

Marcin KLIMEK
Państwowa Szkoła Wyższa w Białej Podlaskiej
Piotr ŁEBKOWSKI
AGH Akademia Górniczo-Hutnicza w Krakowie

HARMONOGRAMOWANIE WSTECZNE DLA PROJEKTU ROZLICZANEGO ETAPOWO

Streszczenie. Artykuł prezentuje zagadnienie harmonogramowania projektu z ograniczonymi zasobami i maksymalizacją zdyskontowanych przepływów pieniężnych z punktu widzenia wykonawcy. Jako ujemne przepływy pieniężne uwzględnione są wydatki wykonawcy związane z wykonywaniem zadań. Wpływami są płatności klienta za zrealizowane etapy projektu. Do rozwiązania zagadnienia proponowana jest procedura harmonogramowania wstecznego przy uwzględnieniu umownych etapów projektu. Skuteczność tej procedury, wykorzystanej do generowania rozwiązań dla algorytmu symulowanego wyżarzania, jest sprawdzana przy użyciu standardowych zadań testowych z dodatkowo zdefiniowanymi przepływami pieniężnymi i umownymi etapami projektu.

BACKWARD SCHEDULING FOR A PROJECT SETTLED WITH STAGES

Summary. The article presents the resource-constrained project scheduling problem with the maximisation of discounted cash flows from the contractor's perspective. As the outflows are included expenses related to the contractor performing activities. Inflows are customer payments for completed stages of the project. To solve the problem, the proposed is backward scheduling procedure taking into account the contractual stages of the project. The effectiveness of the procedure, used to generate solutions for the simulated annealing algorithm, is tested using standard test instances of further defined cash flows and contractual stages of the project.

1. Wprowadzenie

Harmonogramowanie projektu z ograniczoną dostępnością zasobów RCPSP (ang. *Resource Constrained Project Scheduling Problem*) jest przedmiotem badań od wielu lat ze względu na praktyczną przydatność tego zagadnienia. Analizowane są liczne modele optymalizacyjne, z różnymi kategoriami zasobów i różnymi sposobami wykonywania zadań (Hartmann i Briskorn [8]). Jako kryteria optymalizacji najczęściej rozpatrywane są kryteria czasowe np. minimalizacja czasu trwania projektu lub

kryteria finansowe np. maksymalizacja zdyskontowanych przepływów pieniężnych (problem RCPSP-DC – *RCPSP with Discounted Cash flows*).

Przy planowaniu projektu w praktyce istotna jest analiza skutków finansowych podejmowanych decyzji. W związku z tym w wielu pracach badawczych przy harmonogramowaniu prac uwzględniane są przepływy pieniężne związane z realizowanym projektem. Dla projektów długoterminowych przepływy te są dyskontowane – obliczana jest ich bieżąca wartość NPV (ang. *Net Present Value*) przy przyjętej stopie dyskontowej. Jako pierwszy ideę maksymalizacji NPV dla problemu harmonogramowania projektu wprowadził Russell w 1970 r. – model Max-NPV (Russell [26]). Początkowe prace badawcze z optymalizacją NPV nie uwzględniały ograniczeń zasobowych dla projektu (Russell [26], Smith-Daniels i in. [27]). Aktualnie podejmowany jest głównie problem maksymalizacji zdyskontowanych przepływów pieniężnych dla projektu z ograniczoną dostępnością zasobów RCPSP-DC (Mika i in. [24], Vanhoucke [35], Klimek i Łebkowski [15÷19], He i Xu [10], Fink i Homberger [7], Khoshjahan i in. [13], Thiruvady i in. [29] i wiele innych). Przegląd modeli i algorytmów z optymalizacją przepływów pieniężnych można znaleźć w pracach (Hartmann i Briskorn [8], Herroelen i in. [12]). W tej pracy opisane są jedynie wybrane zagadnienia RCPSP-DC, które są istotne z punktu widzenia analizowanego modelu.

Przy obliczaniu wartości NPV dla projektu uwzględniane są wszystkie przepływy pieniężne z nim związane, zdyskontowane w celu uwzględnienia zmiany wartości pieniądza w czasie. Optymalizacji podlega m.in. harmonogram płatności za zrealizowane zadania lub etapy projektu. Analizowane są różne modele rozliczeń między klientem a wykonawcą w ramach harmonogramowania płatności projektowych PPS (ang. *Payment Project Scheduling*) (Bahrami i Moslehi [1], Leyman i Vanhoucke [23], Dayanand i Padman [5], Mika i in. [24], Ulusoy i in. [33]). Poszukiwany jest harmonogram wykonywania płatności z maksymalną wartością NPV projektu z perspektywy wykonawcy i/lub klienta. Ustaleniu podlega np. całkowita kwota płatności klienta za wykonanie przedsięwzięcia, liczba transz płatności, kwoty w poszczególnych transzach i terminy ich wykonywania itp. Kwoty i terminy kolejnych transz mogą brać pod uwagę np. koszty realizacji czynności, stopień zaawansowania prac projektowych itp. Występują rozbieżne interesy klienta i wykonawcy dotyczące rozliczeń projektu: dla wykonawcy korzystne są jak najwcześniejsze wpłaty klienta (większe NPV płatności), natomiast dla klienta opłacalne jest odraczanie płatności. Problem PPS jest rozważany z punktu widzenia wykonawcy (Mika i in. [24], Ulusoy i in. [33]) lub klienta (Dayanand i Padman [5]). Poszukiwane są również rozwiązania satysfakcjonujące zarówno klienta, jak i wykonawcę (Bahrami i Moslehi [1], Ulusoy i Cebelli [31]).

W tej pracy rozpatrywany jest problem PPS z punktu widzenia wykonawcy, dla którego w badaniach RCPSP-DC analizowane są przepływy dodatnie (ang. *cash inflows*) oraz przepływy ujemne (ang. *cash outflows*). Przepływy ujemne to wydatki wykonawcy najczęściej związane z wykonywaniem zadań i z wykorzystywaniem zasobów. Przepływy dodatnie to z reguły płatności klienta na rzecz wykonawcy za zrealizowany projekt lub jego etapy, za wykonane zadania. Rozpatrywane są różne modele wykonywania płatności (Leyman i Vanhoucke [23], Mika i in. [24], Ulusoy i in. [33]):

- model LSP (ang. *Lump-Sum Payment*) – płatności klienta wykonywane są jednokrotnie po ukończeniu prac projektowych,
- model PEO (ang. *Payments at Event Occurences*) – płatności występują dla zdarzeń tj. ukończenie etapu projektu lub zakończenie czynności – model PAC (ang. *Payments at Activities' Completion times*),
- model ETI (ang. *Equal Time Intervals*) – płatności wykonywane są w stałych odstępach czasowych przy wyznaczonej ich liczbie,
- model PP (ang. *Progress Payments*) – płatności wykonywane są w stałych odstępach czasowych przy nieokreślonej ich liczbie.

Przy rozliczaniu projektu między klientem a wykonawcą analizowany jest również model, w którym definiowany jest system kar i nagród (ang. *bonus-penalty system*) (He i Xu [10], Khoshjahan i in. [13], Klimek i Łebkowski [15÷19]). Kary stosowane są za przekroczenie przez wykonawcę umownego terminu ukończenia projektu lub jego etapu, nagrody za wcześniejszą realizację projektu lub jego etapu itp. Określone są dokładne terminy zakończenia prac lub okna czasowe, w których realizacja czynności (etapu) nie jest ani nagradzana, ani karana. Analizowane jest zagadnienie (Khoshjahan i in. [13]), w którym dla każdego z zadań określony jest termin jego realizacji w celu minimalizacji NPV kosztów wykonania zadań przed i po żądanym terminie (ang. *earliness-tardiness penalties*). Rozważane są różne systemy rozliczeń między wykonawcą i klientem z karami i nagrodami za realizację prac (He i Xu [10], Klimek i Łebkowski [15÷19]). System kar i nagród wprowadza się w taki sposób, aby zmobilizować wykonawcę do jak najszybszej realizacji prac. Brak dodatkowych bodźców do szybszego wykonywania prac przez wykonawców, może prowadzić do późniejszej realizacji przedsięwzięcia. Aby system kar i nagród był efektywny z perspektywy wykonawcy, korzyści z nagród powinny być wyższe niż koszty wykonawcy ponoszone w związku z szybszym wykonaniem zadań (etapów) a kary za nieterminowość powinny być wyższe niż korzyści z późniejszej realizacji czynności (etapów). Aby system kar i nagród był efektywny z perspektywy klienta, premia dla wykonawcy za przyspieszone wykonanie czynności (etapów) nie powinna być wyższa niż profit klienta z przyspieszonej realizacji zadań (etapów) a kara za nieterminowe wykonywanie prac powinna być wyższa niż utracone korzyści klienta z terminowej realizacji czynności (etapów). Z różnic w preferencjach klienta i wykonawcy, które dotyczą np. terminów wykonania i wysokości poszczególnych transz płatności za realizację przedsięwzięcia, może wynikać przydatność systemu kar i nagród w praktyce.

W tej pracy proponowany jest etapowy system rozliczeń, w którym klienci ustalają z wykonawcą umowne etapy przedsięwzięcia i określają dla nich terminy realizacji oraz kwoty płatności za ukończenie prac w tych etapach. W przypadku opóźnień w realizacji etapów, stosowane są kary, które zmniejszają płatności klienta za zrealizowane prace. Korzyści z etapowego systemu rozliczeń dla wykonawcy, to możliwość otrzymywania wcześniejszych płatności za wykonane prace, które można przeznaczyć na bieżącą działalność (realizację nowych czynności, zakup materiałów itp.).

Etapowy system rozliczeń, ze zdefiniowanymi kamieniami milowymi (ang. *milestones*) dla przedsięwzięcia, może znaleźć praktyczne zastosowania. Podczas

realizacji projektów w praktyce stosowana jest technika kamieni milowych m.in. przy określaniu stopnia realizacji przedsięwzięcia na poszczególnych jego etapach (koniec etapu to kamień milowy) (Stabryła [28]). Kamieniami milowymi są „punkty koordynacyjne i kontrolne, mające szczególne znaczenie ze względu na: wagę cząstkowych rezultatów projektu dla rezultatów jego następnych etapów i kroków oraz dla rezultatu końcowego (tzw. odbiór końcowy)” (Trocki [30]). Określanie kamieni milowych jest istotnym elementem planowania przedsięwzięcia. Ich liczba zależy od specyfiki projektu, ale nie powinna być zbyt duża, aby była zachowana ich „wyjątkowość”. Harmonogramowanie projektu jest wspierane przez systemy komputerowe np. Microsoft Project, które często umożliwiają stosowanie techniki kamieni milowych, co ułatwia zarządzanie projektem i zwiększa kontrolę nad terminowością jego realizacji, może być także wykorzystane przy rozliczaniu się wykonawcy z klientem.

W badaniach z zakresu harmonogramowania projektu z ograniczonymi zasobami z optymalizacją zdyskontowanych przepływów pieniężnych problem z etapowymi rozliczeniami przedsięwzięcia w tej postaci, poza pracami autorów, nie był rozważany. Rozpatrywane są modele z etapowym rozliczaniem projektu dla problemu z wieloma sposobami wykonywania zadań MMRCPS (ang. *Multi-Mode RCPS*) (He i Xu [10], He i in. [11]).

W tej pracy analizowany jest problem harmonogramowania projektu z ograniczoną dostępnością zasobów RCPS oraz ze zdefiniowanymi umownymi etapami projektu z jednym sposobem wykonania czynności (ang. *Single-Mode RCPS*) z etapowymi rozliczeniami finansowymi (Klimek i Łebkowski [15-19]), w którym funkcją celu jest maksymalizacja sumy zdyskontowanych przepływów pieniężnych: z płatnościami klienta za zrealizowane etapy przedsięwzięcia, z systemem kar za nieterminową realizację umownych etapów projektu oraz z wydatkami wykonawcy ponoszonymi w związku z wykonywanymi zadaniami.

Celem opracowania jest przedstawienie nowej procedury harmonogramowania wstecznego z optymalizacją terminów realizacji umownych etapów projektu i pokazanie jej skuteczności dla analizowanego modelu z etapowymi rozliczeniami projektu. W celu ilustracji zagadnienia i procedury zaprezentowany jest przykład obliczeniowy. Na koniec zaprezentowane są wyniki eksperymentów obliczeniowych dla problemów testowych z biblioteki PSPLIB (ang. *Project Scheduling Problem LIBrary*) (Kolisch i Sprecher [22]), z dodatkowo zdefiniowanymi przepływami pieniężnymi i umownymi etapami rozliczeń.

2. Sformułowanie problemu

Rozpatrywane jest zagadnienie harmonogramowania projektu z ograniczonymi zasobami, w którym zadania (czynności) są niepodzielne i realizowane są tylko jednym, określonym sposobem (ang. *nonpreemptive single-mode RCPS*). Projekt jest przedstawiany jako acykliczny, skierowany graf $G(V, E)$ w reprezentacji *AON* (ang. *Activity On Node*), w którym V to zbiór węzłów reprezentujących zadania, natomiast E to zbiór łuków opisujących zależności kolejnościowe między zadaniami.

Analizowany jest autorski model maksymalizacji NPV z etapowym rozliczaniem prac: z przepływami pieniężnymi określonymi dla zadań i umownych etapów projektu (Klimek i Łebkowski [15÷19]). Model ten można sformułować następująco (wzory 1÷5):

$$\text{Maksymalizacja } F = \sum_{i=1}^{N_A} (CFA_i \cdot e^{-\alpha \cdot ST_i}) + \sum_{m=1}^{N_M} (CFM_m \cdot e^{-\alpha \cdot MT_m}) \quad (1)$$

przy następujących ograniczeniach:

$$ST_i + d_i \leq ST_j \quad \forall (i, j) \in E \quad (2)$$

$$\sum_{i \in J(t)} r_{ik} \leq R_k, \quad \forall t, \forall k \quad (3)$$

i dla zdefiniowanych umownych etapów projektu:

$$MT_m = \max_{i \in MA_m} (FT_i) \quad (4)$$

$$CFM_m = MP_m - MC_m \cdot \max(MT_m - MD_m, 0) \quad (5)$$

gdzie:

- N_A – liczba czynności projektowych,
- i – indeks (numer) czynności projektowej, $i = 1, \dots, N_A$,
- N_M – liczba umownych etapów projektu,
- m – indeks (numer) etapu projektu, $m = 1, \dots, N_M$,
- d_i – czas trwania czynności i ,
- α – stopa dyskontowa,
- ST_i – planowany (w bieżącym harmonogramie) czas rozpoczęcia czynności i ,
- FT_i – planowany (w bieżącym harmonogramie) czas zakończenia czynności i ($FT_i = ST_i + d_i$),
- k – indeks (numer) typu zasobu, $k = 1, \dots, K$ (K – liczba typów zasobów),
- R_k – liczba dostępnych zasobów typu k w każdym momencie trwania projektu,
- r_{ik} – liczba wykorzystywanych zasobów typu k przy realizacji czynności i ,
- $J(t)$ – zbiór czynności realizowanych w przedziale czasu $[t-1, t]$,
- CFA_i – wydatki wykonawcy ponoszone przy realizacji czynności i w momencie jej rozpoczynania,
- CFM_m – płatności klienta za realizację m -tego etapu projektu wyznaczone dla bieżącego harmonogramu (wpływy z punktu widzenia wykonawcy),
- MD_m – umowny termin zakończenia etapu m ,
- MT_m – planowany (w bieżącym harmonogramie) termin zakończenia etapu m ,
- MA_m – zbiór czynności realizowanych w etapie m z określonym umownym terminem ich zakończenia równym MD_m ,
- MP_m – umowna kwota płatności klienta za realizację m -tego etapu projektu,
- MC_m – umowny jednostkowy koszt opóźnień ponoszony przy opóźnionej realizacji m -tego etapu projektu.

Jako rozwiązanie szukany jest wektor czasów rozpoczęcia czynności ST_i , dla którego funkcja celu F (patrz: wzór 1) przyjmuje wartość maksymalną przy uwzględnieniu:

- zależności kolejnościowych występujących między zadaniami typu koniec-początek bez zwłoki (patrz: wzór 2),
- ograniczoności odnawialnych zasobów wykorzystywanych do wykonywania zadań (patrz: wzór 3)
- rozliczeń finansowych związanych z umownymi etapami projektu (patrz: wzory 4÷5).

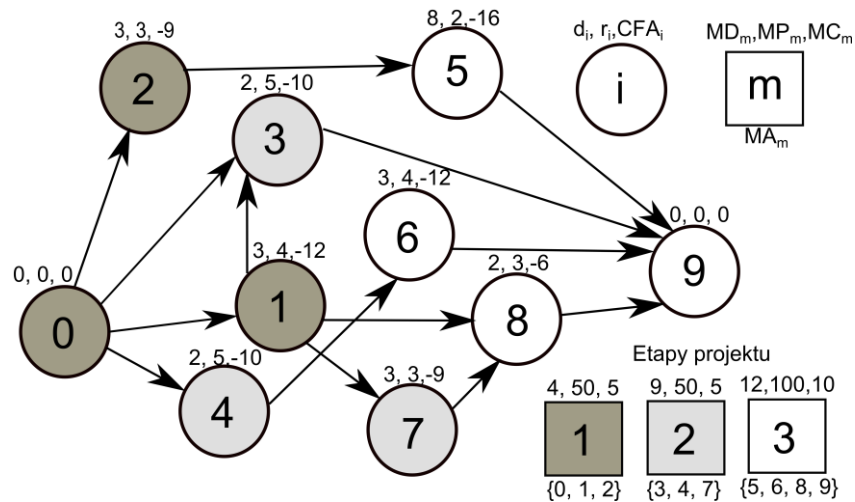
Przy rozliczeniach finansowych prac projektowych z perspektywy wykonawcy zakłada się, że wpływami (dodatnie przepływy pieniężne) dla wykonawcy są wpłaty klienta CFM_m (dla etapów $m = 1, \dots, N_M$), a wydatkami (ujemne przepływy pieniężne) są koszty realizacji zadań CFA_i (dla zadań $m = 1, \dots, N_A$).

Przy kalkulacji płatności klienta CFM_m (patrz: wzór 4) uwzględniane są umowne kwoty płatności MP_m za zrealizowane etapy projektu, które są pomniejszane w przypadku przekroczenia umownych terminów realizacji prac MD_m (kary za opóźnienia są wyznaczone przy uwzględnieniu umownego jednostkowego kosztu opóźnień MC_m). Przy wcześniejszym niż w umowie ukończeniu prac etapowych nie są przewidziane „nagrody” w postaci zwiększonych płatności klienta. W tym przypadku korzyścią dla wykonawcy jest wcześniejsze pozyskanie środków pieniężnych, o większej zdyskontowanej wartości. Założono, że płatności CFM_m są wykonywane przez zleceniodawcę dokładnie w momencie planowanego w bieżącym harmonogramie zakończenia prac MT_m (problem opóźnionych wpłat klienta jest pominięty w rozważanym modelu, jest niezależny od wykonawcy; można przyjąć, że klient jest zobligowany do finansowego zrekompensowania opóźnionych płatności).

Wszystkie wydatki wykonawcy są związane z kosztami realizacji czynności CFA_i wynikającymi np. z zaangażowania zasobów, materiałów, ich transportu itp. Wydatki z tego tytułu ponoszone są w terminie planowanego czasu rozpoczęcia czynności w bieżącym harmonogramie.

Proponowany model etapowych rozliczeń przedsięwzięcia jest korzystny dla wykonawcy, który otrzymuje w ten sposób od klienta środki finansowe na realizację zadań, zakup materiałów itp. przed ukończeniem prac projektowych. Klient wykonuje płatności przed zakończeniem projektu, co jest dla niego niekorzystne, ale z drugiej strony etapowe rozliczenia przedsięwzięcia wprowadzają kontrolę przebiegu prac projektowych w trakcie wykonywania projektu, a system kar umownych mobilizuje wykonawcę do terminowej realizacji etapów projektu.

W celu objaśnienia proponowanego zagadnienia przedstawmy przykład ilustracyjny. Niech projekt składa się z 8 zadań wykonywanych przy dostępności jednego typu zasobu wynoszącej 10. Sieć czynności AON dla projektu z zaprezentowanymi parametrami rozpatrywanego modelu optymalizacyjnego przedstawiona jest na rysunku 1. Zadania 0 i 10 to zadania pozorne (ang. *dummy activities*) reprezentujące odpowiednio wierzchołek początkowy i końcowy w grafie $G(V, E)$.



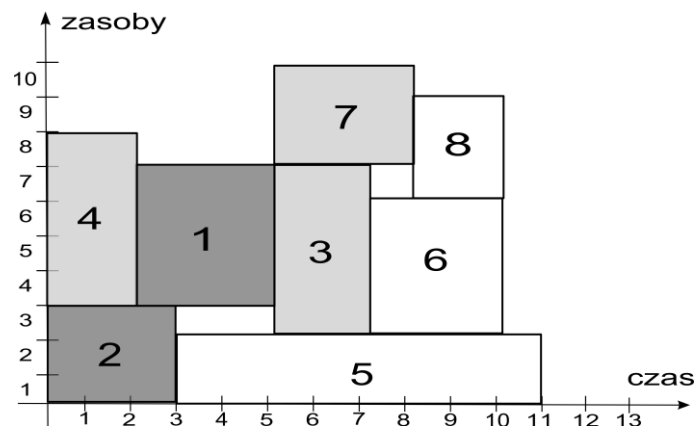
Rys. 1. Sieć czynności AON z umownymi etapami projektu

W projekcie określone są trzy umowne etapy rozliczeń:

- etap 1, w którym realizowane są zadania $MA_1 = \{0, 1, 2\}$, z umownym terminem realizacji $MD_1 = 4$, za który wykonywana jest płatność klienta równa $PM_1 = 50$, płatność ta pomniejszona jest o koszty ewentualnych opóźnień w pracach kalkulowanych na podstawie jednostkowego kosztu $CM_1 = 5$,
- etap 2, w którym realizowane są zadania $MA_2 = \{3, 4, 7\}$, z umownym terminem realizacji $MD_2 = 9$, za który wykonywana jest płatność klienta równa $PM_2 = 50$, płatność ta pomniejszona jest o koszty ewentualnych opóźnień w pracach kalkulowanych na podstawie jednostkowego kosztu $CM_2 = 5$,
- etap 3, kończący przedsięwzięcie, w którym realizowane są zadania $MA_3 = \{5, 6, 8, 9\}$, z umownym terminem realizacji $MD_3 = 12$, za który wykonywana jest płatność klienta równa $PM_3 = 100$, płatność ta pomniejszona jest o koszty ewentualnych opóźnień w pracach kalkulowanych na podstawie jednostkowego kosztu $CM_3 = 10$.

W obliczeniach zdyskontowanych wartości przepływów pieniężnych przyjęta jest stopa dyskontowa $\alpha = 0.01$ przy okresie kapitalizacji wynoszącym 1.

W celu wyjaśnienia sposobu wyznaczania funkcji celu F (patrz: wzór 1) posłużmy się przykładowym harmonogramem z minimalnym czasem trwania prac projektowych zaprezentowanym na rysunku 2.



Rys. 2. Przykładowy harmonogram dla analizowanego projektu

W harmonogramie z rysunku 2 planowane jest opóźnione ukończenie pierwszego etapu przedsięwzięcia ($MT_1 = 5$ przy umownym terminie $MD_1 = 4$) co uszczupla płatności klienta za ten etap o 5 jednostek pieniężnych. Pozostałe etapy ukończone są przed umownym terminem, co jest korzystne dla wykonawcy przez zwiększenie zdyskontowanej wartości płatności klienta. Zdyskontowane wpływy wykonawcy za zrealizowane etapy przedsięwzięcia wynoszą:

$$\sum_{m=1}^{N_M} (CFM_m \cdot e^{-\alpha \cdot MT_m}) = (50 - 5) \cdot e^{-0.05} + 50 \cdot e^{-0.08} + 100 \cdot e^{-0.11} = 178.54.$$

Zdyskontowane wydatki wykonawcy związane z wykonywaniem zadań wynoszą:

$$\begin{aligned} \sum_{i=1}^{N_A} (CFA_i \cdot e^{-\alpha \cdot ST_i}) &= -12 \cdot e^{-0.02} - 9 \cdot e^0 - 10 \cdot e^{-0.05} - 10 \cdot e^0 - 16 \cdot e^{-0.03} - 12 \cdot e^{0.07} + \\ &- 9 \cdot e^{-0.05} - 6 \cdot e^{-0.08} = -81.09. \end{aligned}$$

Łącznie funkcja celu jest równa:

$$F = \sum_{i=1}^{N_A} (CFA_i \cdot e^{-\alpha \cdot ST_i}) + \sum_{m=1}^{N_M} (CFM_m \cdot e^{-\alpha \cdot MT_m}) = -81.09 + 178.54 = 97.45.$$

W analizowanym modelu korzystne dla wykonawcy jest planowanie jego wydatków (związanych z realizacją zadań, wykonywanych w momencie ich rozpoczynania) jak najpóźniej, a pozyskiwanie wpłat klienta (wykonywanych po ukończeniu etapów projektu) jak najwcześniej. Wskazana jest strategia rozpoczynania zadań najpóźniej jak to możliwe *ALAP* (ang. *As Late As Possible*), przy uwzględnieniu etapów przedsięwzięcia, które powinny być realizowane najwcześniej jak to możliwe. Zgodnie z wiedzą autorów brak jest procedur, poza opracowywanymi przez autorów (Klimek i Łebkowski [15÷19]), które generują uszeregowania odpowiednie dla analizowanego modelu optymalizacyjnego.

3. Metody rozwiązania problemu

W algorytmach lokalnych poszukiwań dla problemu harmonogramowania projektu z ograniczonymi zasobami stosowane są reprezentacje pośrednie problemu takie jak wykorzystywana w tej pracy reprezentacja w postaci listy czynności (ang. *activity list*), w której rozwiązanie jest ciągiem numerów kolejnych zadań uwzględniającym relacje kolejnościowe. Rozwiązania w postaci listy czynności są dekodowane w wykonywalny harmonogram w reprezentacji bezpośredniej (czasy rozpoczęcia zadań) przy użyciu schematów generowania harmonogramu SGS (ang. *Schedule Generation Scheme*) (Kolisch [20]). Dla deterministycznego RCPSPP jako procedury dekodujące wykorzystywane są m. in.:

- szeregową (ang. *serial SGS*) – procedura, w której w kolejnych chwilach t ustalany jest czas rozpoczęcia dla pierwszej nieuszeregowanej czynności z listy czynności, w najwcześniejszym możliwym terminie przy uwzględnieniu ograniczeń kolejnościowych i zasobowych,
- równoległą (ang. *parallel SGS*) – procedura, w której w kolejnych chwilach t rozpoczynane są wszystkie nieuszeregowane czynności, analizowane w kolejności

wynikającej z listy czynności, które mogą być rozpoczęte w chwili t przy uwzględnieniu ograniczeń kolejnościowych i zasobowych.

Harmonogram przy użyciu SGS jest budowany w przód (ang. *forward scheduling*). Możliwe jest też zastosowanie harmonogramowania wstecznego (ang. *backward scheduling*), w którym procedury dekodujące ustalają czasy rozpoczęcia zadań „od końca” listy czynności, rozpoczynając planowanie od terminu przyjętego zakończenia projektu.

Dla problemu maksymalizacji zdyskontowanych ujemnych i dodatnich przepływów pieniężnych stosowane są procedury łączące techniki generowania uszeregowania wstecz i w przód (dwukierunkowy schemat generowania harmonogramu – Vanhoucke i in. [34]): zadania z przypisanymi sumarycznymi dodatnimi przepływami pieniężnymi planowane są jak najwcześniej, a zadania z przypisanymi sumarycznymi ujemnymi przepływami pieniężnymi planowane jak najpóźniej.

Poza wykorzystaniem przekształconych procedur dekodujących optymalizacja zdyskontowanych przepływów pieniężnych jest realizowana przez użycie procedur poprawy, w których przesuwane w prawo są zadania o sumarycznych ujemnych przepływach pieniężnych (dla rozwiązań tworzonych metodą harmonogramowania w przód) i/lub w lewo czynności o sumarycznych dodatnich przepływach pieniężnych (dla rozwiązań tworzonych metodą harmonogramowania wstecznego). W badaniach rozważane są różne procedury poprawy rozwiązań (przegląd procedur w pracy Vanhoucke [35]) przez przesunięcia zadań, jednak te procedury nie są dostosowane do analizowanego modelu z etapowymi rozliczeniami prac (patrz: wzory 1-5). Stosowane są dla modeli RCPS-DC, w których zadaniom przypisane są ujemne i/lub dodatnie przepływy pieniężne (Baroum i Patterson [2], Ulusoy i Özdamar [32], Pinder i Marucheck [25], Vanhoucke i in. [34], Vanhoucke [35]).

W modelu optymalizacyjnym z funkcją celu F (patrz: wzór 1) dodatnie przepływy pieniężne (płatności klienta za wykonanie etapów przedsięwzięcia otrzymywane dokładnie w terminie ich ukończenia) powinny być ustalone jak najwcześniej a ujemne przepływy pieniężne (wydatki wykonawcy ponoszone w momencie rozpoczynania czynności) jak najpóźniej. Korzystne jest jak najpóźniejsze rozpoczynanie zadań, przy jak najszybszym wykonywaniu kolejnych umownych etapów projektu. Zwiększenie funkcji celu jest osiągane zawsze przy odroczeniu w czasie zadań (z przypisanymi ujemnymi przepływami pieniężnymi), których opóźnione rozpoczęcie nie zmieni czasu zakończenia etapów przedsięwzięcia. Korzystne może być późniejsze zrealizowanie etapu projektu, jeśli przyniesie to większe profity z odroczonego wydatków ponoszonych na rozpoczynanie czynności. Dla rozważanego problemu autorzy analizują m. in.:

- użycie różnych procedur dekodujących harmonogramowania w przód i wstecz przy uwzględnieniu umownych etapów projektu,
- zastosowanie różnych algorytmów poprawy, w których dla harmonogramu generowanego w przód czynności przesuwane są w prawo lub dla harmonogramu generowanego wstecz czynności przesuwane są w lewo, przesunięcia zadań są wykonywane przy uwzględnieniu ograniczeń zasobowych i kolejnościowych, dla rozwiązań z ustaloną alokacją zasobów lub przy użyciu zmodyfikowanej procedury dekodującej SGS.

Dotychczasowe badania wskazują, że najlepsze rezultaty osiągnane są dla strategii harmonogramowania wstecznego, przy doborze najkorzystniejszych terminów realizacji umownych etapów projektu (Klimek i Łebkowski [15]). W tym artykule proponowana jest nowa procedura harmonogramowania wstecznego, która jest dedykowana dla analizowanego modelu optymalizacyjnego.

4. Procedura harmonogramowania wstecznego z optymalizacją terminów ukończenia umownych etapów projektu

Proponowana procedura generuje harmonogramy wstecz przy uwzględnieniu umownych etapów projektu. Rozwiązania są zapisywane w postaci listy czynności, która jest dekodowana przy użyciu odpowiednio przekształconego szeregowego lub równoległego SGS, uwzględniającego przyjęte terminy ukończenia etapów projektu. Zadania planowane są „od końca” listy czynności, od terminu zakończenia ostatniego etapu przedsięwzięcia. Dla każdego zadania ustalany jest taki czas jego rozpoczęcia, aby przy uwzględnieniu ograniczeń kolejnościowych i zasobowych, nie przekroczyć przyjętych terminów ukończenia etapów projektu MT_m (dla $m = 1, \dots, N_M$). Jeśli dla danej listy czynności i przyjętych MT_m , ustalony harmonogram jest niepoprawny (np. jedno z zadań ma przypisany „ujemny” czas rozpoczęcia), to jest on korygowany przez przesunięcie prawostronne wszystkich czynności aż do utworzenia poprawnego uszeregowania. Proponowana procedura harmonogramowania wstecznego z optymalizacją terminów ukończenia umownych etapów projektu przedstawiona jest na rysunku 3. W kolejnych jej przebiegach wykonywane są jednostkowe przesunięcia lewostronne etapów projektu aż do uzyskania harmonogramu o najwyższej wartości funkcji celu F (patrz: wzór 1).

```

FOR  $m := 1$  TO  $N_m$  DO
     $MT[m] := MD[m]$ ;
     $S^* := SGS(x, MT)$ ;
FOR  $m := 1$  TO  $N_m$  DO
BEGIN
     $MT[m] := MT[m] - 1$ ;
     $S := SGS(x, MT)$ ;
    IF ( $F(S) > F(S^*)$ ) THEN
BEGIN
         $S^* := S$ ;
         $m := m - 1$ ;           //wzrost  $F$ , dalsza poprawa w danym etapie
    END ELSE
         $MT[m] := MT[m] + 1$ ; //brak wzrostu  $F$ , cofnięcie przesunięcia o 1 etapu  $m$ 
END;
Zwróć harmonogram  $S^*$ 

```

gdzie: x – bieżąca analizowana lista czynności, MT – tablica przyjętych terminów realizacji etapów projektu MT_m (dla $m = 1, \dots, N_M$), $F(S)$ – wartość funkcji celu F dla harmonogramu S , $SGS(x, MT)$ – procedura generowania harmonogramu wstecz przy uwzględnieniu przyjętych terminów realizacji projektu MT dla rozwiązania (listy czynności) x , S^* – najlepszy aktualnie znaleziony harmonogram, o najwyższej wartości funkcji celu F , S – aktualnie analizowany harmonogram generowany przy użyciu procedury $SGS(x, MT)$.

Rys. 3. Procedura harmonogramowania wstecznego z optymalizacją terminów ukończenia umownych etapów projektu

Procedura przedstawiona na rysunku 3 jest usprawnieniem algorytmu generowania harmonogramu wstecz przy przyjęciu terminów realizacji etapów

projektu takich jak dla harmonogramu znalezionej w przód przy poszukiwaniu jak największej sumy zdyskontowanych płatności klienta CFM_m , bez uwzględniania wydatków ponoszonych na realizowanie zadań.

Procedura oparta jest na technice harmonogramowania wstecznego i stosuje schematy generowania *SGS* (szeregowy lub równoległy), które tworzą harmonogram analizując kolejne zadania „od końca” rozpatrywanej listy czynności przy uwzględnieniu przyjętych terminów realizacji etapów przedsięwzięcia MT . Przykładowe uszeregowanie wygenerowane dla listy czynności $x = \{1, 4, 2, 5, 6, 3, 8, 7\}$ i przyjętych terminów realizacji etapów projektu równych $MT_1 = MD_1 = 4$, $MT_2 = MD_2 = 9$, $MT_3 = MD_3 = 12$ przedstawiony jest na rysunku 3a. Uszeregowanie poprawione przy wykorzystaniu procedury przesunięć lewostronnych etapów projektu przy dobranych terminach ukończenia etapów przedsięwzięcia $MT_1 = 3$, $MT_2 = 7$, $MT_3 = 11$ zaprezentowany jest na rysunku 4b.



Rys. 4. Harmonogram wygenerowany wstecz przy użyciu szeregowego SGS

4a) przy założeniu $MT_1 = MD_1 = 4$, $MT_2 = MD_2 = 9$, $MT_3 = MD_3 = 12$,

4b) przy założeniu wcześniejszego ukończenia etapów projektu $MT_1 = 3$, $MT_2 = 7$, $MT_3 = 11$.

Harmonogram z rysunku 4a jest zbudowany procedurą wsteczną. Zadania są rozpoczynane najpóźniej jak to możliwe przy uwzględnieniu umownych terminów realizacji etapów projektu MD_m . Zdyskontowane płatności klienta za zrealizowane etapy przedsięwzięcia wynoszą:

$$\sum_{m=1}^{N_M} (CFM_m \cdot e^{-\alpha \cdot MT_m}) = 50 \cdot e^{-0.04} + 50 \cdot e^{-0.09} + 100 \cdot e^{-0.12} = 182.43.$$

Są one wyższe niż dla harmonogramu z rysunku 2, w związku z dotrzymaniem wszystkich umownych terminów realizacji etapów projektu. Dla harmonogramu z rysunku 2 w porównaniu z uszeregowaniem z rysunku 4a występują korzyści z wcześniejszych płatności za etap drugi i trzeci przedsięwzięcia (mają wyższą zdyskontowaną wartość), ale są one niższe niż naliczone koszty umowne za opóźnione ukończenie etapu pierwszego.

Zdyskontowane wydatki wykonawcy związane z wykonywaniem zadań wynoszą:

$$\sum_{i=1}^{N_A} (CFA_i \cdot e^{-\alpha \cdot ST_i}) = -12 \cdot e^{-0.01} - 9 \cdot e^{0.01} - 10 \cdot e^{-0.07} - 10 \cdot e^{0.05} - 16 \cdot e^{-0.04} - 12 \cdot e^{0.09} + \\ - 9 \cdot e^{-0.06} - 6 \cdot e^{-0.1} = -79.87.$$

Wydatki są niższe niż dla harmonogramu z rysunku 2, w związku z późniejszym rozpoczynaniem części zadań.

Łącznie funkcja celu jest równa:

$$F = \sum_{i=1}^{N_A} (CFA_i \cdot e^{-\alpha \cdot ST_i}) + \sum_{m=1}^{N_M} (CFM_m \cdot e^{-\alpha \cdot MT_m}) = -79.87 + 182.43 = 102.56.$$

Procedura poprawy rozwiązania uzyskanego dla procedury wstecznej polega na analizie „opłacalności” jednostkowych przesunięć w czasie planowanych terminów ukończenia etapów projektu MT_m , zaczynając od pierwszego a kończąc na ostatnim etapie. Dla listy czynności $x = \{1, 4, 2, 5, 6, 3, 8, 7\}$ przebieg proponowanej procedury przesunięć lewostronnych etapów projektu wygląda następująco:

- **przesuwanie lewostronne etapu 1**, przy przyjęciu przesuniętego o jednostkę terminu realizacji etapu $MT_1 = 3$ zwiększa się wartość funkcji celu F z 102.56 na 102.83 dzięki zwiększonej zdyskontowanej płatności za etap 1, nie jest możliwe ukończenie etapu 1 w terminie $MT_2 = 2$ ze względu na czasy trwania zadań realizowanych w tym etapie ($d_1 = 3, d_2 = 3$), algorytm przechodzi do poprawy harmonogramu w etapie 2;
- **przesuwanie lewostronne etapu 2**, przy założeniu przesuniętego o jednostkę terminu realizacji etapu $MT_2 = 8$ zwiększa się wartość funkcji celu F z 102.83 na 103.01 dzięki zwiększonej zdyskontowanej płatności za etap 2, przy przyjęciu przesuniętego o dwie jednostki terminu realizacji etapu $MT_2 = 7$ zwiększa się wartość funkcji celu F z 103.01 na 103.20, nie jest możliwe ukończenie etapu 2 w terminie $MT_2 = 6$, ponieważ przy takim terminie czas rozpoczęcia zadania 1 lub 2 ustalony procedurę wsteczną jest „ujemny”, algorytm przechodzi do poprawy harmonogramu w etapie 3;
- **przesuwanie lewostronne etapu 3**, przy przyjęciu przesuniętego o jednostkę terminu realizacji etapu $MT_3 = 11$ zwiększa się wartość funkcji celu F z 103.20 na 103.77 dzięki zwiększonej zdyskontowanej płatności za etap 3, nie jest możliwe ukończenie etapu 3 w terminie $MT_3 = 10$, ponieważ przy takim terminie czas rozpoczęcia zadania 1 lub 2 ustalony procedurę wsteczną jest „ujemny”, algorytm kończy działanie.

Po wykonaniu przesunięć lewostronnych w czasie etapów projektu powstaje harmonogram zaprezentowany na rysunku 4b. Dla uszeregowania tego zdyskontowane wydatki wykonawcy związane z wykonywaniem zadań wynoszą:

$$\sum_{i=1}^{N_A} (CFA_i \cdot e^{-\alpha \cdot ST_i}) = -12 \cdot e^0 - 9 \cdot e^0 - 10 \cdot e^{-0.05} - 10 \cdot e^{0.03} - 16 \cdot e^{-0.03} - 12 \cdot e^{0.08} + \\ - 9 \cdot e^{-0.04} - 6 \cdot e^{-0.09} = -80.95.$$

Zdyskontowane wydatki wykonawcy są wyższe niż dla harmonogramu z rysunku 4a, w związku z wcześniejszym rozpoczynaniem części zadań w wyniku przesunięć

lewostronnych etapów projektu. Przesunięcia te zwiększają zdyskontowane płatności klienta za zrealizowane etapy przedsięwzięcia, które wynoszą:

$$\sum_{m=1}^{N_M} (CFM_m \cdot e^{-\alpha \cdot MT_m}) = 50 \cdot e^{-0.03} + 50 \cdot e^{-0.07} + 100 \cdot e^{-0.11} = 184.72.$$

Łącznie funkcja celu dla harmonogramu z rysunku 4b jest wyższa niż dla uszeregowania z rysunku 3a:

$$F = \sum_{i=1}^{N_A} (CFA_i \cdot e^{-\alpha \cdot ST_i}) + \sum_{m=1}^{N_M} (CFM_m \cdot e^{-\alpha \cdot MT_m}) = -80.95 + 184.72 = 103.77.$$

Obliczenia pokazują, że dla przykładowego projektu harmonogramowanie zgodnie z zasadą *ALAP* przy uwzględnieniu umownych terminów realizacji etapów nie jest korzystne. Harmonogram z rysunku 4a ($F = 102.56$) ma niższą o 1.19 wartość funkcji celu F niż harmonogram z rysunku 4b ($F = 103.77$). Zastosowanie procedury z optymalizacją terminów realizacji etapów projektu zwiększa NPV projektu, dzięki uzyskaniu większych zdyskontowanych płatności klienta za wcześniej ukończone etapy projektu, mimo większych zdyskontowanych wydatków wykonawcy na wcześniejsze realizowanie zadań.

5. Algorytm symulowanego wyżarzania

Problem harmonogramowania projektu z ograniczoną dostępnością zasobów jest zagadnieniem NP-trudnym (Błażewicz [3]). W związku z tym uzasadnione jest przy dużej liczbie zadań generowanie rozwiązania za pomocą algorytmów przybliżonych. Analizę efektywności stosowanych algorytmów dla RCPSPP można znaleźć w pracach przeglądowych (Hartmann i Kolisch [9], Kolisch i Padman [21]). W tej pracy stosowana jest jedna z metaheurystyk: algorytm symulowanego wyżarzania SA (ang. *Simulated Annealing*), który jest efektywny i często stosowany dla RCPSPP (Hartmann i Kolisch [9], Bouleimen i Lecocq [4]). Rozwiązania reprezentowane są w postaci listy czynności i dekodowane w wykonywalny harmonogram przy użyciu procedury SGS (szeregowej lub równoległej) generującej harmonogram wsteczny przy optymalizacji terminów ukończenia umownych etapów projektu, opisaną w rozdziale 4.

Opracowany przez autorów algorytm SA (patrz: rysunek 4) jest zbliżony do podstawowej wersji tej metaheurystyki (Kirkpatrick i in. [14]).

```

Inicjalizacja;
Strojenie;
S' := proceduraSGS(x);
T := T0;
REPEAT
BEGIN
    Wybierz sąsiednie y dla x;
    S := proceduraSGS(y);
    IF (P(-F(S), -F(S'), T) > rand) THEN
        BEGIN
            x := y;

```

```

S' := S;
IF (F(S*) < F(S)) THEN
    S* := S;
END;
Zaktualizuj temperaturę bieżącą T zgodnie z przyjętym schematem chłodzenia;

END
UNTIL (T < Tk)
Zwróć harmonogram S*

```

gdzie: T – temperatura bieżąca, T_0 – temperatura początkowa (maksymalna), T_k – temperatura końcowa (minimalna), $rand$ – liczba losowa z przedziału $(0, 1)$, x – rozwiązanie bieżące w postaci listy czynności, y – rozwiązanie sąsiednie dla x , $proceduraSGS(x)$ – procedura generowania harmonogramu wstecznego przy uwzględnieniu etapów projektu dla rozwiązania (listy czynności) x , $F(S)$ – wartość funkcji celu F dla harmonogramu S , $P(-F(S), -F(S'), T)$ – funkcja akceptacji dla rozwiązania y i wygenerowanego dla y harmonogramu S' , S – aktualnie analizowany harmonogram, S' – aktualnie analizowany harmonogram z przesuniętym w czasie rozpoczęciem zadań, S^* – najlepszy bieżący harmonogram.

Rys. 4. Algorytm symulowanego wyżarzania

Na początku inicjalizowane i dostrajane są parametry algorytmu SA m.in.: temperatura początkowa T_0 , końcowa T_k , rozwiązanie bieżące x itp. Dla przyjętego początkowego rozwiązania x generowany jest harmonogram S' przy użyciu procedury generowania harmonogramu wstecz przy uwzględnieniu etapów projektu oraz ustawiana jest temperatura bieżąca, która równa jest temperaturze początkowej. Następnie realizowane są kolejne iteracje algorytmu powtarzane do momentu spełnienia warunku zatrzymania: osiągnięcia temperatury bieżącej równej temperaturze końcowej T_k . W każdej iteracji SA na początku wykonywany jest ruch generujący nowe rozwiązanie y z otoczenia bieżącego rozwiązania x – jako operatory ruchu stosowane są: zamień, zamień sąsiednie, wstaw. Dla rozwiązania y uruchamiana jest wsteczna procedura dekodująca uwzględniająca etapy projektu, która tworzy harmonogram S . Jeśli spełnione jest kryterium akceptacji dla funkcji $P(-F(S), -F(S'), T) = \exp((F(S) - F(S'))/T)$, rozwiązanie y jest zapisywane jako rozwiązanie bieżące x . Jeśli harmonogram S wygenerowany dla y , jest najlepszym dotychczasowym rozwiązaniem, jest zapamiętywany jako S^* . Na koniec iteracji aktualizowana (zmniejszana) jest temperatura bieżąca T – stosowane są następujące schematy chłodzenia: liniowy, logarytmiczny lub geometryczny. W danej temperaturze T analizowane jest jedno rozwiązanie wygenerowane procedurą wsteczną przy uwzględnieniu etapów projektu.

6. Eksperymenty obliczeniowe

Eksperymenty przeprowadzono na komputerze z Intel Core I7, 3.0 GHz, z RAM 8 GB przy użyciu aplikacji napisanej w języku C# w środowisku Visual Studio.NET. Do obliczeń wykorzystywano 1440 instancji testowych z biblioteki PSPLIB (Kolisch i Sprecher [20]) ze zbiorów J30 (projekty złożone z 30 zadań), J60 (projekty złożone z 60 zadań) i J90 (projekty złożone z 90 zadań). Dla każdego projektu z PSPLIB autorzy wprowadzają trzyetapowe rozliczanie prac (Klimek, Łebkowski [15]). Dla projektów ze zbioru J30 zbiory zadań realizowanych w poszczególnych etapach są następujące: $MA_1 = \{1, 2, \dots, 10\}$, $MA_2 = \{11, 12, \dots, 20\}$, $MA_3 = \{21, 22, \dots, 30\}$.

Dla projektów złożonych z 60 zadań: $MA_1 = \{1, 2, \dots, 20\}$, $MA_2 = \{21, 22, \dots, 40\}$, $MA_3 = \{41, 42, \dots, 60\}$. Dla projektów 90-zadaniowych zbiory te zawierają czynności: $MA_1 = \{1, 2, \dots, 30\}$, $MA_2 = \{31, 32, \dots, 60\}$, $MA_3 = \{61, 62, \dots, 90\}$.

Umowne terminy realizacji etapów projektu określone są na podstawie terminu realizacji projektu *duedate* z instancji testowej PSPLIB:

$$MD_m = \left\lfloor \frac{1.4 \cdot \text{duedate}}{N_M} \cdot m \right\rfloor, \quad m \in \langle 1, N_M \rangle. \quad (6)$$

W rozliczeniach finansowych stosowane są identyczne parametry dla każdej instancji testowej: $MP_1 = 40$, $MP_2 = 40$, $MP_3 = 80$, $MC_1 = 1$, $MC_2 = 1$, $MC_3 = 2$. Koszty związane z wykonywaniem zadań CFA_i są wyliczane jako proporcjonalne do łącznego zapotrzebowania na zasoby i czasu realizacji danej czynności, a ich suma dla wszystkich zadań wynosi 100:

$$CFA_i = \frac{d_i \cdot \sum_{k=1}^K d_{ik}}{\sum_{j=1}^{N_A} (d_j \cdot \sum_{k=1}^K r_{jk})} \cdot 100, \quad \text{dla } i = 1 \dots N_A. \quad (6)$$

Stopa dyskontowa stosowana w eksperymentach jest równa $\alpha = 0.01$.

Do przeszukiwania przestrzeni potencjalnych rozwiązań przy wykorzystaniu proponowanych procedur stosowany jest algorytm symulowanego wyżarzania, dla którego parametry są dobrane w drodze eksperymentalnej. Analizowano różne schematy chłodzenia: logarytmiczny, geometryczny lub liniowy, różne ruchy: zamień, zamień sąsiednie lub wstaw oraz różne temperatury początkowe. W tej pracy rozwiązania generowane są przy użyciu najefektywniejszej konfiguracji *SA* dla analizowanego zagadnienia: geometrycznym schemacie chłodzenia, ruchu zamień i temperaturze początkowej 0.1 (Klimek, Łebkowski, [15]).

Wyniki eksperymentów obliczeniowych przedstawione są w tabeli 1 dla projektów 30-zadaniowych, w tabeli 2 dla projektów 60-zadaniowych oraz w tabeli 3 dla projektów 90-zadaniowych.

Celem eksperymentów obliczeniowych jest sprawdzenie efektywności procedury wstecznej generowania harmonogramu przy optymalizacji terminów realizacji umownych etapów projektu (oznaczonej w tabelach 1, 2, 3 jako Alg2 przy stosowanej szeregowej procedurze SGS i jako Alg4 przy równoległej procedurze SGS). Dla porównania uzyskanych rezultatów przedstawione są również wyniki dla najskuteczniejszego z analizowanych przez autorów (Klimek, Łebkowski [15,18]) algorytmów przesunięć prawostronnych zadań dla harmonogramu wygenerowanego w przód przy przyjętej alokacji zasobów (oznaczonego w tabelach 1, 2, 3 jako Alg1). Dodatkowo w eksperymentach analizowane są procedury generowania rozwiązań Alg3 (z szeregowym SGS) i Alg5 (z równoległym SGS), w których rozwiązania generowane są wstecz przy przyjęciu (bez optymalizacji) terminów realizacji etapów projektu równych umownym terminom.

Ze względu na stochastyczny charakter algorytmu *SA* dla każdej instancji testowej i każdej procedury (Alg1-5) wykonywane są trzy przebiegi algorytmu, co

daje łącznie 1440 uruchomień SA dla każdej z procedur (Alg1-5) dla każdego z zestawów J30, J60 i J90.

Tabela 1
Wyniki eksperymentów obliczeniowych dla zadań testowych ze zbioru J30

Algorytm	t	F_{1000}	F_{5000}	F_M	F_A	$\#F_{best}$	$\%F_{best}$
Alg1	1.19	19.12	19.45	97.61	-78.16	97	-7.5%
Alg2	1.76	20.52	20.76	97.16	-76.40	1065	-1.3%
Alg3	0.06	15.97	16.23	86.50	-70.27	7	-22.8%
Alg4	8.55	14.33	14.41	88.18	-73.77	111	-31.5%
Alg5	0.26	9.29	9.29	78.90	-69.61	0	-55.8%

gdzie: t – średni czas trwania algorytmu w sekundach, F_{1000} – średnia wartość składowej funkcji celu wyznaczona po 1000 przebiegów algorytmu SA , F_{5000} – średnia wartość składowej funkcji celu wyznaczona po 5000 przebiegów algorytmu SA , F_M – średnia wartość składowej funkcji celu wyznaczonej na podstawie zdyskontowanych płatności klienta CFM_m , F_A – średnia wartość składowej funkcji celu wyznaczonej na podstawie zdyskontowanych wydatków wykonawcy CFA_i , $\#F_{best}$ – liczba rozwiązań identycznych z najlepszym znalezionym przez wszystkie analizowane algorytmy (spośród 1440 uruchomień algorytmu – dla 480 instancji testowych i trzykrotnego uruchomienia algorytmu dla każdej instancji), $\%F_{best}$ – średnie odchylenie od najlepszych rozwiązań znalezionych przez wszystkie analizowane algorytmy.

Tabela 2
Wyniki eksperymentów obliczeniowych dla zadań testowych ze zbioru J60

Algorytm	t	F_{1000}	F_{5000}	F_M	F_A	$\#F_{best}$	$\%F_{best}$
Alg1	7.23	8.76	9.21	80.70	-71.49	9	-31.1%
Alg2	6.25	12.26	12.91	80.92	-68.01	698	-3.5%
Alg3	0.16	7.08	7.72	68.45	-60.73	7	-42.3%
Alg4	60.96	5.90	6.27	70.85	-64.58	39	-53.1%
Alg5	0.36	-0.05	0.14	59.86	-59.72	0	-99.0%

oznaczenia jak w tabeli 1

Tabela 3
Wyniki eksperymentów obliczeniowych dla zadań testowych ze zbioru J90

Algorytm	t	F_{1000}	F_{5000}	F_M	F_A	$\#F_{best}$	$\%F_{best}$
Alg1	31.10	2.23	2.64	71.03	-68.39	2	-70.7%
Alg2	12.37	7.69	8.50	71.17	-62.67	567	-5.8%
Alg3	0.27	2.40	3.14	58.07	-54.93	7	-65.2%
Alg4	166.05	2.29	2.67	61.98	-59.30	21	-70.4%
Alg5	0.56	-3.26	-3.09	50.88	-53.96	0	-134.2%

oznaczenia jak w tabeli 1

Średnia wartość funkcji celu F najlepszych znalezionych rozwiązań przez wszystkie zastosowane algorytmy dla zestawów J30, J60 i J90 wynosi odpowiednio 21.03, 13.38 i 9.02. Zastosowanie procedur przesunięć terminów wykonania etapów projektu dla każdej instancji testowej zwiększa NPV projektu.

Rozwiązania o najwyższej wartości F (dla każdego z zestawów J30, J60 i J90) generowane są przy użyciu szeregowej procedury SGS działającej wstecz z optymalizacją terminów wykonania umownych etapów projektu (Alg2).

Rozwiązania o najniższej wartości F (dla każdego z zestawów J30, J60 i J90) generowane są przy użyciu równoległej procedury SGS działającej wstecz (Alg4)

i Alg5). Zastosowanie procedury optymalizacji terminów wykonania umownych etapów projektu (Alg4) poprawia jakość uzyskanych rozwiązań, ale są one gorsze niż dla rozwiązań wygenerowanych przy wykorzystaniu szeregowej procedury SGS.

Stosowanie strategii tworzenia harmonogramów w przód przy przyjętej alokacji zasobów (Alg1) pozwala na osiągnięcie zbliżonych wartości składowej funkcji celu F_M wyznaczonej na podstawie zdyskontowanych płatności klienta CFM_m jak dla procedury Alg2, ale większych zdyskontowanych wydatków (składowa F_A) w związku z wcześniejszym rozpoczynaniem części zadań, których przy przyjętej alokacji nie udało się przesunąć w prawo na wykresie Gantt'a (Klimek i Łebkowski [15,18]).

Najniższe zdyskontowane wydatki (składowa F_A) są wyznaczone przy użyciu harmonogramowania wstecznego przy przyjęciu terminów realizacji etapów projektu równych umownym terminom (Alg3 i Alg5). Jednak dla rozwiązań wygenerowanych za pomocą Alg3 i Alg5 wartość zdyskontowanych płatności klienta (składowa F_M) jest niższa niż dla uszeregowanych znalezionych przy wykorzystaniu proponowanej procedury Alg2 lub Alg4.

Analizowana w pracy procedura optymalizacji terminów realizacji etapów projektu (przesuwania ich w prawo) jest czasochłonna, ale poprawia uzyskane rozwiązania. Zwiększanie liczby analizowanych rozwiązań przez SA poprawia jakość harmonogramów – uszeregowania ustalone po 1000 przebiegów SA są gorsze niż harmonogramy znalezione po 5000 przebiegów SA.

7. Podsumowanie

W artykule przeanalizowano problem maksymalizacji zdyskontowanych przepływów pieniężnych dla projektu rozliczanego etapowo z perspektywy jego wykonawcy. Przedstawiono procedurę generowania harmonogramu wstecz przy optymalizacji terminów wykonania umownych etapów przedsięwzięcia.

Wyniki eksperymentów obliczeniowych, w których stosowano metaheurystykę symulowanego wyżarzania wykorzystującą różne algorytmy generowania rozwiązań dla analizowanego zagadnienia, potwierdzają efektywność rozważanej w artykule procedury.

Podjęte zagadnienie jest aktualne i może być przydatne w praktycznych zastosowaniach.

LITERATURA

1. Bahrami F., Moslehi G.: Study of payment scheduling problem to achieve client–contractor agreement. *International Journal of Advanced Manufacturing Technology*, 64(1), 2013, p. 497-511.
2. Baroum S., Patterson J.: The development of cash flow weight procedures for maximizing the net present value of a project. *Journal of Operations Management*, 14(3), 1996, p. 209-27.

3. Błażewicz J., Lenstra J., Kan A. R.: Scheduling subject to resource constraints - classification and complexity. *Discrete Applied Mathematics*, 5, 1983, p. 11-24.
4. Bouleimen K., Lecocq H.: A new efficient simulated annealing algorithm for the resource constrained project scheduling problem and its multiple version. *European Journal of Operational Research*, 149, 2003, p. 268-281
5. Dayanand N., Padman R.: Project contracts and payment schedules: The client's problem. *Management Science*, 47, 2001, p. 1654-1667.
6. Dayanand N., Padman R.: On modelling payments in projects. *Journal of the Operational Research Society*, 48, 1997, p. 906-918.
7. Fink A., Homberger J.: An ant-based coordination mechanism for resource-constrained project scheduling with multiple agents and cash flow objectives, *Flexible Services and Manufacturing Journal*, 25, 2013, p. 94-121.
8. Hartmann S., Briskorn D.: A Survey of Variants and Extensions of the Resource-Constrained Project Scheduling Problem, *European Journal of Operational Research*, Vol. 207(1), 2012. p. 1-14.
9. Hartmann S., Kolisch R., Experimental evaluation of state-of-the-art heuristics for the resource-constrained project scheduling problem. *European Journal of Operational Research*, 127, 2000, p. 394-407.
10. He Z. W., Xu Y.: Multi-mode project payment scheduling problems with bonus-penalty structure. *European Journal of Operational Research*, 189, 2008, p. 1191-1207.
11. He Z., Wang N., Jia T., Xu Y.: Simulated annealing and tabu search for multimode project payment scheduling. *European Journal of Operational Research* 198 (3), 2009, p. 688-696.
12. Herroelen W., Reyck B. D., Demeulemeester E.: Project network models with discounted cash flows: A guided tour through recent developments, *European Journal of Operational Research*, 100, 1997, p. 97-121.
13. Khoshjahan Y., Najafi A., Nadjafi B.: Resource constrained project scheduling problem with discounted earliness-tardiness penalties: Mathematical modeling and solving procedure, *Computers & Industrial Engineering* 66, 2013, p. 293-300.
14. Kirkpatrick S., Gelatt C.D., Vecchi M.P., Optimization by simulated annealing, *Science*, 220, 1983, p. 671-680.
15. Klimek M. Łebkowski P.: Harmonogramowanie projektu rozliczanego etapowo, *Monografia Wydawnictw AGH*, Kraków, 2015.
16. Klimek M., Łebkowski P.: Robustness of schedules for project scheduling problem with cash flow optimisation. *Bulletin of the Polish Academy of Sciences Technical Sciences*, Vol. 61, No. 4, 2013, p. 1005-1015.
17. Klimek M., Łebkowski P.: A Two-Phase Algorithm for Resource Constrained Project Scheduling Problem with Discounted Cash Flows. *Decision Making in Manufacturing and Services*, Vol. 7, No. 1-2, 2013, p. 49-66.

18. Klimek M., Łebkowski P.: Heuristics for project scheduling with discounted cash flows optimisation. *Bulletin of the Polish Academy of Sciences Technical Sciences*, Vol. 63, No. 3, 2015, p. 613-622.
19. Klimek M., Łebkowski P.: Algorytm dla problemu maksymalizacji zdyskontowanych przepływów pieniężnych projektu rozliczanego etapowo. *Innowacje w zarządzaniu i inżynierii produkcji*, T. 1 / pod red. Ryszarda Knosali. Opole : Oficyna Wydawnicza PTZP, 2014, s. 572-582.
20. Kolisch R., Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation. *European Journal of Operational Research*, 90, 1996, p. 320-333.
21. Kolisch R., Padman R.: An integrated survey of deterministic project scheduling. *OMEGA The International Journal of Management Science*, 29, 2001, p. 249-272.
22. Kolisch R., Sprecher A.: PSPLIB – a project scheduling library. *European Journal of Operational Research*, 96, 1997, p. 205-216.
23. Leyman P., Vanhoucke M.: Payment models and net present value optimization for resource-constrained project scheduling, *Computers & Industrial Engineering* 91, 2016, p. 139-153.
24. Mika M., Waligóra G., Węglarz J.: Simulated annealing and tabu search for multi-mode resource-constrained project scheduling with positive discounted cash flows and different payment models, *European Journal of Operational Research*, 2005, vol. 164, no. 3 p. 639-668.
25. Pinder J.P., Maruchek A.S.: Using discounted cash flow heuristics to improve project net present value. *Journal of Operations Management* 14, 1996, p. 229-240.
26. Russell A.H.: Cash flows in networks, *Management Science*, 16, 1970, p. 357-373.
27. Smith-Daniels D. E., Padman R., Smith-Daniels V. L.: Heuristic scheduling of capital constrained projects. *Journal of Operations Management*, 14(3), 1996, p. 241-254.
28. Stabryła A.: Zarządzanie projektami ekonomicznymi i organizacyjnymi. Wydawnictwo Naukowe PWN, Warszawa, 2006.
29. Thiruvady D., Wallace M., Gu H., Schutt A.: A lagrangian relaxation and ACO hybrid for resource constrained project scheduling with discounted cash flows, *Journal of Heuristics*, 20, 2014, p. 643-676.
30. Trocki M.: Nowoczesne zarządzanie projektami, Polskie Wydawnictwo Ekonomiczne, Warszawa, 2012.
31. Ulusoy G., Cebelli S.: An equitable approach to the payment scheduling problem in project management. *European Journal of Operational Research* 127(2), 2000, p. 262-278.

32. Ulusoy G., Özdamar L.: A heuristic scheduling algorithm for improving the duration and net present value of a project. *International Journal of Operations and Production Management*, 15 , 1995, p.89-98.
33. Ulusoy G., Sivrikaya-Serifoglu F., Sahin S.: Four Payment Models for the Multi-Mode Resource Constrained Project Scheduling Problem with Discounted Cash Flows. *Annals of Operations Research*, 102, 2001, p. 237-261.
34. Vanhoucke M., Demeulemeester E., Herroelen W.: Maximizing the net present value of a project with linear time-dependent cash flows. *International Journal of Production Research*, 39(14), 2001, p. 3159-3181.
35. Vanhoucke M.: A scatter search procedure for maximizing the net present value of a resource-constrained project with fixed activity cash flows, Working Paper 2006/417, Gent, p. 1-23.