STOCHASTYCZNY PROBLEM PRZEPŁYWOWY Z TERMINAMI ZAKOŃCZENIA OPERACJI

Wojciech BOŻEJKO, Paweł RAJBA, Mieczysław WODECKI

Streszczenie: W pracy przedstawiamy problem przepływowy z minimalizacją sumy kar za nieterminowe wykonanie zadań. Niepewne parametry są reprezentowane przez zmienne losowe o rozkładzie normalnym lub Erlanga. Ponieważ problem jest NP-trudny, więc do jego rozwiązania stosujemy algorytm aproksymacyjny oparty na metodzie przeszukiwania z tabu. Przeprowadzone eksperymenty obliczeniowe wykazały dużą odporność rozwiązań na potencjalne zakłócenia parametrów.

Słowa kluczowe: szeregowanie zadań, koszt spóźnienia, rozkład normalny, rozkład Erlanga, poszukiwanie z zabronieniami, stabilność rozwiązań

1. Wprowadzenie

Klasyczne problemy przepływowe są jednymi z prostszych i częściej analizowanych modeli systemów produkcyjnych. Mają praktyczne zastosowanie w pewnych procesach przemysłu mechanicznego, samochodowego oraz w budownictwie. Tworza podstawy do modelowania i analizy bardziej złożonych struktur szeregowo-równoległych dominujących we współczesnych systemach wytwarzania. Są ponadto traktowane jako wskaźniki praktycznych możliwości rozwiązywania trudnych numerycznie problemów przy użyciu narzędzi i metod teorii szeregowania zadań. Istnieje wiele odmian problemów przepływowych różniących się dodatkowymi ograniczeniami. Poza nielicznymi szczególnymi przypadkami problemy te są NP-trudne. Zatem kosztowne obliczeniowo metody dokładne mogą być stosowane tylko dla przykładów o niewielkim rozmiarze lub np. problemów z kryterium czasu cyklu. Dla większych przykładów stosuje sie w praktyce szybkie algorytmy przybliżone. Obecnie najlepsze znane metody pozwalają rozwiązywać problemy z kryterium $C_{\max}$ w czasie kilku minut na PC z dokładnościa poniżej 1-1.5%.

W pracy rozpatrujemy problem przepływowy z minimalizacją sumy kosztów spóźnień (w skrócie *TWT*). Każde z zadań należy kolejno wykonać na wszystkich maszynach, przy czym kolejność wykonywania zadań na każdej maszynie musi być taka sama. Dane są czasy wykonywania zadań, najpóźniejsze żądane terminy zakończenia zadań (na każdej z maszyn) oraz kary za przekroczenie tych terminów. Należy wyznaczyć kolejność wykonywania zadań, która minimalizuje sumę kar. W literaturze problem ten jest oznaczany przez $F||\Sigma\Sigma w_{i,j}U_{i,,j}$ i należy on do klasy problemów NP-trudnych.

Problemy szeregowania zadań z sumokosztowymi funkcjami celu mają już długą historię. Zdecydowana większość prac dotyczy szeregowania na jednej maszynie. Dla problemu przepływowego, jedną z pierwszych jest praca Hariri i Pottsa [7]. Przedstawiono w niej algorytm dla problemu $F||\Sigma w_iU_i$ oparty na metodzie podziału i ograniczeń, którym w rozsądnym czasie można rozwiązywać przykłady do 25 zadań wykonywanych na 3 maszynach. Dla tego problemu, bardzo dobre algorytmy genetyczne są zamieszczone w pracy Bartela i Billauta [2]. W ostatnich latach ukazują się prace głównie poświęcone algorytmom aproksymacyjnym. Oparte na różnych metodach konstrukcji algorytmów najnowsze heurystyki i metaheurystyki (głównie ewolucyjne i symulowanego wyżarzania) są przedstawione przez: Kima [9], Rajendrana i Zieglera [15], Hasija i Rajendrana [8], Adenso-Diasa [1] oraz Onwubolu i Davendra [13]. Przegląd metod, algorytmów i publikacji dotyczący wielomaszynowych problemów szeregowania zadań z minimalizacją sumy kosztów opóźnień jest zamieszczony w pracach Vallady i Ruiza [17] oraz Vallady i in. [18].

Z praktyki zarządzania powszechnie uważa się, że dane są zazwyczaj niepewne i nieprecyzyjne. Ponadto, często zmieniaja swoje wartości juz w trakcie realizacji przyjętego rozwiazania, niszcząc jego optymalność, czasami takze dopuszczalność. Wybór podejścia do modelowania i analizy niepewnych danych wynika z cech systemu, możliwości wykonania pomiarów danych, wiarygodności danych, skuteczności narzędzi teoretycznych, mocy dostępnych pakietów programowych, itp. Znajomość wszystkich wymienionych elementów jest niezbędna do sprawnego rozwiazywania praktycznych problemów. Przykładowo dane dotyczące czasu trwania czynności można przyjąć za deterministyczne (np. normatywne lub ustalone przez eksperta), dokonać pomiaru ich cech losowych (wykonać serię pomiarów oraz zweryfikować hipotezę o typie rozkładu i jego parametrach wartości średniej i wariancji), dokonać pomiaru w celu aproksymowania wartością deterministyczną, dokonać pomiaru w celu określenia funkcji przynależności dla reprezentacji rozmytej, ustalić funkcję przynależności w oparciu o opinię eksperta. Niektóre procesy z natury mają charakter losowy. Zależą od pogody, natężenia ruchu, liczby wypadków, warunków geologicznych, awaryjności sprzętu, itp. Jeżeli ponadto posiadają pewną "historię", więc na bazie istniejących danych statystycznych można określić ich rozkłady.

W pracy rozpatrujemy stochastyczną wersję problemu *TWT*, tj. problem z losowymi czasami wykonywania zadań lub żądanymi najpóźniejszymi terminami zakończenia, o rozkładzie normalnym lub Erlanga. W skrócie problem ten będziemy oznaczali przez *STWT*. Do jego rozwiązywania stosujemy algorytm, którego konstrukcja jest oparta na metodzie przeszukiwania z tabu. Badamy stabilność wyznaczanych przez algorytm rozwiązań, tj. wpływ zmian parametrów zadań na wartości funkcji celu (zobacz także np. Klimek i Łepkowski [10] - rozwiązania proaktywne). Niniejsza praca zawiera pewne uogólnienia wyników badań autorów, dotyczących szeregowania zadań na jednej maszynie w systemach produkcji JIT ([3],[4] i [14]), na problemy wielomaszynowe.

2. Definicje i oznaczenia

W problemie *TWT*, dany jest zbiór n zadań $J=\{1, 2,...,n\}$, które należy wykonać za pomocą m maszyn ze zbioru $M=\{1,2,...,m\}$. Zadanie $i \in J$ należy kolejno wykonać na każdej maszynie, przy czym wykonywanie zadania na maszynie j (operacja $O_{i,j}$) może się rozpocząć dopiero po zakończeniu wykonywania tego zadania na maszynie j-1 (j=2, 3,...,m) oraz kolejność wykonywania zadań na każdej z maszyn musi być taka sama. Maszyna w dowolnej chwili może wykonywać, co najwyżej jedno zadanie oraz, wykonywanie zadania nie może być przerwane. Dla zadania $i \in J$, niech $p_{i,j}$ będzie *czasem wykonywania* na maszynie $j \in M$, $d_{i,j}$ *żądanym terminem zakończenia*, a $w_{i,j}$ *wagą funkcji kary* (za spóźnienie wykonania operacji). Dla ustalonej kolejności zadań, przez $C_{i,j}$ oznaczamy *termin zakończenia* zadania $i \in J$ na maszynie $j \in M$ (operacji $O_{i,j}$). Wówczas,

$$U_{i,j} = \begin{cases} 0, & C_{i,j} \le d_{i,j}, \\ 1, & C_{i,j} > d_{i,j}, \end{cases} \tag{1}$$

jest *spóźnieniem*, a $w_{i,j}U_{i,j}$ *kosztem spóźnienia* wykonania operacji $O_{i,j}$. Należy wyznaczyć kolejność wykonywania zadań (taką samą na każdej maszynie), minimalizującą sumę *kosztów spóźnień*, tj. sumę $\sum_{i=1}^{n}\sum_{j=1}^{m} w_{i,j}U_{i,j}$.

Każde rozwiązanie problemu *TWT* może być reprezentowane przez permutację $\pi = (\pi(1),\ \pi(2),...,\pi(n))$ zadań ze zbioru *J*. Przez Φ oznaczamy zbiór wszystkich permutacji elementów z *J*. Dla dowolnej permutacji $\pi \in \Phi$ terminy zakończenia wykonywania zadań $C_{i,j}$ $(i \in J,\ j \in M)$ na poszczególnych maszynach można wyznaczyć z następujących zależności rekurencyjnych:

$$C_{\pi(i),j} = \begin{cases} \sum_{k=1}^{i} p_{\pi(i),j}, & j = 1, \\ C_{\pi(i),j-1} + p_{\pi(i),j}, & i = 1, j > 1, \\ \max\{C_{\pi(i),j-1}, C_{\pi(i-1),j}\} + p_{\pi(i),j}, & i > 1, j > 1, \end{cases}$$

Przy tych oznaczeniach $C_{\pi(i),m}$ jest terminem zakończenia zadania $\pi(i)$ na ostatniej, *m*-tej maszynie. Jeżeli $C_{i,j} > d_{i,j}$, to zadanie jest *spóźnione*, a w przeciwnym przypadku *terminowe*.

Niech

$$W(\pi) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{\pi(i),j}U_{\pi(i),j}. \tag{2}$$

będzie *kosztem permutacji* π (sumą kar spóźnień). Problem *TWT* sprowadza się więc do wyznaczenia permutacji *optymalnej* (o minimalnym koszcie) w zbiorze wszystkich permutacji Φ.

3. Randomizacja

W praktyce istnieją duże trudności z określeniem rozkładów prawdopodobieństwa zmiennych losowych będących parametrami modelu. Szczególnie, gdy brak jest odpowiednich danych statystycznych. Zazwyczaj w takim przypadku korzystamy z wiedzy eksperta, który ustala zarówno rozkłady, jak i ich parametry. W tym rozdziale rozpatrujemy problem przepływowy, w którym czasy wykonania lub terminy zakończenia zadań na maszynach są zmiennymi losowymi o rozkładzie normalnym lub Erlanga. Pierwszy z rozkładów jest najczęściej stosowany przy modelowaniu "naturalnej" losowości (np. pogoda, popyt, natężenie ruchu, itp), drugi z kolei - awarie maszyn, absencja, błędy pracowników, itp.

Niech $\pi \in \Phi$ będzie pewną kolejnością wykonywania zadań dla problemu *TWT*. Jeżeli czasy wykonywania $\tilde{p}_{i,j}$ lub żądane terminy zakończenia zadań $\tilde{d}_{i,j}$ są niezależnymi zmiennymi losowymi, to spóźnienia $\tilde{U}_{i,j}$ (odpowiednik (1)) oraz funkcja celu będąca odpowiednikiem (2)

$$\tilde{W}(\pi) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{\pi(i),j}\tilde{U}_{\pi(i),j}. \tag{3}$$

są także zmiennymi losowymi.

W algorytmach rozwiązywania problemów optymalizacyjnych porównuje się wartości funkcji celu dla różnych rozwiązań. W przypadku, gdy funkcją tą jest zmienna losowa (3) będziemy stosowali pewne jej momenty lub ich kombinacje. Wstępnie przeprowadzone eksperymenty obliczeniowe wykazały, że najlepsze wyniki otrzymywano, gdy do porównywania rozwiązań była stosowana wartość oczekiwana i odchylenie standardowe. Dlatego też jako kryteria porównawcze rozwiązań będą stosowane dwie funkcje:

$$\Omega_1(\pi) = E(\widetilde{\Omega}(\pi)) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{\pi(i),j} E(\widetilde{U}_{\pi(i),j}), \tag{4}$$

$$\Omega_2(\pi) = E(\widetilde{\Omega}(\pi)) + \delta \cdot D(\widetilde{\Omega}(\pi)) = \sum_{i=1}^{n}\sum_{j=1}^{m} \left(w_{\pi(i),j} E(\widetilde{U}_{\pi(i),j}) + \delta \cdot D(\widetilde{U}_{\pi(i),j}) \right) \tag{5}$$

Parametr δ $(0 < \delta \le 1)$ jest wyznaczany eksperymentalnie.

Niech $(p_{i,j}, d_{i,j}, w_{i,j})_{i=1,2,\ldots,n,\ j=1,2,\ldots,m}$ będzie pewnym ustalonym przykładem danych detrministycznych dla problemu *TWT.* W celu uproszczenia zapisu przyjmujemy, że bieżącym rozwiązaniem problemu jest permutacja naturalna, tj. $\pi = (1,2,\ldots,n)$. W tym rozdziale, podobnie jak w pracy [3]) rozpatrujemy stochastyczne warianianty problemu, w których pewne parametry zadań (czasy wykonywania lub terminy zakończenia) są zmiennymi losowymi:

(A) czasy wykonywania $\widetilde{p}_{i,j}$: $N(p_{i,j}, a \cdot p_{i,j})$, deterministyczne: $w_{i,j}$ oraz $d_{i,j}$,

(B) terminy zakończenia $\widetilde{d}_{i,j}$: $N(d_{i,j}, c \cdot d_{i,j})$, deterministyczne: $p_{i,j}$ oraz $w_{i,j}$,

(C) czasy wykonywania $\widetilde{p}_{i,j}$: $\mathrm{E}(\alpha_{i,j}, \lambda)$, deterministyczne: $w_{i,j}$ oraz $d_{i,j}$,

(D) terminy zakończenia $\widetilde{d}_{i,j}$: $\mathrm{E}(\beta_{i,j}, \mu)$, deterministyczne: $p_{i,j}$ oraz $w_{i,j}$,

gdzie $N(\varepsilon, c \cdot \varepsilon)$ jest rozkładem normalnym o średniej ε i odchyleniu standardowym $c \cdot \varepsilon$ (parametr c jest wyznaczany eksperymentalnie). Dla rozkładów Erlanga E parametr $\alpha_{i,j} = p_{i,j}\lambda,\ \beta_{i,j} = d_{i,j}\mu,$ gdzie $\lambda = 1$ oraz $\mu = \max\{2(\min\{d_{i,j} : 1 \le i \le n\})^{-1}, 1\}$. Łatwo sprawdzić, że dla obu rozkładów wartość oczekiwana $E(\widetilde{p}_{i,j}) = p_{i,j}$. Poszczególne przypadki będziemy w skrócie zapisywali stosując modyfikację notacji Grahama i in. [6]. Dla każdego z nich podamy wzory umożliwiające obliczanie wartości funkcji Ω_1 oraz Ω_2. Łatwo pokazać, że sprowadza się to do obliczenia wartości oczekiwanych zmiennych $E(\widetilde{U}_{i,j})$ oraz $E(\widetilde{U}_{i,j}^2)$.

W dalszej części pracy, jeżeli X jest zmienną losową, to przez F_X oznaczamy jej dystrybuantę. Ponadto, będziemy korzystali z następujących własności.

Lemat 1. Jeżeli $\tilde{p}_k$ $(k = 1, 2, \ldots, n)$ są niezależnymi zmiennymi losowymi, to sumy $\tilde{C}_i = \tilde{p}_1 + \tilde{p}_2 +, \ldots, + \tilde{p}_i$ $(i = 1, 2, \ldots, n)$ są także zmiennymi losowymi, przy czym jeżeli zmienne $\tilde{p}_k$ mają rozkład:

(a) normalny $\widetilde{p}_k$: $N(p_k, \sigma_k)$, to

$$\tilde{C}_i \sim N\left(\sum_{j=1}^{i} p_j, \sqrt{\sum_{j=1}^{i} \sigma_j^2}\right).$$

(b) Erlanga $\widetilde{p}_k :\ \mathrm{E}(\alpha_k, \lambda)$, to

$$\widetilde{C}_i :\ \mathrm{E}(\sum_{j=1}^{i} \alpha_j, \lambda).$$

W dalszej części tego rozdziału rozpatrujemy stochastyczne warianty problemu *TWT*, w których pewne parametry zadań są zmiennymi losowymi:

Przypadek A (problem: $1 | \widetilde{p}_{i,j} :\ N(p_{i,j}, a \cdot p_{i,j}) | \sum_{i=1}^{n} \sum_{j=1}^{m} w_{\pi(i),j} \tilde{U}_{\pi(i),j}$).

Ponieważ

$$\tilde{U}_{i,j} = \begin{cases} 0, & \tilde{C}_{i,j} \le d_{i,j}, \\ 1, & \tilde{C}_{i,j} > d_{i,j}, \end{cases}$$

więc

$$E(\tilde{U}_{i,j}) = P(\tilde{C}_{i,j} > d_{i,j}) = 1 - F_{\tilde{C}_{i,j}}(d_{i,j}),$$

$$D^2(\tilde{U}_{i,j}) = F_{\tilde{C}_{i,j}}(d_{i,j})(1 - F_{\tilde{C}_{i,j}}(d_{i,j})).$$

Ostatecznie

$$\Omega_1(\pi) = \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,j} (1 - F_{\widetilde{C}_{i,j}}(d_{i,j})),$$

$$\Omega_2(\pi) = \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,j} (1 - F_{\widetilde{C}_{i,j}}(d_{i,j})^2).$$

Przypadek B (problem: $1 | \widetilde{d}_{i,j} :\ N(d_{i,j}, a \cdot p_{i,j}) | \sum_{i=1}^{n} \sum_{j=1}^{m} w_{\pi(i),j} \tilde{U}_{\pi(i),j}$).

Z definicji

$$\tilde{U}_{i,j} = \begin{cases} 0, & C_{i,j} \le \tilde{d}_{i,j}, \\ 1, & C_{i,j} > \tilde{d}_{i,j}. \end{cases}$$

Wprowadzamy funkcję pomocniczą

$$\tilde{h}_{i,j} = C_{i,j} - \tilde{d}_{i,j} \ \square \ N(C_{i,j} - d_{i,j}, c\sqrt{C_{i,j}^2 - d_{i,j}^2}$$

i otrzymujemy

$$E(\tilde{U}_{i,j}) = P(C_{i,j} > \tilde{d}_{i,j}) = P(C_{i,j} - \tilde{d}_{i,j} > 0) = P(\tilde{h}_{i,j} > 0) = 1 - F_{\tilde{h}_{i,j}}(0),$$

$$D^2(\tilde{U}_{i,j}) = F_{\tilde{h}_{i,j}}(0)(1 - F_{\tilde{h}_{i,j}}(0)).$$

Kryteria porównawcze rozwiązań

$$\Omega_1(\pi) = \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,j} (1 - F_{\widetilde{h}_{i,j}}(0)),$$

$$\Omega_2(\pi) = \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,j} (1 - (F_{\widetilde{h}_{i,j}}(d_{i,j})^2).$$

Przypadek C (problem: $1 | \widetilde{p}_{i,j} :\ \mathrm{E}(\alpha_{i,j}, \lambda) | \sum_{i=1}^{n} \sum_{j=1}^{m} w_{\pi(i),j} \tilde{U}_{\pi(i),j}$).

Ponieważ

$$\tilde{U}_{i,j} = \begin{cases} 0, & \tilde{C}_{i,j} \le d_{i,j}, \\ 1, & \tilde{C}_{i,j} > d_{i,j}, \end{cases}$$

więc

$$E(\tilde{U}_{i,j}) = P(\tilde{C}_{i,j} > d_{i,j}) = 1 - F_{\tilde{C}_{i,j}}(d_{i,j}),$$

$$D^2(\tilde{U}_{i,j}) = F_{\tilde{C}_{i,j}}(d_{i,j})(1 - F_{\tilde{C}_{i,j}}(d_{i,j}))$$

oraz

$$\Omega_1(\pi) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{i,j}(1 - F_{\tilde{C}_{i,j}}(d_{i,j})),$$

$$\Omega_2(\pi) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{i,j}(1 - (F_{\tilde{C}_{i,j}}(d_{i,j})^2).$$

Przypadek D (problem: $1 | \tilde{d}_{i,j} :\ \mathrm{E}(\beta_{i,j}, \mu) | \sum_{i=1}^{n}\sum_{j=1}^{m} w_{\pi(i),j}\tilde{U}_{\pi(i),j}$).

Ponieważ

$$\tilde{U}_{i,j} = \begin{cases} 0, & C_{i,j} \le \tilde{d}_{i,j}, \\ 1, & C_{i,j} > \tilde{d}_{i,j}, \end{cases}$$

więc

$$E(\tilde{U}_{i,j}) = P(C_{i,j} > \tilde{d}_{i,j}) = F_{\tilde{d}_{i,j}}(C_{i,j}),$$

$$D^2(\tilde{U}_{i,j}) = F_{\tilde{d}_{i,j}}(C_{i,j})(1 - F_{\tilde{d}_{i,j}}(C_{i,j})).$$

i ostatecznie

$$\Omega_1(\pi) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{i,j}(F_{\tilde{d}_{i,j}}(C_{i,j})),$$

$$\Omega_2(\pi) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{i,j}(1 - (F_{\tilde{d}_{i,j}}(C_{i,j}))^2).$$

4. Stabilność rozwiązań oraz algorytmów

Stabilność jest pewną miarą odporności rozwiązania (harmonogramu) na zaburzenia parametrów procesu [14]. Umożliwia szacowanie wpływu ewentualnych zmian występujących w trakcie realizacji zadań na końcowe wyniki.

Niech δ będzie przykładem danych deterministycznych dla problemu *TWT*. Przez $D(\delta)$ oznaczamy zbiór danych generowanych z δ przez zaburzenie (losowo wyznaczone wartości) pewnych parametrów, np. czasów wykonywania zadań. Ponadto, niech AD i AP będą odpowiednio algorytmem deterministycznym i stochastycznym (tj. rozwiązującymi przykłady z parametrami deterministycznymi lub określonymi przez rozkłady zmiennych losowych) dla problemu *TWT*. Rozwiązaniem jest permutacja, a wartością rozwiązania – suma kar za nieterminowe wykonanie zadań (2).

Przez $W(\pi, \delta)$ oznaczamy wartość funkcji kosztu wykonania zadań (2), dla przykładu δ, gdy zadania są wykonywane w kolejności określonej przez permurtację $\pi \in \Phi$. Dalej, niech π_δ^A będzie najlepszym rozwiązaniem wyznaczonym przez algorytm $A \in \{AD, AP\}$ dla danych δ (detrministycznych lub określonych przez zmienne losowe). Wówczas,

$$\Delta(A, \delta, D(\delta)) = \frac{1}{|D(\delta)|} \sum_{\phi \in D(\delta)} \frac{W(\pi_\delta^A, \phi) - W(\pi_\phi^A, \phi)}{W(\pi_\phi^A, \phi)},$$

nazywamy *stabilnością* rozwiązania π_δ^A (tj. rozwiązania przykładu δ wyznaczonego przez algorytm A) na zbiorze danych zaburzonych $D(\delta)$.

Niech Ω będzie zbiorem instancji deterministycznych problemu. *Współczynnik stabilności algorytmu* A na zbiorze danych Ω definiujemy następująco:

$$S(A, \Omega) = \frac{1}{|\Omega|} \sum_{\delta \in \Omega} \Delta(A, \delta, D(\delta)). \tag{8}$$

Parametr ten jest średnią wartością zmian funkcji kosztu (2), gdy kolejność zadań została wyznaczone przez algorytm A dla danych $\delta \in D$, a realizacja jest dla pewnej instancji ze zbioru danych zaburzonych $D(\delta)$. Im wartość ta jest większa tym rozwiązania są mniej stabilne, tzn. mała zmiana parametrów może powodować znaczną zmianę wartości funkcji celu.

5. Eksperymenty obliczeniowe

W pracy Grabowskiego i Wodeckiego [5] został przedstawiony jeden z najlepszych obecnie algorytm przeszukiwania z tabu rozwiązywania problemu przepływowego $F \| C_{\max}$. Został on zaadoptowany do rozwiązywania rozpatrywanych w tej pracy problemów (detrministycznego *TWT* oraz stochastycznego *STWT*) z kryterium $w_{i,j}U_{i,j}$. W skrócie, algorytm deterministyczny będziemy oznaczali przez $\mathrm{A}\Delta$, a stochastyczny przez $\mathrm{A}\Pi$, przy czym przez $\mathrm{A}\Pi_1$ oznaczamy algorytm stochastyczny z kryterium wyboru elementu z otoczenia Ω_1 (4), a przez $\mathrm{A}\Pi_2$ - z kryterium Ω_2 (5). Algorytmy zostały zaprogramowane w języku C++ i uruchomione na komputerze osobistym z procesorem 2.4 GHz. Obliczenia wykonano na bazie sześciu grup przykładów Taillarda (dla problemu $F \| C_{\max}$) zamieszczonych na stronie OR Library [12]. Każda grupa *n*x*m*: 20x5, 20x10, 20x20, 50x5, 50x10, 50x20 zawiera 10 przykładów (w sumie 60 przykładów). Wagi funkcji kary $w_{i,j}$ były generowane według rozkładu jednostajnego ze zbioru {1,2,…,10}, a żądane terminy zakończenia zadań wyznaczono stosując następująca procedurę (Taillard [16]):

Krok 1: Obliczyć

$$P_{i,j} = \sum_{k=1}^{j} p_{i,k},\ i = 1,2,...,n,\ \ j = 1,2,...,m.$$

Krok 2: Dla każdej operacji $O_{i,j}$ wyznaczyć żądany termin zakończenia

$$d_{i,j} = \lceil P_{i,j}(1 + 3\mu) \rceil.$$

Parametr μ jest realizacją zmiennej losowej o rozkładzie jednostajnym na odcinku [0,1]. Podobny sposób generowania danych jest opisany w pracy Hasija i Rajendrana [8]. Zbiór wszystkich 60 przykładów deterministycznych oznaczamy przez Ω.

Dla każdego przykładu danych deterministycznych wyznaczono przykład danych probabilistycznych z następującymi rozkładami zmiennych losowych (pozostałe wartości są deterministyczne):

a) czasy wykonywania: $\tilde{p}_{i,j} \sim N(p_{i,j}, \alpha \cdot p_{i,j})$, gdzie $\alpha = 0.2$,

b) najpóźniejsze terminy zakończenia: $\tilde{d}_{i,j} \sim N(d_{i,j}, c \cdot d_{i,j})$, $c = 0.1$,

c) czasy wykonywania: $\tilde{p}_{i,j}$: $\mathrm{E}(\alpha_{i,j}, \lambda_\alpha)$, $\alpha_{i,j} = p_{i,j} \cdot \lambda_\alpha$, $\lambda_\alpha = 1$,

d) najpóźniejsze terminy zakończenia: $\tilde{d}_{i,j}$: $\mathrm{E}(\beta_{i,j}, \mu_\beta)$, $\beta_{i,j} = d_{i,j} \cdot \mu_\beta$, $\mu_\beta = 1$,

Wartości parametrów $\alpha = 0.2$ (przypadek (a)), $c = 0.1$ (przypadek (b)) oraz $\lambda_\alpha = \mu_\beta = 1$ (przypadek (c) i (d)) ustalono na podstawie znanych danych statystycznych dotyczących budownictwa drogowego oraz wielu przeprowadzonych eksperymentów obliczeniowych. Dla uproszczenia, każdy z tych zbiorów danych (odpowiednio dla przypadku (a), (b), (c) i (d)) oznaczymy przez $\tilde{\Omega}$.

Permutacja startowa.
Rozwiązanie startowe algorytmów, których konstrukcję oparto na metodzie przeszukiwania z tabu, wyznaczono korzystając z algorytmu *NEH* (Nawaz, Enscore, Ham [11]).

Parametry algorytmów.
Biorąc po dwa przykłady z każdej grupy ustalono następujące wartości parametrów:

- długość listy tabu, $lTS(iter) = \lceil \sqrt{n} \rceil$,
- liczba iteracji algorytmu, $Maxiter = n^2$,
- współczynnik $\delta = 0,1$ w kryterium (5).

5.1 Wyniki obliczeniowe

Ponieważ dla rozpatrywanego problemu nie ma w literaturze przykładowych danych porównawczych, dlatego ograniczamy się do porównania jedynie pomiędzy sobą wyników przedstawionych w pracy algorytmów. W pierwszej kolejności zbadano jakość wyznaczanych przez poszczególne algorytmy rozwiązań. Dla algorytmów *STWT* przyjęto, że czasy wykonywania zadań sa zmiennymi losowymi o rozkładzie normalnym (zobacz przypadek (a)). Porównaliśmy wyniki algorytmu deterministycznego $\mathrm{A}\Delta$ oraz obu probabilistycznych $\mathrm{A}\Pi_1$ i $\mathrm{A}\Pi_2$ z rozwiązaniami wyznaczonymi przez algorytm NEH. Dla dowolnego przykładu, niech F^{A} będzie wartością funkcji celu dla rozwiązania wyznaczonego przez algorytm $\mathrm{A} \in \{NEH, \mathrm{A}\Delta, \mathrm{A}\Pi_1, \mathrm{A}\Pi_2\}$ i $(F^{\mathrm{A}} - F^{NEH})/F^{NEH} \cdot 100\%$ – *procentowym błędem względnym* (poprawą) rozwiązania F^{A} względem wartości rozwiązania F^{NEH} wyznaczonym przez algorytm *NEH.*

W Tabeli 1 przedstawiono średnie δ_{aprd} oraz maksymalne δ_{mrpd} procentowe poprawy (błędy względne) dla poszczególnych grup danych.

Tabela 1. Średnia względna poprawa rozwiązania wyznaczonego przez algorytm NEH.

nxm	Algorytm AΔ		Algorytm AΠ$_1$		Algorytm AΠ$_2$	
	δ_{aprd}	δ_{mrpd}	δ_{aprd}	δ_{mrpd}	δ_{aprd}	δ_{mrpd}
20x5	6,12	9,41	4,14	7,37	3,28	8,16
20x10	5,24	8,78	5,07	11,03	3,94	12,79
20x20	9,57	16,04	5,98	10,11	3,78	9,54
50x5	8,99	11,63	7,11	13,21	4,16	8,77
50x10	11,36	19,88	6,24	9,88	5,90	11.02
50x20	13,68	18,53	7,91	14,06	4,37	9,16
średnio	**9,16**	**14,04**	**6,07**	**10,94**	**4,23**	**9,68**

Porównując średni błąd względny δ_{aprd}, okazuje się, że niezależnie od liczby zadań i maszyn, algorytm deterministyczny AΔ wyznacza zdecydowanie lepsze rozwiązania niż oba algorytmy probabilistyczne. W tym przypadku, średnia poprawa rozwiązań algorytmu AΔ wynosi 9,16% i jest zdecydowanie większa niż algorytmów probabilistycznych. Szczególnie jest to widoczne dla przykładów o większej liczbie zadań. Średnie maksymalne błędy δ_{mrpd} zachowują podobne proporcje. Czas obliczeń jednego algorytmu, wszystkich 60 przykładów, nie przekracza 5 sek. (na PC z zegarem 2.4Ghz).

5.2 Stabilność algorytmów

Aby zbadać stabilność algorytmów wygenerowano odpowiednie zbiory danych zaburzonych. Podstawą było 60 przykładów (Taillarda) danych deterministycznych ze zbioru Ω opisanych powyżej). Dla przykładu danych deterministycznych $\delta = (p_{i,j}, w_{i,j}, d_{i,j})_{i=1,2,\ldots,n,\, j=1,2,\ldots,m}, \delta \in \Omega$ wygenerowano losowo po 100 przykładów danych zaburzonych $D(\delta)$ zgodnie z następującymi rozkładami:

a) problem $1 \,|\, \tilde{p}_{i,j} \sim N(p_{i,j}, a \cdot p_{i,j}) \,|\, \sum\sum w_{i,j} U_{i,j}$, losowe czasy wykonywania operacji $\tilde{p}_{i,j} \sim N(p_{i,j}, 0.2 \cdot p_{i,j})$,

b) problem $1 \,|\, \tilde{d}_{i,j} \sim N(d_{i,j}, c \cdot d_{i,j}) \,|\, \sum\sum w_{i,j} U_{i,j}$, losowe żądane terminy zakończenia operacji $\tilde{d}_{i,j} \sim N(d_{i,j}, 0.1 \cdot d_{i,j})$,

c) problem $1 \,|\, \tilde{p}_{i,j} \sim \mathrm{E}(\alpha_{i,j}, \lambda) \,|\, \sum\sum w_{i,j} U_{i,j}$, losowe czasy wykonywania operacji $\tilde{p}_i \sim \mathrm{E}(p_{i,j}, 1)$,

d) problem $1 \,|\, \tilde{d}_{i,j} \sim \mathrm{E}(\beta_{i,j}, \chi) \,|\, \sum\sum w_{i,j} U_{i,j}$, losowe żądane terminy zakończenia operacji $\tilde{d}_{i,j} \sim \mathrm{E}(d_{i,j}, 1)$.

W sumie, dla każdego wariantu problemu probabilistycznego (**a**), (**b**), (**c**) oraz (**d**) wygenerowano 2 400 przykładów danych zaburzonych. Zostały one następnie rozwiązane przez algorytm AΔ , które to rozwiązania stanowiły bazę do wyznaczenia współczynnika stabilności badanych algorytmów.

W Tabeli 2 zamieszczono wyniki dla problemów z losowymi parametrami o rozkładzie normalnym. Oba algorytmy probabilistyczne mają współczynnik stabilności zdecydowanie mniejszy niż algorytm deteministyczny. Dla losowych czasów wykonywania zadań algorytm probabilistyczny $A\Pi_1$ ma współczynnik stabilności $3,55\%$ i jest on prawie dwukrotnie mniejszy od $6,21\%$ - współczynnika algorytmu deterministycznego $A\Delta$. Współczynnik $3,55\%$ algorytmu $A\Pi_1$ oznacza, że losowe zaburzenie czasów wykonywania zadań powoduje pogorszenie wartości funkcji celu (w stosunku do rozwiązań algorytmu $A\Delta$) średnio o $3,55\%$. Algorytm $A\Pi_2$ jest tylko nieznacznie gorszy od $A\Pi_1$.

W przypadku losowych terminów zakończenia zadań najmniejszy współczynnik stabilności ma algorytm $A\Pi_2$. Wynosi on bowiem $6,28\%$. Nieznacznie większy $6,82\%$ ma algorytm $A\Pi_1$. Największy, $10,16\%$ ma algorytm deterministyczny $A\Delta$.

Tabela 2. Współczynnik stabilności $S(A,\Omega)$ dla przykładów z losowymi parametrami o rozkładzie normalnym (przypadek **A** oraz **B**).

nxm	**Przypadek A** ($\tilde{p}_{i,j} \sim N(p_{i,j}, a\times p_{i,j})$)			**Przypadek B** ($\tilde{d}_{i,j} \sim N(d_{i,j}, c\times d_{i,j})$)		
	$A\Delta$	$A\Pi_1$	$A\Pi_2$	$A\Delta$	$A\Pi_1$	$A\Pi_2$
20x5	4,17	2,21	2,04	4,62	4,24	3,54
20x10	4,61	2,18	2,18	5,99	3,98	3,15
20x20	6,13	3,47	3,48	8,11	6,54	6,37
50x5	6,97	3,82	4,56	12,89	7,12	7,55
50x10	8,06	4,54	5,17	14,13	9,26	8,12
50x20	7,31	5,11	5,84	15,22	9,77	8,96
Średnio	**6,21**	**3,55**	**3,89**	**10,16**	**6,82**	**6,28**

Wyniki obliczeniowe dla dla przykładów z czasami wykonywania lub terminami zakończenia zadań o rozkładzie Erlanga zamieszczono w Tabeli 3.

Tabela 3. Współczynnik stabilności $S(A,\Omega)$ dla przykładów z losowymi parametrami o rozkładzie Erlanga (przypadek **C** oraz **D**).

nxm	**Przypadek C** ($\tilde{p}_{i,j} :\ \mathrm{E}(\alpha_{i,j},\lambda)$)			**Przypadek D** ($\tilde{d}_{i,j} :\ \mathrm{E}(\beta_{i,j},\mu)$)		
	$A\Delta$	$A\Pi_1$	$A\Pi_2$	$A\Delta$	$A\Pi_1$	$A\Pi_2$
20x5	2,56	1,83	2,01	4,89	3,34	2,86
20x10	3,16	1,97	1,62	5,11	3,82	3,44
20x20	4,27	2,64	3,07	7,93	5,56	4,73
50x5	5,08	3,37	3,12	9,36	7,28	8,12
50x10	7,11	4,27	4,68	13,17	9,88	7,87
50x20	9,48	5,13	5,26	16,58	9,66	10,08
Średnio	**5,28**	**3,20**	**3,29**	**9,50**	**6,59**	**6,18**

Dla przypadków, gdy parametry zadań są zmiennymi losowymi o rozkładzie Erlanga, stabilność rozwiązań (Tabela 3) jest podobna jak dla rozkładu normalnego (Tabela 2). Oba algorytmy probabilistyczne są znacznie stabilniejsze od algorytmu deterministycznego i to zarówno dla losowych czasów wykonywania zadań, jak i żądanych terminów zakończenia.

Reasumując, na podstawie zamieszczonych wyników można stwierdzić, że algorytmy probabilistyczne są znacznie stabilniejsze. Wyznaczane przez nie rozwiązania są zdecydowanie mniej wrażliwe na ewentualne losowe zmiany parametrów problemu. Dla losowych czasów wykonywania zadań stabilniejszy jest algorytm $A\Pi_1$, a dla czasów zakończenia zadań - nieznacznie algorytm $A\Pi_2$.

6. Uwagi i wnioski

W pracy rozpatrywano sumokosztowy problem przepływowy z niepewnymi danymi reprezentowanymi przez zmienne losowe o rozkładzie normalnym lub Erlanga. Rozkłady te dobrze opisuje "naturalną" losowość, z jaką najczęściej mamy do czynienia w praktyce zarządzania. Do rozwiązywania problemów zastosowano algorytm, którego konstrukcja jest oparta na metodzie przeszukiwania z tabu. Przeprowadzono eksperymenty obliczeniowe w celu zbadania stabilności algorytmów, tj. wpływu zaburzeń parametrów problemu na zmiany wartości optymalizowanego kryterium. Otrzymane wyniki jednoznacznie wskazują, że znacznie stabilniejsze są algorytmy probabilistyczne tj. algorytmy, w których za kryterium porównawcze rozwiązań przyjęto funkcje momentów centralnych losowej funkcji celu. Zastosowanie elementów probabilistyki w adaptacji metody przeszukiwania z tabu pozwala skutecznie rozwiązywać duże przykłady z niepewnymi danymi dla wielu trudnych praktycznych zagadnień optymalizacyjnych. Podobne wnioski, wynikające z badań dotyczących jednomaszynowego problemu szeregowania zadań z żądanymi terminami zakończenia zadań, zostały zamieszczone w pracy [3].

Praca została częściowo sfinansowana ze środków Narodowego Centrum Nauki przyznanych na podstawie decyzji numer DEC-2012/05/B/ST7/00102.

Literatura

1. Adenso-Dias B., Restricted neighborhood in the tabu search for the flowshop problem, European Journal of Operational Research, 62, 1992, 27-37.
2. Bertel S., Billaut J.C., A genetic algorithm for an industrial multiprocessor flowshop scheduling problem with recirculation, European Journal of Operational Research, 159, 2004, 651-662.
3. Bożejko W. Rajba P. Wodecki M., Scheduling Problem with Uncertain Parameters in Just in Time System; 13th Intern. Confer. ICAISC 2014 Zakopane; Proceed. Vol 2; Springer, 2014, 456-467.
4. Bożejko W., Rajba P., Wodecki M., Scheduling problem with uncertain parameters, Proceedings of the International Conference on ICT Management for Global Competitiveness and Economic Growth in Emerging Economies ICTM 2012, 262-273.
5. Grabowski J., Wodecki M., A very fast tabu search algorithm for the permutation flow shop problem with makespan criterion, Computers and Operations Research, 31, 2004, 1891-1909.
6. Graham R.L., Lawler E.L., Lenstra J.K, Rinnooy Kan A.H.G., Optimization and approximation in deterministic sequencing and scheduling, Annals of Discrete Mathematics, 5, 1979, 287-326.
7. Hariri A.M.A., Potts C.N., Branch and Bound Algorithm to Minimize the Number of Late Jobs in a Permutation Flow shop, European Journal of Operational Research, 38, 1989, 228-237.

8. Hasija S., Rajendran C., Scheduling in flowshop to minimize total tardiness of jobs, International Journal of Production Research, 42(11), 2004, 2289-2301.
9. Kim Y.D., Heuristics for flowshop scheduling problemsw minimizing mean tardiness, Journal of the Operational Research Society, 44, 1993, 19-28.
10. Klimek M., Łepkowski P., Alokacja zasobów dla problemów harmonogramowania projektu z niestabilnymi kosztami niestabilności, Automatyka t. 5, z. 2, 2011, 237-245.
11. Nawaz M., Enscore E.E., Ham I., A heuristic algorithm for the *m*-machine, *n*-job flow-shop sequencing problem, OMEGA, 11, 1983, 91-95.
12. OR Library http://www.ms.ic.ac.uk/info.html
13. Onwubolu G., Davendra D., Scheduling flow shop using differential evolution algorithm, European Journal of Operational Research, 171, 2006, 674-692.
14. Rajba P., Wodecki M., Stability of scheduling with random processing times on one machine, Applicationes Mathematicea, 39, 2, 2012, 169-183.
15. Rajendran C., Ziegler H., Scheduling to minimize the sum of weighted flowtime and weighted tardiness of jobs in a flowshop with sequence-dependent setup times, European Journal of Operational Research, 147(3), 2003, 513-522.
16. Taillard E., Benchmarks for basic scheduling problems, European Journal of Operational Research, 64, 1993, 278-285.
17. Valada E., Ruiz R., New genetic algorithms with path relinking for the minimum tardiness permutation flow shop problem, 2006
18. Valada E., Ruiz R., Millea G., Minimizing total tardiness in the m-machine flowshop problem: a review and evaluation heuristics and metaheuristics, Computers & Operations Research, 2008, 1350-1373.

Dr hab. Wojciech BOŻEJKO, prof. nadzw.
Katedra Automatyki, Mechatroniki i Systemów Sterowania
Wydział Elektroniki, Politechnika Wrocławska
Wyb. Wyspiańskiego 27, 50-370 Wrocław
e-mail: wojciech.bozejko@pwr.wroc.pl

Mgr Paweł RAJBA
Dr hab. Mieczysław WODECKI, prof. nadzw.
Instytut Informatyki Uniwersytetu Wrocławskiego
ul. Joliot-Curie 50-383 Wrocław
e-mail: pawel.rajba@ii.uni.wroc.pl
mieczyslaw.wodecki@ii.uni.wroc.pl