 2012, 50:24, s.7198-7209.
8. Kis T.: On the complexity of the car sequencing problem. *Operations Research Letters*, 2004, Vol. 32, s. 331-335.
9. Krenczyk D.: Zastosowanie algorytmów heurystycznych do sekwencjonowania produktów w wielowersyjnych liniach montażowych. *Inżynieria zarządzania : Cyfryzacja produkcji. Aktualności badawcze 3* (Knosala R. red.), 2021, Warszawa, Polskie Wydaw. Ekonomiczne, s.659-669.
10. Parrello B., Kabat W., Wos L.: Job-shop scheduling using automated reasoning: A case study of the car-sequencing problem. *Journal of Automated Reasoning*, 1996, Vol. 2, s. 1-42.
11. Solnon C., Cung V.D., Nguyen A., Artigues C.: The car sequencing problem: Overview of state-of-the-art methods and industrial case-study of the

- ROADEF'2005 challenge problem. *European Journal of Operational Research*, 2008, Vol. 191(3), s.912-927.
12. Tapkan P., Ozbakir L., Baykasoglu A.: Modeling and solving constrained two-sided assembly line balancing problem via bee algorithms. *Applied Soft Computing* Volume 2012, 12, Issue 11, s. 3343-3355.
 13. Yagmahan B.: Mixed-model assembly line balancing using a multi-objective ant colony optimization approach. *Expert Systems with Applications*, 2011, Vol. 38(10), s. 12453-12461.
 14. Zhang W., Yuan L., Shentu H.: The model of balancing and scheduling for mix-model assembly line based on bi-level programming 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC).