

Problem przepływowy ze sprzężeniami czasowymi maszyn

Wojciech Bożejko¹, Radosław Idzikowski², Mieczysław Wodecki³

Wprowadzenie

W procesie zarządzania linią produkcyjną czy planowaniem robót budowlanych pojawia się problem harmonogramowania zadań, w którym z powodu dostępnych technologii pojawia się wiele ograniczeń. Proces taki jest w literaturze utożsamiany z klasycznym problemem przepływowym (ang. *flowshop problem*) [Bożejko, Hejducki, Wodecki 2012], należącym do klasy problemów silnie NP-trudnych, co organiczna stosowanie algorytmów dokładnych tylko dla małych wielkości instancji problemu. Dla większych rozmiarów stosuje się algorytmy metaheurystyczne [Ruiz, Maroto 2005].

W pracy przedstawiamy różne sprzężenia czasowe (zobacz także: [Adriri, Pohlryles 1982]) zależne od maszyny. W literaturze możemy spotkać ograniczenie bez przestojów na maszynach (ang. *no idle*) [Bożejko, Idzikowski, Wodecki 2016], my proponujemy dwa nowe ograniczenia, gdzie będziemy dopuszczać ograniczony przestój lub nawet go wymuszać.

Sformułowanie problemu z dodatkowymi ograniczeniami

W problemie przepływowym dany jest zbiór n zadań:

$$J = \{J_1, J_2, \dots, J_n\}, \quad (1)$$

które należy wykonać na m maszynach ze zbioru:

$$M = \{M_1, M_2, \dots, M_m\}. \quad (2)$$

¹ Politechnika Wrocławska, wojciech.bozejko@pwr.edu.pl

² Politechnika Wrocławska, radoslaw.idzikowski@pwr.edu.pl

³ Politechnika Wrocławska, mieczyslaw.wodecki@pwr.edu.pl

Każde zadanie składa się z m operacji $O_{i,j}$, z których każdą trzeba wykonać przez $p_{i,j}$ czasu na odpowiedniej maszynie M_j :

$$J_i = \{O_{i,1}, O_{i,2}, \dots, O_{i,m}\} \quad (3)$$

W przepływowym problemie permutacyjnym na każdej maszynie jest ta sama kolejność zadań, którą oznaczmy jako π . Liczba wszystkich możliwych rozwiązań wynosi $n!$. Omawiany problem sprowadza się to wyznaczenia takiej kolejności zadań, aby czas wszystkich zadań był jak najkrótszy.

Model matematyczny

W celu utworzenia harmonogramu, przez $S_{\pi(i),j}$ oznaczmy moment rozpoczęcia zadania $\pi(i)$ na maszynie M_j oraz przez $C_{\pi(i),j}$ moment zakończenia tego zadania na tej samej maszynie. Zadania wykonywane są nieprzerwanie, więc:

$$C_{\pi(i),j} = S_{\pi(i),j} + p_{\pi(i),j} \quad (4)$$

gdzie:

$p_{\pi(i),j}$ — czas wykonywania operacji j z zadania $\pi(i)$ na maszynie M_j .

Na każdej maszynie jednocześnie może być wykonywane tylko jedno zadanie:

$$C_{\pi(i+1),j} \geq C_{\pi(i),j} + p_{\pi(i+1),j} \quad (5)$$

W ramach zadań musi być zachowany porządek technologiczny:

$$C_{\pi(i),j+1} \geq C_{\pi(i),j} + p_{\pi(i),j+1} \quad (6)$$

Przy założeniach, że moment rozpoczęcia pierwszego zadania z permutacji na maszynie M_1 wynosi 0: $S_{\pi(1),1} = 0$. Na podstawie wzorów (4)–(6) jednoznacznie możemy stwierdzić, że czas zakończenia wszystkich zadań to moment zakończenia ostatniego zadania w permutacji π na maszynie M_m :

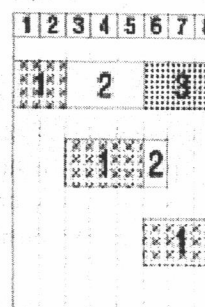
$$C_{\max} = C_{\pi(n),m} \quad (7)$$

W tabelicy 1 przedstawiono przykład danych wejściowych ze wszystkimi czasami operacji $p_{i,j}$ dla 5 zadań ($n = 5$) i 3 maszyn ($m = 3$). Rozpatrzmy uszeregowanie zadań klasycznego problemu przepływowego dla permutacji naturalnej $\pi = (1, 2, 3, 4, 5)$. Dokładny harmonogram przedstawia schemat Gantta na rysunku 1, gdzie widoczne jest, że czas zakończenia wszystkich zadań to $C_{\max} = 34$.

TABLICA 1. Czasy wykonywania operacji

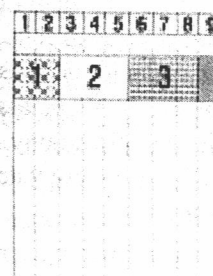
Zadanie	1	2	3	4	5	6
Maszyna 1	2	3	3	6	5	6
Maszyna 2	3	1	1	2	5	4
Maszyna 3	3	4	4	3	3	5

Źródło: Opracowanie własne.



Źródło: Opracowanie własne.

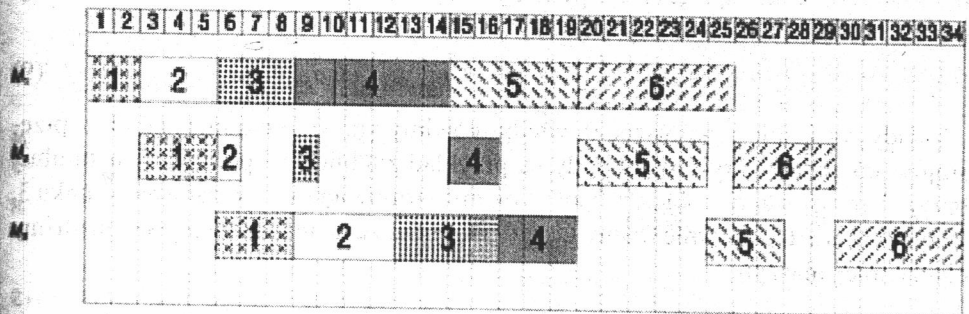
W literaturze najczęściej przestojów (ang. *no idl* zadania na danej maszynie) nie ma. Taki efekt można zobaczyć dla permutacji naturalnej $C_{\max} = 38$. Istotną zmianą nierówności z wzoru 5 r



Źródło: Opracowanie własne.

Przy analizowaniu przykładu stojów zaistniała potrzeba, aby nie ma potrzeby, aby

RYSUNEK 1. Klasyczny problem przepływowy



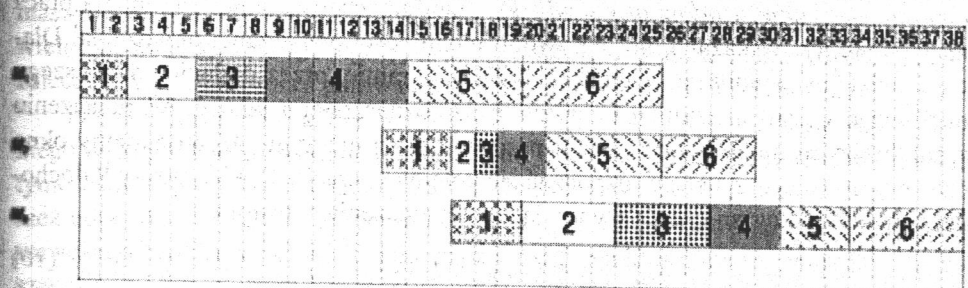
Źródło: Opracowanie własne.

Ograniczenie „bez przestojów”

W literaturze najczęściej spotykanym sprzężeniem czasowym jest ograniczenie bez przestojów (ang. *no idle*), które wymusza natychmiastowe rozpoczęcie kolejnego zadania na danej maszynie. Skutkiem jest opóźnienie rozpoczęcia operacji i dosunięcie harmonogramu do prawej, co w wielu przypadkach wydłuża harmonogram. Taki efekt można zobaczyć na rysunku 2, gdzie pokazano ponownie harmonogram dla permutacji naturalnej na przykładzie z tabeli 1 i otrzymaliśmy w rezultacie $C_{\max} = 38$. Istotną zmianą wprowadzaną przez to ograniczenie jest zmiana łagodnej nierówności z wzoru 5 na równanie:

$$C_{\pi(i+1),j} = C_{\pi(i),j} + p_{\pi(i+1),j} \quad (8)$$

RYSUNEK 2. Problem przepływowy bez przestojów



Źródło: Opracowanie własne.

„Ograniczony przestój”

Przy analizowaniu przykładów praktycznych dla problemu przepływowego bez przestojów zaistniała potrzeba rozwinięcia tego ograniczenia, ponieważ okazało często, że nie ma potrzeby, aby na wszystkich maszynach wymuszać ciągłą pracę maszyn.

każdą trzeba wykonać

(3)

na maszynie jest ta sama kolejność możliwych rozwiązań takiej kolejności

(4)

na maszynie tylko jedno zadanie:

(5)

logiczny:

(6)

zadania z permutacji mają w (4)–(6) jednoznacznie określony moment zakończenia

(7)

zgodnie ze wszystkimi warunkami (3). Rozpatrzmy uszeregowaną permutację naturalną i diagram Gantta na rysunku 1, dla którego $C_{\max} = 34$.

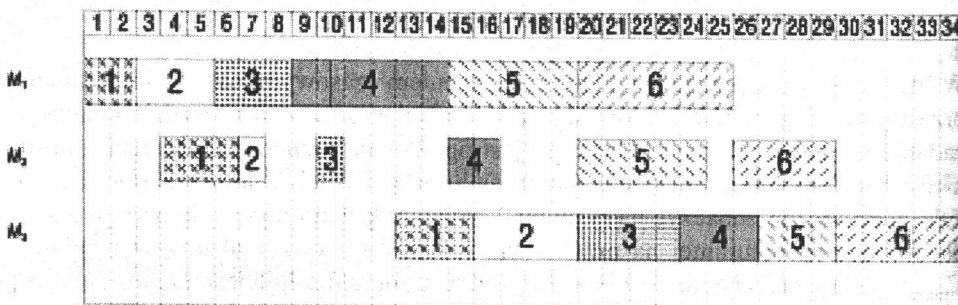
	5	6
5	5	6
5	5	4
3	3	5

Przez \hat{d}_j oznaczymy najdłuższy możliwy czas przestoju po wykonaniu operacji na maszynie M_j . Takie sprzężenie będziemy nazywać ograniczonym przestojem (ang. *limited idle*). Ponownie następuje modyfikacja wzoru 5:

$$C_{\pi(i+1),j} \leq C_{\pi(i),j} + p_{\pi(i+1),j} + \hat{d}_j \quad (9)$$

Gdy wszystkie \hat{d}_j są równe 0, wtedy mówimy znowu o ograniczeniu bez przestoju, co widać na rysunku 2. Gdy na przykład z tablicy 1 i permutacji naturalnej przyjmujemy $\hat{d}_1 = 2$, $\hat{d}_2 = 4$ i $\hat{d}_3 = 0$, otrzymamy harmonogram pokazany na rysunku 3. W tym wypadku ponownie mamy $C_{\max} = 34$, mimo że na maszynie M_3 wymusiliśmy ciągłą pracę maszyn.

RYSUNEK 3. Problem przepływowi z ograniczonym przestojem



Źródło Opracowanie własne.

Okna czasowe maszyn

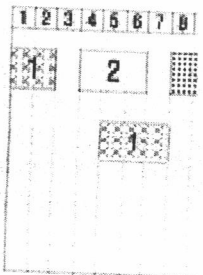
Ostatnie zaproponowane sprzężenie czasowe to ograniczenie z oknami czasowymi maszyn (ang. *time window of machines*). Znajduje zastosowanie, gdy ciągła praca maszyn nie jest wskazana, np. z powodu możliwości przegrzania urządzenia. Dlatego będziemy wymuszać przerwy w harmonogramie, przez dodanie najkrótszego możliwego czasu przestoju na maszynie M_j , oznaczonego jako \hat{r}_j . W połączeniu z najdłuższymi możliwymi czasami przestoju \hat{d}_j na maszynie M_j otrzymamy okno czasowe, w którym musi się rozpocząć się kolejna operacja. Oprócz wzoru 9 dojdzie nam kolejne ograniczenie wynikające z wymuszenia przestoju:

$$C_{\pi(i+1),j} \geq C_{\pi(i),j} + p_{\pi(i+1),j} + \hat{r}_j \quad (10)$$

Pamiętając, że:

$$\hat{r}_j \leq \hat{d}_j \quad (11)$$

przyjmijmy $\hat{r}_1 = 1$, $\hat{r}_2 = 2$, $\hat{r}_3 = 0$, dla permutacji naturalnej, danych z przykładu z tablicy 1, $\hat{d}_1 = 2$, $\hat{d}_2 = 4$ i $\hat{d}_3 = 0$ otrzymamy $C_{\max} = 39$, co widać na harmonogramie na rysunku 4.



Źródło: Opracowanie własne.

Problem przepływowi
podku budowy fundam
Szczególnym przykład
formowanych ciągle (a
terenach podmokłych. I
1. Wiercenie na żądar
2. Podciąganie świdra
pod ciśnieniem.
3. Usunięcie urobku v
4. Osadzenie zbrojeni

Każdy z etapów n.
 etapu, dlatego będzie
 maszynę M_2 w problemi
 wielotonowego sprzętu
 zaleca się wykonać proc
 Przed palowaniem
 niego wykopu, wykarcz
 tymczasowej drogi ze s
 meż odpowiednio zabez
 wawczego będziemy tra
 sząc przestojuów.

Kolejnym etapem ł
 maszynę M_3 . Najpierw n
 dowego. Następnie wykc
 5 cm powyżej warstwy b
 nie z projektem. Aby prz
 trzeba odczekać min. 6-

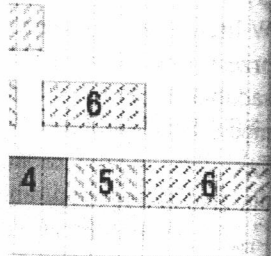
wykonaniu operacji na zonym przestoje (ang.

(9)

ograniczeniu bez prze- i permutacji naturalnej pokazany na rysunku 3. szynie M_3 wymusiliśmy

stojem

1 2 5 26 27 28 29 30 31 32 33 34

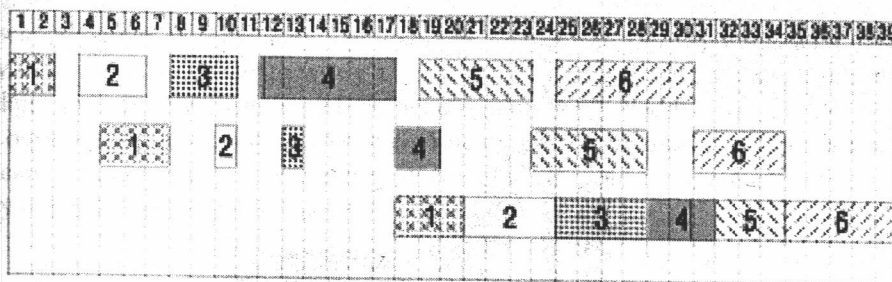


(10)

(11)

j, danych z przykładu widać na harmonogra-

RYSUNEK 4. Problem przepływowy z oknami czasowymi maszyn



Źródło: Opracowanie własne.

Analiza przypadku

Problem przepływowy ze sprzężeniami czasowymi znajduje zastosowanie w przypadku budowy fundamentów pod słupy sieci energetycznych wysokiego napięcia. Szczególnym przykładem są fundamenty palowane, wykonywane przy użyciu pali formowanych ciągle (ang. *continuous flight auger piles*), które stosuje się często na terenach podmokłych. Proces formowania pala można podzielić na 4 etapy.

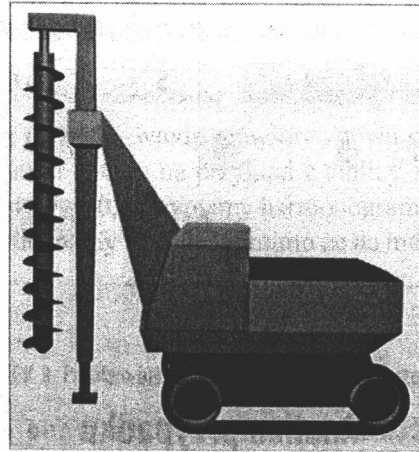
1. Wiercenie na żądaną głębokość.
2. Podciąganie świdra przy jednoczesnym pompowaniu przez rdzeń świdra betonu pod ciśnieniem.
3. Usunięcie urobku wyniesionego przez świder.
4. Osadzenie zbrojenia w betonie.

Każdy z etapów należy wykonać natychmiast po zakończeniu poprzedniego etapu, dlatego będziemy traktować całość jako jeden proces, czyli jako jedną maszynę M_2 w problemie przepływowym. Proces wymaga użycia specjalistycznego, wielotonowego sprzętu (dźwigu), jak na rysunku 5, co generuje duże koszty, dlatego zaleca się wykonać proces palowania na wszystkich stanowiskach bez przestojów.

Przed palowaniem należy przygotować stanowisko poprzez wykonanie płytkiego wykopu, wykarczowanie przejazdu lub w szczególnych przypadkach ułożenie tymczasowej drogi ze specjalistycznych płyt. Oczywiście stanowisko należy również odpowiednio zabezpieczyć przed osobami trzecimi. Całość procesu przygotowawczego będziemy traktować jako maszynę M_1 , na której nie ma potrzeby wymuszać przestojów.

Kolejnym etapem będzie przygotowanie głowic pali, które oznaczymy jako maszynę M_3 . Najpierw należy pogłębić wykop do poziomu położenia betonu podkładowego. Następnie wykonać warstwę z betonu podkładowego. Rozkuć głowice pala 5 cm powyżej warstwy betonu podkładowego oraz ustawić pręty zbrojeniowe zgodnie z projektem. Aby przystąpić do etapu wykonywanego na maszynie M_3 , najpierw trzeba odczekać min. 6–7 dni w celu osiągnięcia przez beton odpowiedniej wytrzy-

RYSUNEK 5. Dźwig terenowy ze świdrem CFA



Źródło: Opracowanie własne.

małości. Jednak nie można czekać dłużej niż 9–10 dni, ponieważ wtedy beton osiągnie zbyt dużą wytrzymałość i przy kuciu może dojść do rozległych pęknięć. Dlatego trzeba zrobić to w oknie czasowym najlepiej między 6. a 8. dniem.

Czwartym procesem, M_4 będzie przygotowanie płyty fundamentowej. Najpierw należy wykonać zbrojenie płyty, łącząc ją ze zbrojeniem pali. Kolejnym krokiem jest wykonanie szalunku i zalanie całości odpowiednią mieszanką betonu. Po tym etapie należy odczekać około 21 dni, zanim przystąpi się do kolejnego etapu. Tym razem nie ma ograniczonego czekania.

Ostatnim etapem jest rozszalowanie fundamentu oraz niwelacja terenu wokół stanowiska. Jeśli była ułożona droga tymczasowa, należy ją rozebrać. Proces ten oznaczymy jako maszynę M_5 .

Algorytmy

Dla wszystkich omówionych sprzężeń czasowych zaimplementowano jeden algorytm dokładny, bazujący na metodzie podziału i ograniczeń [Smutnicki 2012] (ang. *branch and bound*) oraz po dwa algorytmy przybliżone: symulowane wyżarzanie (ang. *simulated annealing*) [van Laarhoven, Aarts 1987] oraz przeszukiwanie z zabronieniami (ang. *tabu search*) [Glover 1989; Glover 1990].

Metoda podziału i ograniczeń

Metoda podziału i ograniczeń opiera się na przeglądaniu drzewa stanu, które reprezentuje wszystkie możliwe ścieżki, prowadzące do wszystkich rozwiązań zapisanych w liściach. Przejrzenie wszystkich ścieżek i liści wymaga zbyt dużego nakładu

obliczeniowego, dlatego (ang. *lower bound*) [Gov celu w poddrzewie. W on lepszy (ang. *best first*), dla kolejce priorytetowej dolne ograniczenie jest g odrzucane i traktowane j niemy najlepsze rozwiąz

Algorytm symulowanego sąsiedztwa, inspirowaną p wego i wysokiej tempera zanie, jest podatne na zm zadań w aktualnym rozw jemy je, ale z pewnym p wiązanie. Po każdym kro tury zmniejsza się szansa kończy się, gdy temperat według zadanego współc powtórzyć operację poszu

P

Algorytm przeszukiwania (w tym wypadku typu mum lokalnym stosuje się machów, które doprowadz mentuje się w postaci ma

E

W celu zbadania efektyw sprzężeń czasowych na wyniki z metodą dokładr PC, wyposażonym w pro i dysk SSD pracujący poc

Algorytmy testowane ograniczenia metody poc ograniczeń z limitowany: zastosowano procentowe

obliczeniowego, dlatego algorytm dla każdego węzła wyznacza dolne ograniczenie (ang. *lower bound*) [Gowerishankar, Rajendran, Srinivasan 2001] wartości funkcji celu w poddrzewie. W omawianym przykładzie zastosowano strategię najpierw najlepszy (ang. *best first*), dlatego węzły z częściowym rozwiązaniem są umieszczane w kolejce priorytetowej według ich dolnego ograniczenia. Poddrzewa, dla których dolne ograniczenie jest gorsze niż pewne górne ograniczenie (ang. *upper bound*), są odrzucane i traktowane jako przejrane pośrednio. Przez górne ograniczenie rozumiemy najlepsze rozwiązanie znalezione do tej pory.

Symulowane wyżarzanie

Algorytm symulowanego wyżarzania jest probabilistyczną techniką przeszukiwania sąsiedztwa, inspirowaną procesem metalurgicznym. Zaczynamy od rozwiązania losowego i wysokiej temperatury początkowej, kiedy materiał, w tym wypadku rozwiązanie, jest podatny na zmiany. W każdej iteracji generujemy losową zamianę dwóch zmiennych w aktualnym rozwiązaniu. Gdy nowe rozwiązanie jest lepsze, wtedy akceptujemy je, ale z pewnym prawdopodobieństwem możemy zaakceptować gorsze rozwiązanie. Po każdym kroku temperatura jest obniżana. Wraz ze spadkiem temperatury zmniejsza się szansa na zaakceptowanie gorszego rozwiązania. Praca algorytmu kończy się, gdy temperatura materiału osiągnie temperaturę końcową, która spada według zadanego współczynnika. W celu poprawy wydajności tej metody możemy powtórzyć operację poszukiwania zadaną liczbę razy dla każdej wartości temperatury.

Przeszukiwanie z zabronieniami

Algorytm przeszukiwania z zabronieniami wybiera najlepsze rozwiązanie z sąsiedztwa (w tym wypadku typu „zamień”). W celu uniknięcia utknięcia algorytmu w minimum lokalnym stosuje się listę zakazów (ang. *tabu list*), gdzie umieszczamy atrybuty ruchów, które doprowadziły do aktualnego rozwiązania. Najczęściej listę tabu implementuje się w postaci macierzy lub tablicy.

Eksperymenty obliczeniowe

W celu zbadania efektywności algorytmu heurystycznego i sprawdzenia wpływu sprzężeń czasowych na jego jakość, przeprowadzono serię testów i porównano wyniki z metodą dokładną. Badania zostały przeprowadzone na komputerze klasy PC, wyposażonym w procesor Intel i7-6700K o taktowaniu 4,00 GHz, 16GB RAM i dysk SSD pracujący pod systemem operacyjnym Microsoft Windows 10.

Algorytmy testowano na przykładach, które wygenerowano, biorąc pod uwagę ograniczenia metody podziału i ograniczeń oraz brak przykładowych danych dla ograniczeń z limitowanym przestojem lub oknami czasowymi. Jako miarę jakości zastosowano procentowe względne odchylenie (*PRD*).

$$PRD(\pi) = \frac{100\% (C_{max}(\pi) - C_{max}(\pi^{ref}))}{C_{max}(\pi^{ref})}, \quad (12)$$

gdzie:

- π — znalezione rozwiązanie,
 π^{ref} — rozwiązanie referencyjne,
 $C_{max}(\pi)$ — wartość funkcji celu dla danego rozwiązania.

Jako algorytm referencyjny wybrano metodę podziału i ograniczeń. Następnie przetestowano algorytm symulowanego wyżarzania i poszukiwania z zabronieniami z parametrem 5 tys. liczby iteracji. Aby oba algorytmy miały przybliżony czas działania w symulowanym wyżarzaniu; dla każdej temperatury powtórzono szukanie sąsiada $\frac{n^2}{3}$ razy. W celu wyeliminowania błędów algorytmy przybliżone uruchomiono po 50 razy dla wszystkich przykładów, sprawdzając wszystkie zaproponowane sprzężenia czasowe, następnie wyniki uśredniono i zebrano w tablicach 2–5, gdzie:

- n — liczba zadań,
 m — liczba maszyn,
 t_{BB} — czas pracy algorytmu podziału i ograniczeń,
 t_{SA} — czas pracy algorytmu symulowanego wyżarzania,
 t_{TS} — czas pracy algorytmu przeszukiwania z zabronieniami,
 PRD_{SA} — procentowe względne odchylenie algorytmu symulowanego wyżarzania,
 PRD_{TS} — procentowe względne odchylenie algorytmu przeszukiwania z zabronieniami.

W tablicy 2 umieszczono wyniki dla klasycznego problemu przepływowego dla wygenerowanych 80 zestawów danych testowych, po 10 sztuk dla danej wielkości problemu. Łącznie przebadano 8 różnych rozmiarów instancji, zmieniając liczbę zadań i maszyn. Pliki zostały zapisane w formacie zaproponowanym przez Taillarda [1993]. Przez złożoność metody podziału i ograniczeń oraz funkcji liczącej granice pomiary dokonano maksymalnie dla 16 zadań na 10 maszynach.

TABLICA 2. Wyniki dla problemu przepływowego

Nazwa	Wymiar $n \times m$	Pomiary				
		t_{BB} [ms]	t_{SA} [ms]	t_{TS} [ms]	PRD_{SA} [%]	PRD_{TS} [%]
ri001-ri010	10×5	16,20	33,02	24,49	0,31	0,00
ri011-ri020	10×10	38,50	54,53	49,07	1,57	0,00
ri021-ri030	12×5	291,50	52,85	42,43	0,60	0,01
ri031-ri040	12×10	364,30	89,99	86,77	1,52	0,01
ri041-ri050	14×5	1 736,9	77,37	66,47	0,58	0,01

Nazwa	Wymiar
	$n \times m$
ri051-ri060	14×10
ri061-ri070	16×5
ri071-ri080	16×10

Zródło: Opracowanie własne.

W drugim etapie 1 sycznego problemu prz przestojów. Wyniki zan gramu i bardziej złożo 12 zadań na 10 maszyn

TABLICA

Nazwa	Wymiar
	$n \times m$
ri001-ri010	10×5
ri011-ri020	10×10
ri021-ri030	12×5
ri031-ri040	12×10

Zródło: Opracowanie własne.

W kolejnym etapie wygenerować najdłuższ przetestować problem p ilości problemu i wyniki

TABLICA 4. W

Nazwa	Wymiar
	$n \times m$
ri001-ri010	10×5
ri011-ri020	10×10
ri021-ri030	12×5

$\frac{ef}{\dots}$, (12)

Nazwa	Wymiar	Pomiary				
	$n \times m$	t_{BB} [ms]	t_{SA} [ms]	t_{TS} [ms]	PRD_{SA} [%]	PRD_{TS} [%]
051-ri060	14×10	5 948,30	136,20	139,60	1,50	0,16
061-ri070	16×5	4 529,20	108,20	97,97	0,34	0,05
071-ri080	16×10	10 853,30	194,36	208,30	1,51	0,31

Źródło: Opracowanie własne.

tu i ograniczeń. Następnie zukiwania z zabronieniami przybliżony czas działy przybliżone uruchomiono wszystkie zaproponowane w tablicach 2–5, gdzie

W drugim etapie testów użyto tych samych plików wejściowych, co dla klasycznego problemu przepływowego, następnie przebadano sprzężenie czasowe bez przestojów. Wyniki zamieszczono w tablicy 3. Ze względu na rozsunięcie harmonogramu i bardziej złożoną funkcję celu maksymalnie zbadano instancje o wielkości 12 zadań na 10 maszynach.

TABLICA 3. Wyniki dla problemu przepływowego bez przestoju

Nazwa	Wymiar	Pomiary				
	$n \times m$	t_{BB} [ms]	t_{SA} [ms]	t_{TS} [ms]	PRD_{SA} [%]	PRD_{TS} [%]
001-ri010	10×5	289,50	80,30	86,06	0,89	0,00
011-ri020	10×10	649,00	138,22	163,4	2,55	0,00
021-ri030	12×5	4 010,60	131,59	143,38	0,90	0,00
031-ri040	12×10	69 680,80	352,49	412,74	2,15	0,00

Źródło: Opracowanie własne.

ania, nieniami, symulowanego wyżarzania przeszukiwania z zabronieniami. Problem przepływowego dla 10 sztuk dla danej wielkości instancji, zmieniając liczbę proponowanym przez Taillandiera oraz funkcji liczącej granice maszynach.

W kolejnym etapie badań do wcześniej użytych danych wejściowych należało wygenerować najdłuższe możliwe czasy przestojów na maszynach ($\hat{d}_j \in [0, 9]$), aby przetestować problem przepływowy z ograniczonym czekaniem. Przebadano 6 wielkości problemu i wyniki przedstawiono w tablicy 4.

owego

	PRD_{SA} [%]	PRD_{TS} [%]
1)	0,31	0,00
7)	1,57	0,00
3)	0,60	0,01
7)	1,52	0,01
7)	0,58	0,01

TABLICA 4. Wyniki dla problemu przepływowego z limitowanym przestojem

Nazwa	Wymiar	Pomiary				
	$n \times m$	t_{BB} [ms]	t_{SA} [ms]	t_{TS} [ms]	PRD_{SA} [%]	PRD_{TS} [%]
001-ri010	10×5	101,70	89,53	98,74	1,44	0,00
011-ri020	10×10	567,70	182,81	211,84	2,74	0,00
021-ri030	12×5	464,40	144,57	164,38	1,27	0,00

cd. tablicy 4

Nazwa	Wymiar	Pomiary				
	$n \times m$	t_{BB} [ms]	t_{SA} [ms]	t_{TS} [ms]	PRD_{SA} [%]	PRD_{TS} [%]
ri031-ri040	12×10	31 774,30	320,63	382,19	2,41	0,00
ri041-ri050	14×5	89 053,90	234,09	273,88	1,20	0,04
ri051-ri060	14×10	11 4069,13	495,22	606,86	3,90	0,04

Źródło: Opracowanie własne.

Na końcu przetestowano problem przepływowy z oknami czasowymi. Do już istniejących danych wejściowych należało wygenerować najkrótsze możliwe czasy przestoju na maszynach (\hat{r}_j), pamiętając, że $\hat{r}_j \leq \hat{d}_j$. Uśrednione wyniki dla 60 zestawów wejściowych przedstawiono w tablicy 5.

TABLICA 5. Wyniki dla problemu przepływowego z oknami czasowymi na maszynie

Nazwa	Wymiar	Pomiary				
	$n \times m$	t_{BB} [ms]	t_{SA} [ms]	t_{TS} [ms]	PRD_{SA} [%]	PRD_{TS} [%]
ri001-ri010	10×5	85,80	122,80	132,09	1,51	0,00
ri011-ri020	10×10	528,60	244,22	273,95	2,72	0,00
ri021-ri030	12×5	402,90	197,45	220,90	1,02	0,00
ri031-ri040	12×10	33 109,60	437,10	505,73	2,58	0,01
ri041-ri050	14×5	111 949,40	323,94	379,40	0,98	0,01
ri051-ri060	14×10	161 143,75	719,27	873,82	3,78	0,02

Źródło: Opracowanie własne.

Podsumowanie

W pracy przedstawiono problem szeregowania z praktyki budowlanej z ograniczeniem w postaci sprzężeń maszyn, z ograniczonym przestojem i z oknami czasowymi maszyn. Czas działania wszystkich prezentowanych algorytmów wzrósł w porównaniu z wersją bez ograniczeń na co wpływ miały fakt, że funkcje celu wymagały większego nakładu obliczeń oraz nie było możliwości wykorzystania tzw. akceleratorów, stosowanych dla klasycznych problemów przepływowych. Mimo wydłużenia czasu działania implementacji zaproponowanych algorytmów jakość otrzymanych wyników się nie pogorszyła, chociaż dodano kolejne ograniczenia.