

Czesław SMUTNICKI¹
Jarosław RUDY²
Radosław IDZIKOWSKI²
Wojciech BOŻEJKO²

¹Katedra Informatyki Technicznej, Politechnika Wrocławska

²Katedra Automatyki, Mechatroniki i Systemów Sterowania, Politechnika Wrocławska

PODEJŚCIE DUALNE DLA PROBLEMU MINIMALIZACJI WAŻONEJ SUMY SPÓŹNIEŃ Z WYKORZYSTANIEM METAHEURYSTYK

Streszczenie. W pracy przedstawiono podejście dualne dla problemu minimalizacji ważonej sumy spóźnień. Podejście to można zastosować zarówno do rozwiązania problemu wyjściowego, jak i do uzyskania dobrej jakości dolnego ograniczenia dla metody podziału i ograniczeń. Opisane podejście zostało zaimplementowane z wykorzystaniem metody subgradientu jak i metaheurystyki poszukiwania lokalnego. Przedstawiony przykłady ilustrujące cechy zaimplementowanych algorytmów oraz jakość uzyskiwanych rozwiązań.

METAHEURISTIC-BASED LAGRANGIAN RELAXATION APPROACH TO MINIMIZE THE TOTAL WEIGHTED TARDINESS

Summary. In this paper a Lagrangian relaxation approach to the problem of minimization of the total weighted tardiness is presented. The approach can be used both to solve the original problem as well as to obtain high-quality lower bound for the branch-and-bound method. The approach was then implemented using several variants of the subgradient method as well as a local-search metaheuristic. Examples are shown to illustrate the characteristics of the proposed algorithm as well as quality of obtained solutions.

1. Wprowadzenie

Zagadnienie minimalizacji ważonej sumy spóźnień zadań, oznaczane jako $1||\sum w_i T_i$ w notacji Grahama, jest jednym z najczęściej rozważanych jednomaszynowych problemów szeregowania zadań, zarówno w kontekście teoretycznym jak i praktycznym [2, 19]. Ze względu na NP-trudność problemu, jedną z najbardziej popularnych dokładnych metod jego rozwiązania pozostaje metoda podziału i ograniczeń [16]. Metoda ta wymaga do działania obliczania wartości tzw. górnych (UB) i dolnych (LB) ograniczeń problemu. Ponieważ dowolne rozwiązanie dopuszczalne problemu $1||\sum w_i T_i$ stanowi jego górne ograniczenie, to UB może zostać łatwo wyznaczone przy użyciu dowolnej metaheurystyki dla danego problemu. Obliczenie wartości LB jest w praktyce trudniejsze, szczególnie w przypadku problemów takich jak $1||\sum w_i T_i$. W poniższej pra-

cy proponujemy podejście dualne, pozwalające na obliczenie wartości LB przy użyciu pewnego algorytmu hybrydowego z wykorzystaniem metaheurystyki.

2. Problem podstawowy

Zacznijmy od sformalizowania podstawowego problemu $1||\sum w_i T_i$. Dany jest zbiór n zadań $\mathcal{N} = \{1, 2, \dots, n\}$. Dla każdego zadania dany jest czas wykonywania $p_i > 0$, żądany termin zakończenia $d_i > 0$ oraz waga $w_i > 0$, $i \in \mathcal{N}$. Należy określić harmonogram tj. wektor terminów rozpoczęcia zadań $S = (S_1, S_2, \dots, S_n)$, tak by zminimalizować sumę ważonych spóźnień poszczególnych zadań, mianowicie:

$$\min_S \sum_{i=1}^n w_i [S_i + p_i - d_i]^+ = \min_S \sum_{i=1}^n w_i T_i, \quad (1)$$

gdzie $T_i \stackrel{\text{def}}{=} [S_i + p_i - d_i]^+$, $[x]^+ \stackrel{\text{def}}{=} \max\{x, 0\}$, jest spóźnieniem zadania i . Mimo iż T_i jest funkcją zależną od S_i , dalej dla zwięzłości zapisu, będziemy używać T_i zamiast $[S_i + p_i - d_i]^+$. Maszyna może wykonywać co najwyżej jedno zadanie jednocześnie, więc rozwiązanie S musi spełniać ograniczenie:

$$\forall_{i,j \neq i} : (S_i + p_i \leq S_j) \vee (S_j + p_j \leq S_i). \quad (2)$$

Dla sformułowanego problemu, z uwagi na regularność funkcji kryterialnej, harmonogram S jest dosunięty w lewo na osi czasu zatem może być jednoznacznie reprezentowany przez kolejność wykonywania (permutację) zadań $\pi = (\pi(1), \pi(2), \dots, \pi(n))$, gdzie $\pi(i)$ określa które zadanie jest wykonywane jako i -te w kolejności. Wtedy dla każdego π terminy S wyznacza się ze wzoru rekurencyjnego:

$$S_{\pi(1)} = 0, \quad (3)$$

$$S_{\pi(i)} = S_{\pi(i-1)} + p_{\pi(i-1)}, \quad i = 2, 3, \dots, n \quad (4)$$

w czasie liniowym $O(n)$.

3. Relaksacja i dolne ograniczenie

Tak zwane podejście dualne opiera się na relaksacji części ograniczeń oryginalnego problemu oraz uwzględnieniu ich poprzez funkcję kary [7]. Dzięki ogólnej naturze podejście to zostało zastosowane dla szerokiej klasy problemów optymalizacji. Przykłady z dziedziny szeregowania zadań obejmują problem przepływowy [8, 14] i gniazdowy [5], szeregowanie stochastyczne [15], harmonogramowanie tras pociągów [20] i pracy dźwigów [9] oraz procesy przemysłowe np. odlewanie [6]. Poza szeregowaniem, podejście dualne znalazło zastosowanie w zagadnieniach marszrutyzacji [13], alokacji zasobów [11], optymalizacji portfolio [18], problemach minimalizacji kosztu w sieciach przepływowych [4] a nawet w klasyfikacji [10].

W naszym przypadku relaksujemy ograniczenie (2), pozwalając by maszyna wykonywała dowolną liczbę zadań równocześnie. Założeniem istotnym dla dalszych rozważań jest całkowitoliczbowość czasów wykonywania p_i , pozwalające na wprowadzenie dyskretyzacji upływu czasu w harmonogramowaniu. W konsekwencji, terminy rozpoczęcia i zakończenia zadań przyjmują wyłącznie wartości całkowite ze zbioru

$\{0, 1, \dots, H\}$, gdzie H jest górnym ograniczeniem horyzontu czasowego dla harmonogramu. W praktyce $H = \sum_{i=1}^n p_i$. To umożliwia nam zdefiniowanie zbioru zadań wykonywanych w przedziale czasu $[t - 1, t]$:

$$\mathcal{I}_t(S) \stackrel{\text{def}}{=} \{i \in \mathcal{N} : t > S_i \geq t - p_i\}, \quad (5)$$

oraz liczbę zadań wykonywanych w tym przedziale:

$$g_t(S) \stackrel{\text{def}}{=} |\mathcal{I}_t(S)|. \quad (6)$$

Problem z rozdziału 2 można teraz zapisać jako zadanie optymalizacji nieliniowej:

$$\min_S \sum_{i=1}^n w_i T_i \quad (7)$$

przy ograniczeniach:

$$g_t(S) = 1, \quad t = 1, 2, \dots, H, \quad (8)$$

$$0 \leq S_i \leq H - p_i, \quad i = 1, 2, \dots, n. \quad (9)$$

Ograniczenie (8) wymusza by w każdym przedziale czasowym $[t - 1, t]$ liczba wykonywanych zadań była równa jeden ze względu na jednostkową przepustowość stanowiska. Ograniczenie (9) uwzględniamy bezpośrednio, wprowadzając zbiór:

$$\mathcal{S} \stackrel{\text{def}}{=} \{S = (S_1, S_2, \dots, S_n) : 0 \leq S_i \leq H - p_i, \quad i = 1, 2, \dots, n\}, \quad (10)$$

zaś dla ograniczenia (8) uwzględniamy poprzez zmienne dualne $u \stackrel{\text{def}}{=} (u_1, u_2, \dots, u_H)$, gdzie u_t jest dla ustalonego t . Wartość u_t może być interpretowane jako koszt użycia maszyny w przedziale $[t - 1, t]$. W efekcie otrzymujemy następującą funkcję Lagrange'a:

$$L(S, u) \stackrel{\text{def}}{=} \sum_{i=1}^n w_i T_i + \sum_{t=1}^H u_t (g_t(S) - 1). \quad (11)$$

Zauważmy, że w oryginalnym problemie $1 \parallel \sum w_i T_i$ dla dowolnego dosuniętego w lewo dopuszczalnego harmonogramu S i każdego $t = 1, 2, \dots, H$ musi zachodzić $g_t(S) = 1$. Wtedy wzór (11) dostarcza wartość funkcji celu $\sum_{i=1}^n w_i T_i$ tego harmonogramu.

Funkcję (11) przekształcimy dalej do wygodniejszej postaci.

$$L(S, u) = \sum_{i=1}^n (w_i T_i + \sum_{t=S_i+1}^{S_i+p_i} u_t) - \sum_{t=1}^H u_t = \sum_{i=1}^n L_i(S_i, u) - U_H, \quad (12)$$

gdzie

$$L_i(S_i, u) \stackrel{\text{def}}{=} w_i T_i + \sum_{t=S_i+1}^{S_i+p_i} u_t = w_i T_i + U_{S_i+p_i} - U_{S_i} \quad (13)$$

oraz

$$U_t = \sum_{s=1}^t u_s, \quad t = 1, 2, \dots, H. \quad (14)$$

Niech S^* będzie optymalnym rozwiązaniem problemu oryginalnego (7)–(9), zaś $W(u)$ niech będzie optymalnym rozwiązaniem problemu $\min_S L(S, u)$ dla pewnego ustalonego u . Zauważmy, że wtedy dla każdego u zachodzi:

$$\sum_{i=1}^n w_i [S_i^* + p_i - d_i]^+ = \sum_{i=1}^n w_i [S_i^* + p_i - d_i]^+ + \sum_{t=1}^H u_t (g_t(S^*) - 1) \geq$$

$$\geq \min_{S \in \mathcal{S}} L(S, u) \stackrel{\text{def}}{=} W(u), \quad (15)$$

czyli $W(u)$ dostarcza wartości LB dla funkcji celu (7) na mocy pojęcia relaksacji. Ponieważ interesuje nas jak największa wartość LB, to otrzymamy ją z maksymalizacji:

$$\max_u W(u) = \max_u \min_S L(S, u). \quad (16)$$

W ogólności opisany powyżej model może posłużyć także do rozwiązania problemu podstawowego (7)–(9). Istotnie, na mocy definicji relaksacji, jeśli dla pewnego u znajdziemy S^* z minimalizacji $\min_S L(S, u) = L(S^*, u)$ dla, którego $g_t(S^*) = 1$, $t = 1, 2, \dots, H$, to S^* jest rozwiązaniem optymalnym problemu (7)–(9). Problem (16) można zdekomponować na dwa podproblemy: $\max_u W(u)$ nazywany dalej problemem górnego poziomu, oraz $\min_S L(S, u)$ nazywany dalej problemem dolnego poziomu. Problem dolnego poziomu zwykle jest rozwiązywany wielokrotnie dla różnych wartości u . Dla rozwiązania problemu górnego poziomu możemy użyć metody przybliżonej (heurystyki, metaheurystyki), jednakże dolny poziom musimy zawsze rozwiązać optymalnie, by otrzymana wartość była dolnym ograniczeniem, zgodnie z zależnością (15). Dalej omówimy podejścia do rozwiązania obu pod-problemów.

4. Problem dolnego poziomu

Uwzględniając (10)–(13), dla ustalonego u otrzymujemy:

$$W(u) = \min_{S \in \mathcal{S}} L(S, u) = \sum_{i=1}^n \min_{0 \leq S_i \leq H-p_i} L_i(S_i, u) - U_H = \sum_{i=1}^n V_i(u) - U_H \quad (17)$$

gdzie

$$V_i(u) \stackrel{\text{def}}{=} \min_{0 \leq S_i \leq H-p_i} L_i(S_i, u) = \min_{S_i=0,1,\dots,H-p_i} L_i(S_i, u). \quad (18)$$

Dla ustalonego u każdy problem (18) można rozwiązać poprzez bezpośredni przegląd możliwych terminów rozpoczęcia S_i . Ponieważ wzór (14) może być zapisany w formie rekurencyjnej $U_t = U_{t-1} + u_t$, $t = 1, 2, \dots, H$, $U_0 = 0$, to obliczenie wartości U_t można zrealizować w czasie $O(H)$. Zatem wyliczenie (13) poprzez (18) wymaga czasu $O(H)$, zaś wyznaczenie $W(u)$ ze wzoru (17) dla ustalonego u wymaga czasu $O(nH)$.

5. Problem górnego poziomu

Problem (16) jest zadaniem optymalizacji ciągłej, nieliniowej (dokładniej odcinkami liniowej), słabo wypukłej, nieróżniczkowalnej w punktach "sklejanych", o dużej liczbie składowych wektora u . Formalnie można go sprowadzić do zadania programowania liniowego (PL) jednakże o znacznym rozmiarze. Przykładowo dla $H = 1000$, $n = 100$ oczekiwany rozmiar PL to 10^3 zmiennych i 10^{300} warunków ograniczających. Ze względu na brak różniczkowalności, proponowane dalej jest kilka alternatywnych podejść: (1) metoda subgradientowa, (2) metaheurystyka poszukiwania lokalnego, (3) metoda hybrydowa. Przedstawimy je kolejno w pewnej syntetycznej formie.

5.1. Metoda subgradientu

Ta często stosowana w problemach dualnych metoda (SUB) korzysta z pojęcia subgradientu dla funkcji nieróżniczkowalnych [12]. W skrócie, generuje ona ciąg wektorów u^0, u^1, u^2, \dots według zależności

$$u_t^{k+1} = u_t^k + \alpha^k (g_t(S^k) - 1), \quad t = 1, 2, \dots, H, \quad (19)$$

gdzie α^k jest długością kroku w k -tej iteracji, zaś S^k jest optymalnym rozwiązaniem problemu dolnego poziomu w k -tej iteracji, tzn. $\min_S L(S, u^k) = L(S^k, u^k)$. Dobór ciągu α^k ma znaczący wpływ na własności metody, takie jak stabilność, zbieżność oraz szybkość zbieżności LB do UB. Teoretycznie, zbieżność zapewnia już ciąg taki, że $\lim_{k \rightarrow \infty} \alpha^k = 0$, $\sum_{k=1}^{\infty} \alpha^k = \infty$, czyli na przykład ciąg harmoniczny $\alpha^k = c/k$, dla pewnej stałej c . Otwartym zagadnieniem pozostaje dobór $c \stackrel{\text{def}}{=} \alpha^1$, bowiem zależy on silnie od rodzaju i wielkości instancji testowych. Wymaga to żmudnego „strojenia” metody. Brak wzrostu $W(u^k)$ dla $k = 1, 2, \dots$, może być powodowany zjawiskiem „zygzakowania” przy szybkim spadku α^k w kierunku zera. Podobne wnioski otrzymano także dla wolniej zbiegającego ciągu $\alpha^k = c/k^\gamma$, $\gamma \ll 1$ o podanych właściwościach. Wśród innych rekomendowanych podejść z udowodnioną zbieżnością jest także propozycja:

$$\alpha^k = \gamma^k \frac{W(u^k) - W^*}{\|g(S^k) - 1\|^2}, \quad (20)$$

gdzie W^* jest optymalną wartością funkcji celu. Mianownik jest normą naruszenia ograniczeń dla S^k , przykładowo

$$\|g(S^k) - 1\|^2 = \sum_{t=1}^H (g_t(S^k) - 1)^2. \quad (21)$$

Dla α^k określonego wzorem (20) wykazano zbieżność SUB. Wobec nieznajomości W^* praktycznie sugerowane jest wykorzystanie chwilowego górnego ograniczenia UB^k , choć takie podejście nie gwarantuje teoretycznej zbieżności. Zaleca się aby UB^k było aktualizowane w każdym kroku k przez uruchomienie pewnej *pomocniczej* metody heurystycznej. Jedną z najprostszych propozycji jest obliczenie wartości funkcji celu w oparciu o (3)–(4) dla permutacji π otrzymanej przez uporządkowanie zadań według niemalejących wartości S_i^k . Możliwe jest zastosowanie w tym miejscu innych metaheurystyk mając świadomość, że poprawi do oszacowanie UB^k lecz równocześnie zwiększy łączny czas działania algorytmu. Tematyka zbieżności metody dualnej i jej odmian pozostaje objektem badań [3].

5.2. Metaheurystyka poszukiwania lokalnego

Naszym celem jest rozwiązanie problemu $\max_u W(u)$ przy wykorzystaniu metaheurystyk oraz dokonanie oceny tego podejścia. Według naszej wiedzy jak do tej pory nie opisano takiego podejścia w literaturze. Opis zastosowania metaheurystyki podamy dla przypadku *zmodyfikowanego symulowanego wyzarzania (m-SA)*, choć możliwych do zastosowania jest tu wiele różnych innych metod przybliżonych.

Proponowana metoda m-SA generuje ciąg wektorów u^0, u^1, \dots podobnie jak typowe SA z tym, że proces losowania jest odmienny. Wektor u^k będziemy nazywać rozwiązaniem bieżącym, zaś funkcją oceny tego rozwiązania jest $W(u^k) =$

$\min_S L(S, u^k) = L(S^k, u^k)$. Lokalne sąsiedztwo $N(u^k)$ rozwiązania u^k określamy jako

$$N(u^k) = \{u = (u_1, \dots, u_H) : u_t \in [u_t^k - A, u_t^k + (n-1)A], t = 1, \dots, H\} \quad (22)$$

gdzie A jest pewną liczbą zwaną elementarnym przyrostem kary, zaś przedział wartości u_t jest ciągły. Kolejne rozwiązanie $u^{k+1} \in N(u^k)$ jest wybierane w sąsiedztwie rozwiązania bieżącego u^k i generowane w następujący sposób. Najpierw losujemy jedno rozwiązanie zaburzone \tilde{u} według zasady

$$\tilde{u}_t = u_t^k + Z_t \cdot (g_t(S^k) - 1), t = 1, 2, \dots, H, \quad (23)$$

gdzie $Z_t, t = 1, \dots, H$ są liczbami losowymi o rozkładzie równomiernym z przedziału $[0, A]$. Zauważmy, że wybór rozwiązania zaburzonego jest odmienny niż w SA bowiem rozkład losowania jest nie tylko nierównomierny w otoczeniu, ale także dynamiczny na skutek użycia wartości funkcji $g_t(S^k)$. Następnie obliczamy $\Delta = W(\tilde{u}) - W(u^k)$. Jeśli $\Delta \leq 0$ to rozwiązanie \tilde{u} akceptujemy bezwarunkowo, tzn. $u^{k+1} := \tilde{u}$. Jeśli $\Delta > 0$ to rozwiązanie \tilde{u} akceptujemy z prawdopodobieństwem $e^{(-\Delta/T)}$, gdzie T jest parametrem zwanym temperaturą. W praktyce akceptacja następuje jeśli $e^{(-\Delta/T)} < R$, gdzie R jest liczbą losową o rozkładzie równomiernym na przedziale $[0, 1]$. Jeśli rozwiązanie \tilde{u} nie zostało zaakceptowane ani przez pierwszy ani przez drugi warunek, to podstawiamy $u^{k+1} := u^k$.

Zmiana temperatury jest dokonywana okresowo lub co iterację i jest określona przez tzw. schemat studzenia. Jest znanych wiele schematów i modeli studzenia, patrz np. rozdział w książce [1]. W badaniach opisanych dalej używaliśmy schematu geometrycznego $T^{k+1} = \lambda T^k, k = 0, 1, \dots$. Proponowany algorytm ma wiele elementów strojących. Niektóre są typowe dla SA, jak na przykład u^0, T^0, λ , warunek stopu. Warto wspomnieć, że istnieje wariant SA z auto-strojeniem, który pozwala na automatyczny wybór wspomnianych wyżej parametrów, [1]. Może on być implementowany także tu.

Wielkości $W(u^k)$ wahają się losowo, zatem jako rozwiązanie metody przyjmujemy $LB = \max\{W(u^0), W(u^1), \dots\}$. Jeśli dodatkowo w każdej iteracji k zastosujemy pomocniczy algorytm przybliżony, tak jak przy metodzie subgradientu, szeregując zadania według niemalejących wartości S_i^k otrzymamy UB z ciągu górnych oszacowań $UB = \max\{UB(S^0), UB(S^1), \dots\}$.

5.3. Podejścia hybrydowe

W tym rozdziale rozważymy kilka modyfikacji algorytmów SUB i m-SA zmierzające do zmniejszenia ich złożoności obliczeniowej oraz poprawy zbieżności. Mogą być one postrzegane jako “strojenie” algorytmu, ale niektóre warianty powinny być postrzegane jako hybrydy. Rozpocznijmy od modelu (20). Istnieje co najmniej kilka modyfikacji tego schematu prowadzące do algorytmów z różnymi własnościami obserwowanymi w eksperymentach: (1) inicjowanie u^0 wartościami losowymi, (2) sterowanie celowe ciągiem γ^k , (3) sterowanie losowe ciągiem γ^k .

Propozycja (1) jest oczywista, bowiem zerowe wartości u powodują niekorzystne zachowanie się $W(u)$ w początkowej fazie algorytmu, zarówno dla SUB jak i m-SA. Istotnie dla zerowych u^0 problem $\min L(S, U)$ lokuje wszystkie zadania równocześnie w terminie zero. W konsekwencji, w serii początkowych kroków, obserwowane są liczne ujemne wartości $W(u)$ zanim nastąpi faktyczny wzrost wartości $W(u)$. Przejdźmy dalej do analizy alternatywnych podejść w zakresie sterowania ciągiem γ^k , odpowiednio do punktów (2) i (3), w kierunku przyśpieszenia szybkości zbieżności $W(u)$ do UB.

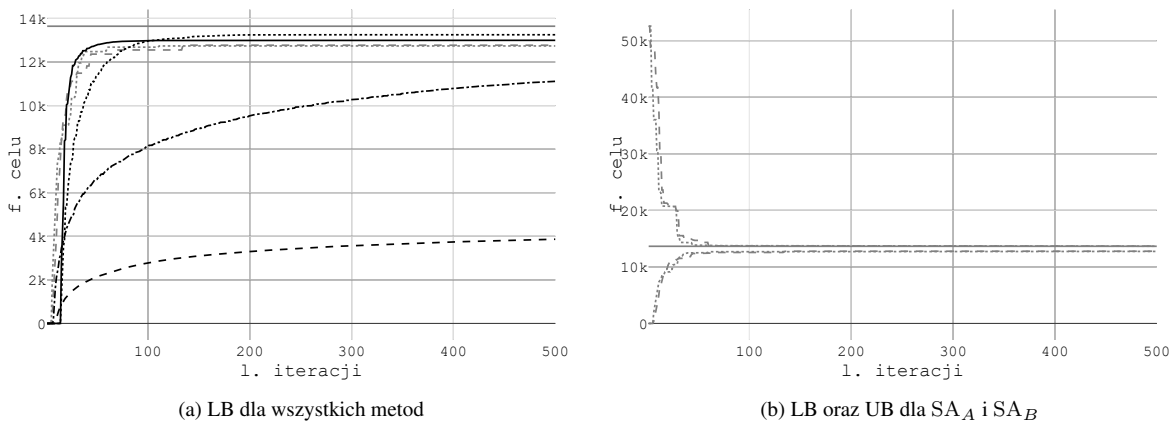
Sterowanie celowe ciągiem γ^k ma na celu automatyczne dopasowanie wielkości γ^k do instancji, tak by skrócić początkową fazę algorytmu charakteryzującą się ujemnymi wartościami $W(u)$. Sterowanie może być realizowane w różny sposób. Pierwszy, najczęściej stosowany, to: ustalić $\gamma^0 \approx 2.0$ oraz zmniejszyć γ^k o połowę, jeśli w kolejnych 5 krokach nie wystąpi wzrost wartości $W(u^k)$. Drugi, stosuje model spadek-wzrost według zasady: jeśli w jednym kroku nastąpi spadek wartości $W(u^k)$ to zmniejszyć $\gamma^{k+1} = \sigma \cdot \gamma^k$, $\sigma < 1$; jeśli nastąpił wzrost $W(u^k)$ to zwiększyć $\gamma^{k+1} = \tau \cdot \gamma^k$, $\tau > 1$. Wartości σ oraz τ powinny być bliskie jedynce. Losowy model zmian γ^k może być postrzegany jako inny wariant SA lub algorytm będący kombinacją metod VNS (Variable Neighbourhood Search) oraz RS (Random Search). W dużym uproszczeniu, polega on na wprowadzeniu do wartości γ^k we wzorze (20) losowego niewielkiego zaburzenia. Tak otrzymana zmienna losowa γ^k wykazuje zmianę wartości średniej zgodnie z (20). Eksperymenty pokazują, że losowość zapobiega wystąpieniu zjawiska zygzakowania charakterystycznego dla metody SUB.

6. Badania numeryczne

Przedstawione w rozdziale 5 metody zostały zaimplementowane w kilku wariantach, w celu ilustracji wybranych cech numerycznych oraz oceny przydatności podejścia dualnego i metod przybliżonych dla wyznaczania wartości LB i UB. Zaimplementowano następujące warianty algorytmów:

1. SUB_A – metodę subgradientu z zastosowaniem ciągu harmonicznego $\alpha^k = \frac{c}{k}$, $c = 1$; linia przerywana czarna (patrz Rys. 2a),
2. SUB_B – metodę subgradientu z zastosowaniem ciągu $\alpha^k = \frac{c}{\sqrt{k}}$, $c = 1$; linia przerywana z kropkami czarna (patrz Rys. 2a),
3. SUB_C – metodę subgradientu z zastosowaniem wzorów (20)–(21), gdzie γ początkowo ma wartość 2 i jest zmniejszane dwukrotnie jeśli w ostatnich 5 iteracjach nie znaleziono nowej najlepszej wartości $W(u)$; linia kropkowana czarna (patrz Rys. 2b),
4. SUB_D – metodę subgradientu z zastosowaniem wzorów (20)–(21), gdzie γ^k jest realizacją zmiennej losowej z rozkładu jednostajnego $[0.95; 1.05]$; linia ciągła czarna (patrz Rys. 2b),
5. SA_A – metoda m-SA przy założeniu $u^0 = \mathbf{u}$, $\lambda = 0.99$, $T^0 = 1000$, linia przerywana szara (patrz Rys. 1b),
6. SA_B – metoda m-SA przy założeniu $u^0 = \mathbf{u}$, $\lambda = 0.9$, $T^0 = 100$; linia kropkowana szara (patrz Rys. 1b).

Wielkość \mathbf{u} oznacza początkowy wektor cen dualnych o losowych wartościach $u_t \in [-1.5...3.5]$, $t = 1, \dots, H$. Algorytmy zostały zaprogramowane w języku Julia. W celu porównania różnych podejść, wszystkie wykresy dotyczą tego samego przykładu testowego. Przykład wygenerowano metodą przedstawioną na stronie OR-library [17]. Wartości p_i , w_i oraz d_i są losowymi liczbami całkowitymi z przedziałów odpowiednio $[1; 100]$, $[1; 10]$ oraz $[1 - TF - RDD/2; 1 - TF + RDD/2]$, gdzie $TF, RDD \in \{0.2, 0.4, 0.6, 0.8, 1.0\}$. Wygenerowany przykład miał $n = 30$ oraz $TF = 0.6$ $RDD =$



Rys. 1. Zbieżność zbadanych metod

0.4. Dla przedstawienia wielu krzywych na jednym wykresie bez użycia kolorów, posłużono się różnymi rodzajami czarnych i szarych linii. Postać linii we wszystkich wykresach jest dla danego algorytmu taka sama (wymieniona ją w punktach 1..6). Linia poziomą zaznaczano wartość optymalną dla testowanej instancji.

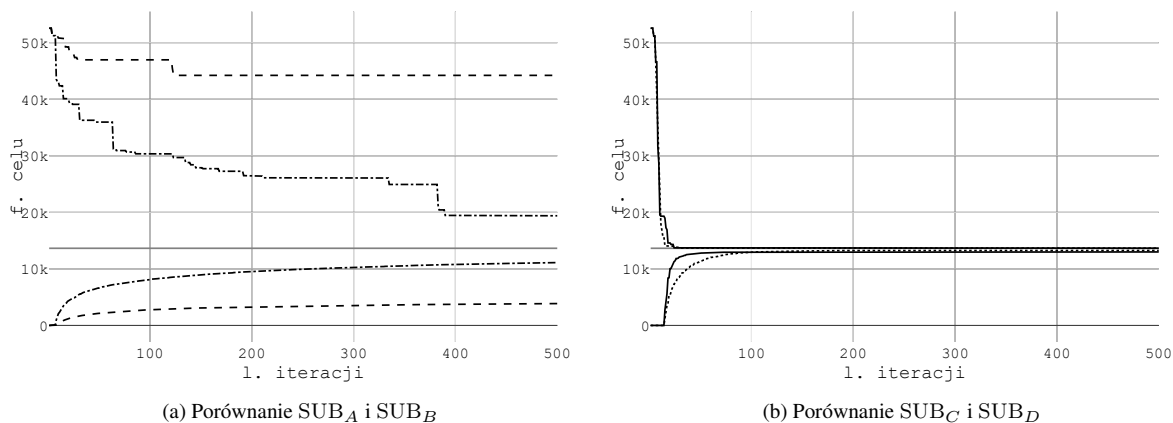
Zachowanie się metod jest różne dla różnych instancji, chociaż obserwowane tendencje zmian są typowe, patrz Rys. 1a. Poprawne wystrojenie metod jest procesem żmudnym i wymaga wielu badań eksperymentalnych. Użycie ciągu harmonicznego (przebieg na Rys. 4a) dla zmian α w metodzie SUB gwarantuje teoretyczną zbieżność, jednak w praktyce jest ona bardzo wolna, ostatecznie pozostawiając dużą lukę (gap) pomiędzy LB i UB, patrz Rys. 1a oraz 2a. Kłopotem jest tutaj właściwy dobór stałej c . Zmniejszenie szybkości zbiegania α w kierunku zera (przebieg na Rys. 4b) zwykle „poprawia” przebieg SUB (ponownie Rys. 1a i 2a).

Bardziej złożony model modyfikacji α za pomocą γ , $W(u)$ i UB (przebiegi na Rys. 4c oraz 4d) mimo braku teoretycznej zbieżności zachowują się zdecydowanie bardziej stabilnie, dostarczając lepszej jakości LB i UB, patrz Rys. 1a oraz 2b. Co ciekawe pomysł oscylujących losowo zmian γ , mający pewien związek z metodą SA, Rys. 1b, jest wyraźnie lepszy od jego deterministycznego ekwiwalent

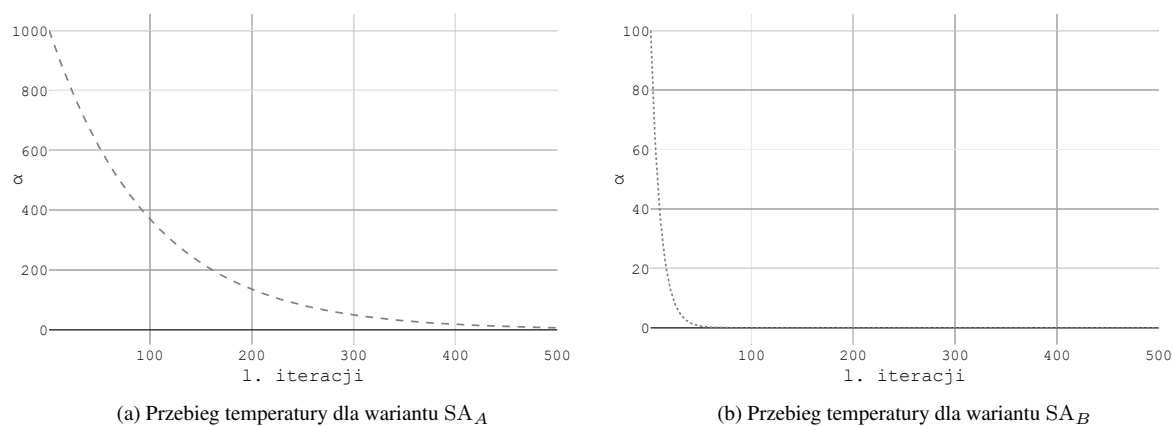
Odnosząc się do SA, pomysł polegający na wykorzystaniu metaheurystyki lokalnego poszukiwania zamiast SUB dla zadania $\max_u W(u)$ okazał się bardzo interesujący. Testowany przykład pokazuje szybką zbieżność metody w porównaniu do SUB, patrz Rys. 1a oraz 1b. Oczywiście, zachowanie się metody SA zależy od przyjętego schematu studzenia (przebiegi dla obu wariantów na Rys. 3). Proces „ręcznego” strojenia SA można ominąć stosując wariant algorytmu SA z autostrojeniem.

7. Wnioski i dalsza praca

W pracy przedstawiono propozycję podejścia dualnego dla problemu minimalizacji ważonej sumy spóźnień, które może być wykorzystane zarówno do obliczania dolnego ograniczenia dla metody podziału i ograniczeń, jak i dla rozwiązywania oryginalnego problemu. Zaproponowano również kilka wariantów algorytmów opartych o metodę subgradientu oraz metaheurystykę symulowanego wyżarzania. Przedstawione przykłady działania algorytmu wskazują na (1) skuteczność omówionego podejścia,



Rys. 2. Zbieżność UB i LB dla wariantów metody SUB



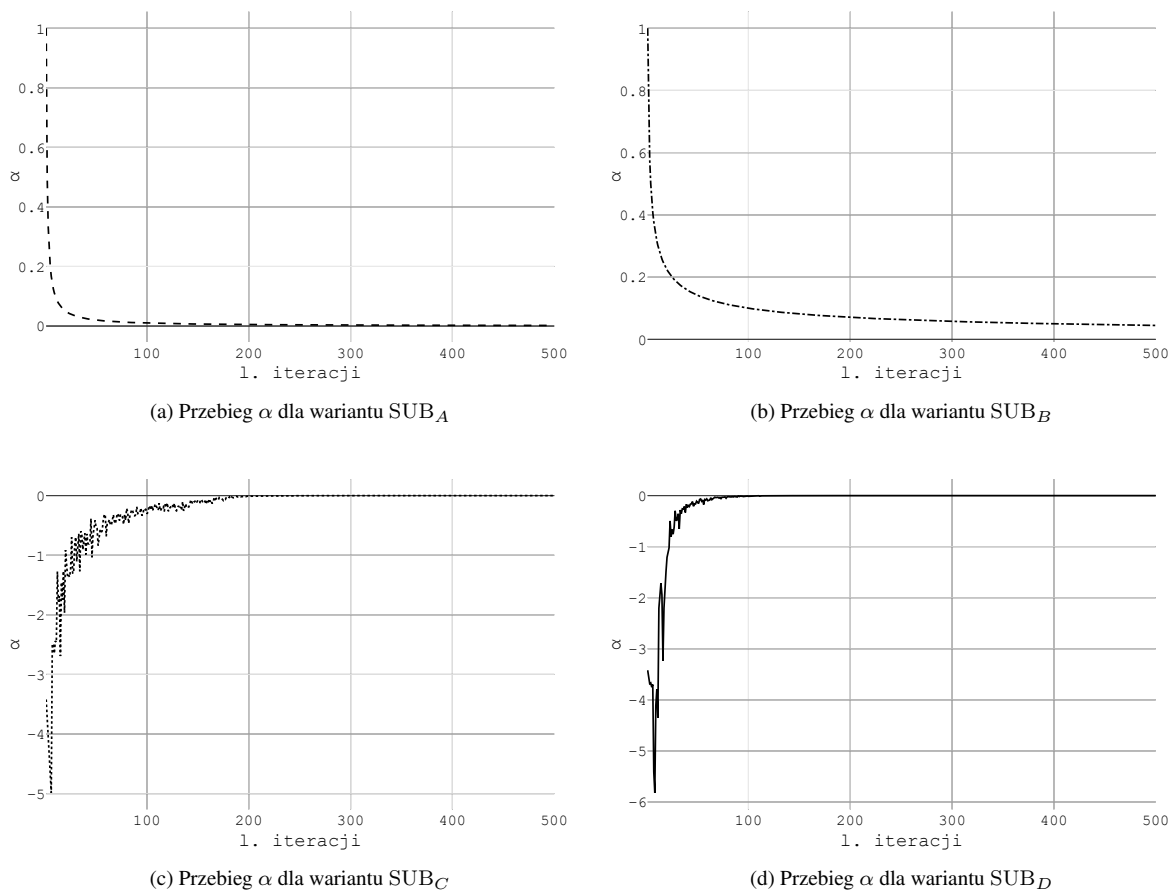
Rys. 3. Schematy studzenia w metodzie SA

(2) konkurencyjność metody symulowanego wyżarzania oraz (3) trudności z doбором parametrów algorytmów.

W związku z obiecującą jakością wyników, planowane są dalsze badania tego tematu obejmujące dwa zasadnicze kierunki. Pierwszy kierunek dotyczy przeprowadzenia obszerniejszych badań na zbiorze instancji testowych o większym rozmiarze, w celu potwierdzenia ogólnej jakości dostarczanych rozwiązań. Drugi kierunek obejmuje prace nad sposobami strojenia proponowanych algorytmów rozwiązania, w szczególności w oparciu o metody hybrydowe. Kolejne pomysły hybrydyzacji zmierzające w kierunku zmniejszenia złożoności problemu dolnego poziomu, zostały już opracowane. Jednak, ze względu na ograniczenie objętości nie zostały włączone do pracy.

LITERATURA

1. Bożejko W., Pempera J., (red.), Optymalizacja dyskretna w informatyce, automatyce i robotyce, Oficyna Wydawnicza Politechniki Wrocławskiej, 2012.
2. Bożejko W., Wodecki M., Parallel algorithm for some single machine scheduling



Rys. 4. Modele zmian α w wariantach metody SUB

problem, *Automatyka*, z. **134**, 2002, 81–90.

3. Bragin M.A., Luh P.B., Yan J.H., Stern G.A., Convergence of the Surrogate Lagrangian Relaxation Method. *Journal of Optimization Theory and Applications*, 164, 2015, 173–201.
4. Butt A.A., Collins R.T., Multi-target Tracking by Lagrangian Relaxation to Min-cost Network Flow. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, 1846–1853.
5. Chen H., Luh P.B., An alternative framework to Lagrangian relaxation approach for job shop scheduling. *European Journal of Operational Research*, 149(3), 2003, 499–512.
6. Cui H., Luo X., An improved Lagrangian relaxation approach to scheduling steelmaking-continuous casting process. *Computers & Chemical Engineering*, 106, 2017, 133–146.
7. Fisher L.M., The Lagrangian Relaxation Method for Solving Integer Programming Problems. *Management Science*, 50(12), 2004, 1861–1871.
8. Fisher, M. L., A dual algorithm for the one-machine scheduling problem. *Mathematical Programming*, 11(1), 1976, 229–251.
9. Fu Y-M., Diabat A., A Lagrangian relaxation approach for solving the integrated

- quay crane assignment and scheduling problem. *Applied Mathematical Modelling*, 39(3–4), 2015, 1194–1201.
10. Gaudioso M., Gorgone E., Labbé M., Rodriguez-Chia A.M., Lagrangian relaxation for SVM feature selection. *Computers & Operations Research*, 87, 2017, 137–145.
 11. Gocgun Y., Ghate A., Lagrangian relaxation and constraint generation for allocation and advanced scheduling. *Computers & Operations Research*, 39(10), 2012, 2323–2336.
 12. Held M., Wolf P., Crowder H.P, Validation of subgradient optimization, *Mathematical Programming*, 6, 1974, 62–88.
 13. Imai A., Nishimura E., Current J., A Lagrangian relaxation-based heuristic for the vehicle routing with full container load. *European Journal of Operational Research*, 176(1), 2007, 87–105.
 14. Nishi T., Hiranaka Y., Inuiguchi M., Lagrangian relaxation with cut generation for hybrid flowshop scheduling problems to minimize the total weighted tardiness, *Computers & Operations Research*, 37(1), 2010, 189–198.
 15. Nowak M.P., Römisch W., Stochastic Lagrangian Relaxation Applied to Power Scheduling in a Hydro-Thermal System under Uncertainty. *Annals of Operations Research*, 100, 2000, 251–272.
 16. Wodecki M., A branch-and-bound parallel algorithm for single-machine total weighted tardiness problem, *The International Journal of Advanced Manufacturing Technology*, 37, 2008, 996–1004
 17. ORlibrary, <http://people.brunel.ac.uk/~mastjjb/jeb/info.html>, dostęp 10-06-2022.
 18. Shaw D.X., Liu S., Kopman L., Lagrangian relaxation procedure for cardinality-constrained portfolio optimization. *Optimization Methods and Software*, 23(3), 2008, 411–420.
 19. Villarreal F.J., Bulfin R.L., Scheduling a single machine to minimize the weighted number of tardy jobs, *IEE Transaction*, 15, 1983, 337–343.
 20. Zhang C., Gao Y., Yang L., Gao Z., Qi J., Joint optimization of train scheduling and maintenance planning in a railway network: A heuristic algorithm using Lagrangian relaxation. *Transportation Research Part B: Methodological*, 134, 2020, 64–92.