Autoreferat

1. Dane kontaktowe

Imię i nazwisko	Krzysztof Fleszar
Adres	American University of Beirut, Olayan School of Business PO BOX 11-0236, Riad El Solh, Beirut 1107 2020, Lebanon
Email	kf09@aub.edu.lb
Strona	http://staff.aub.edu.lb/~kf09/

2. Posiadane dyplomy i stopnie naukowe

- Doktor nauk technicznych w zakresie informatyki. Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych. Tytuł rozprawy: *Techniki konstrukcji i redukcji w algorytmach szeregowania i alokacji*. 25.01.2005.
- Magister inżynier w zakresie informatyki. Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych. Tytuł pracy: *Zastosowanie analizy wielokryterialnej do wspomagania decyzji strategicznych*. 19.10.2001.

3. Informacje o zatrudnieniu

2005-obecnie	American University of Beirut, Olayan School of Business Associate Professor (od 2010) Assistant Professor (do 2010) Convener (Chairman) of Business Information and Decision Systems Track (kierownik Specjalności Systemów Informacyjnych i Decyzyjnych, od 2011)
2003-2005	Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych Instytut Automatyki i Informatyki Stosowanej Adiunkt (od marca 2005) Asystent (do lutego 2005)
2000-2001	SAS Institute Polska, Junior Consultant
1999-2000	Brunel University of West London, Systems Engineering Department Research Student

4. Wykaz publikacji będących podstawą wniosku habilitacyjnego

Poniższe publikacje wchodzą w skład osiągnięcia, o którym mowa w art. 16 ustawy z dnia 14 marca 2003 r. o stopniach naukowych i tytule naukowym oraz o stopniach i tytule w zakresie sztuki. Przy każdej pozycji podany jest Impact Factor (IF) czasopisma oraz liczba cytowań (jeśli niezerowa) według Web of Science. Pod każdą pozycją określony jest również indywidualny wkład wnioskodawcy w autorstwo publikacji. Pozostały dorobek wnioskodawcy przedstawiony jest w osobnych załącznikach.

[1] **Fleszar K**, Charalambous C. Average-Weight-Controlled Bin-Oriented Heuristics for One-Dimensional Bin-Packing Problem. European Journal of Operational Research 210, 176-184, Apr 2011, **IF=1.815, 3 cytowania**
Wkład własny: 70%, pomysły, zaprojektowanie i implementacja algorytmów, testy, przygotowanie artykułu.

[2] Charalambous C, **Fleszar K**. A constructive bin-oriented heuristic for the two-dimensional bin packing problem with guillotine cuts. Computers and Operations Research 38(10), 1443-1451, Oct 2011, **IF=1.720, 1 cytowanie**
Wkład własny: 40%,pomysły i zaprojektowanie algorytmów, przygotowanie artykułu.

[3] **Fleszar K**. Three insertion heuristics and a justification improvement heuristic for two-dimensional bin packing with guillotine cuts. Computers and Operations Research 40(1), pp. 463-474, Jan 2013, **IF=1.720**
Wkład własny: 100%, pomysły, zaprojektowanie i implementacja algorytmów, testy, przygotowanie artykułu.

[4] **Fleszar K**, Osman IH, Hindi KS. A variable neighbourhood search algorithm for the open vehicle routing problem. European Journal of Operational Research 195(3), 803-809, Jun 2009, **IF=1.815, 18 cytowań**
Wkład własny: 70%, pomysł, zaprojektowanie i implementacja algorytmu, testy, przygotowanie artykułu.

[5] **Fleszar K**, Hindi KS. An effective VNS for the capacitated p-median problem. European Journal of Operational Research 191(3), 612-622, Dec 2008, **IF=1.815, 10 cytowań**
Wkład własny: 80%, pomysł, zaprojektowanie i implementacja algorytmu, testy, przygotowanie artykułu.

5. Opis cyklu publikacji będących podstawą wniosku habilitacyjnego

„Przybliżone metody rozwiązywania problemów alokacji"

Podstawą wniosku habilitacyjnego jest cykl publikacji na temat przybliżonych metod rozwiązywania różnych problemów alokacji. Obejmuje on prace nad problemami pakowania jednowymiarowego [1] i dwuwymiarowego [2], [3], otwartym problemem planowania dostaw [4] oraz problemem lokalizacji *p* punktów obsługi o ograniczonych pojemościach [5]. Wszystkie te problemy łączy fakt, że istotnym elementem ich rozwiązania jest przypisanie pewnego rodzaju obiektów do pewnego rodzaju pojemników o ograniczonych pojemnościach. W problemach pakowania obiekty pakowane są w pojemnikach, w problemie planowania dostaw klienci przypisywani są do pojazdów, a w problemie lokalizacji punktów obsługi klienci przypisywani są do punktów obsługi. Poniżej opisane są główne osiągnięcia przedstawione w powyższych artykułach.

<u>Pakowanie jednowymiarowe</u>

Jednowymiarowy problem pakowania (ang. one-dimensional bin packing) polega na zapakowaniu zbioru obiektów o zadanych wagach w jak najmniejszej liczbie jednakowych pojemników w taki sposób, aby sumaryczna waga obiektów w każdym pojemniku nie przekroczyła pojemności pojemnika. Problem ten ma bezpośrednie zastosowania, np. w problemie załadunku z ograniczeniem wagowym. Występuje on też często jako podproblem w wielu bardziej złożonych problemach. Praca [1] nad metodami rozwiązywania tego problemu jest kontynuacją pracy [14] wykonanej jeszcze przed ukończeniem doktoratu, dlatego najpierw przedstawione zostaną pokrótce wyniki pracy [14].

Inspiracją pracy [14] była metoda Gupty i Ho [31] zwana minimum bin slack (MBS), która konstruuje rozwiązanie w następujący sposób: wykonaj enumerację wszystkich możliwych podzbiorów obiektów mieszczących się w pierwszym pojemniku i umieść w nim podzbiór, który wypełnia go najlepiej; następnie powtórz tę operację dla pojemnika drugiego, trzeciego itd., aż zapakowane zostaną wszystkie obiekty. Aby ograniczyć czas enumeracji, obiekty rozważane są w kolejności nierosnących wag. Dodatkowo enumeracja jest przerywana, gdy tylko znaleziony zostanie podzbiór całkowicie wypełniający pojemnik. Pomimo powyższych zabiegów metoda MBS ma wykładniczą złożoność obliczeniową. Jednakże w wielu trudnych instancjach problemu pakowania czas obliczeń MBS jest bardzo krótki, a uzyskiwane wyniki są lepsze niż wyniki prostych metod takich jak first-fit-decreasing (FFD) lub best-fit-decreasing (BFD) [32].

W artykule [14], opublikowanym przed ukończeniem doktoratu razem z prof. Hindim, zaproponowaliśmy szereg metod pakowania jednowymiarowego bazujących na MBS. Po pierwsze, w oryginalnej metodzie MBS wprowadziliśmy kilka usprawnień, które ograniczają czas enumeracji poprzez eliminację zdominowanych podzbiorów, lecz nie zmieniają uzyskiwanych wyników. Po drugie, zaproponowaliśmy modyfikację metody MBS, nazwaną MBS', w której czas enumeracji został zredukowany w następujący sposób: zanim rozpoczęta zostanie enumeracja, umieść w pojemniku największy dostępny obiekt, a następnie wykonaj enumerację możliwych podzbiorów wypełniających pozostałe miejsce w pojemniku. Eksperymenty obliczeniowe pokazały, że taki zabieg poprawia średnią jakość wyników, a jednocześnie skraca średni czas obliczeń. Po trzecie, na bazie metody MBS', zaprojektowaliśmy trzy metody poprawy rozwiązań: Relaxed MBS', Perturbation MBS', oraz Sampling MBS'. Z powyższych metod, najciekawsze wyniki uzyskała Perturbation MBS', która wielokrotnie modyfikuje rozwiązanie początkowe w następujący sposób: wybierz losowo obiekt zapakowany w pojemniku o stosunkowo dużym wolnym miejscu, przenieś ten obiekt do nowego pojemnika, uruchom enumerację wykorzystującą wszystkie pozostałe obiekty, aby znaleźć podzbiór najlepiej wypełniający pozostałe miejsce w nowym pojemniku, przenieś obiekty wyznaczonego podzbioru do nowego pojemnika i usuń z rozwiązania wszystkie powstałe puste pojemniki. Wielokrotne wykonanie powyższej perturbacji w niektórych instancjach prowadzi do poprawy rozwiązania początkowego. Perturbation MBS' była jedyną z metod zaproponowanych w pracy [14] rozwiązującą wszystkie instancje testowe typu triplets. Instancje te są wyjątkowo trudne dla wielu heurystyk, gdyż zostały wygenerowane w taki sposób, by istniały rozwiązania optymalne, w których każdy pojemnik wypełniony jest całkowicie przez dokładnie trzy obiekty. Większość prostych heurystyk nie rozwiązuje optymalnie prawie żadnej instancji typu triplets, a Perturbation MBS' rozwiązuje je wszystkie. Po czwarte, w artykule [14] przedstawiliśmy zastosowanie metaheurystyki Variable Neighborhood Search (VNS) do rozwiązywania problemu pakowania jednowymiarowego. VNS jest ogólną metodą rozwiązywania trudnych problemów optymalizacji, zaproponowaną przez Hansena i Mladenovića [33], wykorzystującą optymalizację lokalną oraz stosującą losowe przesunięcia o rosnących wielkościach w celu ucieczki z optimum lokalnego. W eksperymentach obliczeniowych zaprezentowanych w pracy [14] najlepsze wyniki uzyskane zostały z wykorzystaniem kombinacji metod MBS' + Perturbation MBS' + VNS.

W pracy [1], wykonanej po doktoracie we współpracy z dr Charalambousem, zaproponowaliśmy prostą zasadę, której wykorzystanie w heurystykach pakowania jednowymiarowego zorientowanych na pojemnik (takich jak np. MBS czy MBS') pozwala na znaczą poprawę rozwiązań oraz redukcję czasu obliczeń. Heurystyki *zorientowane na pojemnik* (ang. bin-oriented) zostały zdefiniowane jako heurystyki, które w każdej fazie do częściowo skonstruowanego rozwiązania dodają jeden nowy pojemnik i wybierają podzbiór obiektów, który zostanie do niego zapakowany. Rozpoczynają one pakowanie nowego pojemnika tylko wtedy, gdy zbiór obiektów dla wcześniejszego pojemnika zostanie ostatecznie zatwierdzony. Warto zauważyć, że typowe szybkie heurystyki pakowania jednowymiarowego, takie jak first-fit-decreasing (FFD) lub best-fit-decreasing (BFD) [32], są *zorientowane na obiekt* (ang. item-oriented): w każdej fazie rozważany jest pojedynczy obiekt, a heurystyka wybiera jeden z dostępnych pojemników (lub dodaje nowy pojemnik), do którego rozważany obiekt zostaje zapakowany.

W pracy [1] opisaliśmy cztery wcześniej zaproponowane heurystyki pakowania zorientowane na pojemnik: odpowiednio zaprogramowana heurystyka FFD, best-2-fit (B2F) [34], MBS [31] oraz MBS' [14]. Heurystyka FFD zorientowana na pojemnik wypełnia pojedynczy pojemnik dodając obiekty w kolejności nierosnących wag, pomijając obiekty, dla których nie ma wystarczająco miejsca.

Heurystyka B2F konstruuje dwa podzbiory i wybiera ten, który lepiej wypełnia pojemnik. Pierwszy podzbiór konstruowany jest metodą FFD, a drugi konstruowany jest przez modyfikację pierwszego, w której ostatni dodany obiekt jest zastępowany dwoma mniejszymi dostępnymi obiektami jak najlepiej wypełniającymi dostępne miejsce. Heurystyka MBS rozważa dla każdego pojemnika wszystkie możliwe podzbiory dostępnych obiektów i pakuje ten, który pozostawia w pojemniku najmniej wolnego miejsca. Heurystyka MBS' działa podobnie do MBS, lecz ogranicza wybór tylko do tych podzbiorów, które zawierają największy aktualnie dostępny obiekt.

Warto zauważyć, że wszystkie cztery heurystyki zorientowane na pojemnik różnią się jedynie zakresem przeszukiwania możliwych podzbiorów dla pojedynczego pojemnika: FFD rozważa tylko jeden podzbiór, B2F rozważa dwa podzbiory, MBS' rozważa wszystkie podzbiory zawierające największy niezapakowany obiekt, a MBS rozważa wszystkie możliwe podzbiory. Warto także zauważyć, że pierwsze dwie metody mają złożoność obliczeniową $O(n^2)$ (n = liczba obiektów), a pozostałe dwie metody zaprogramowane z wykorzystaniem rekursji mają złożoność $O(2^n)$, przy czym jeśli w pojemniku mieści się maksymalnie u obiektów, to złożoność MBS redukuje się do $O(n^{u+1})$, a złożoność MBS' redukuje się do $O(n^u)$. Zarówno MBS jak i MBS' mogą być również zaprogramowane z wykorzystaniem programowania dynamicznego o złożoności pseudo-wielomianowej $O(n^2c)$ (c = pojemność pojemnika), lecz w praktyce taka implementacja jest zazwyczaj znacznie wolniejsza.

Podstawowym problemem w heurystykach pakowania zorientowanych na pojemnik, które rozważają więcej niż jeden podzbiór dla każdego pojemnika, jest to, że takie heurystyki często wykorzystują znaczną część małych obiektów we wczesnych etapach pakowania, pozostawiając na koniec tylko obiekty o dużych wagach, trudnych do efektywnego zapakowania. W heurystyce MBS' z pracy [14] zjawisku temu zapobiega w pewnym stopniu wymuszenie zapakowania w każdym pakowanym pojemniku największego dostępnego obiektu. W pracy [1] zaproponowaliśmy znacznie lepszy mechanizm zapobiegający temu zjawisku. Zdefiniowaliśmy *zasadę wystarczającej średniej wagi* (ang. sufficient average weight, w skrócie SAW): podzbiór obiektów pakowanych do pojemnika ma wystarczająco dużą średnią wagę, jeśli średnia waga wybranych obiektów jest nie mniejsza niż średnia waga wszystkich aktualnie niezapakowanych obiektów. W oparciu o tę zasadę zaproponowane zostały zmodyfikowane heurystyki zorientowane na pojemnik, nazwane SAWB2F, SAWMBS oraz SAWMBS', które działają podobnie do B2F, MBS oraz MBS', lecz ograniczają wybór podzbioru dla każdego pojemnika tylko do podzbiorów maksymalnych (tzn. nierozszerzalnych) spełniających warunek wystarczającej średniej wagi obiektów. Jeśli żaden ze zbiorów maksymalnych nie osiąga wystarczającej średniej wagi, wybierany jest podzbiór maksymalny o największej średniej wadze obiektów. Dodanie warunku maksymalności było w tym wypadku konieczne, gdyż w niektórych przypadkach zasada wystarczającej średniej wagi mogłaby wymusić zapakowanie podzbioru niemaksymalnego.

Zasada wystarczającej średniej wagi bazuje na założeniu, że problem pakowania jest tym łatwiejszy, im mniejsza jest średnia waga obiektów. Jeśli do pewnego pojemnika zostanie zapakowanych stosunkowo dużo małych obiektów, to średnia waga obiektów pozostałych do zapakowania wzrasta, a więc wzrasta również trudność problemu pakowania. Wprowadzona zasada zapobiega takiej sytuacji, gdyż wymaga pakowania do każdego pojemnika podzbioru obiektów o średniej wadze nie mniejszej niż średnia waga wszystkich aktualnie dostępnych obiektów. W rezultacie problem pozostały po zapakowaniu danego pojemnika jest nie trudniejszy niż przed zapakowaniem pojemnika, gdyż średnia waga obiektów pozostałych po zapakowaniu pojemnika jest nie większa niż przed jego zapakowaniem.

W pracy [1] zaproponowaliśmy również wykorzystanie technik redukcji w ramach heurystyk zorientowanych na pojemnik. Przed uruchomieniem każdej heurystyki zastosowaliśmy metodę redukcji MTRP [32] o złożoności $O(n^2)$. Dodatkowo zaproponowaliśmy wykorzystanie po zapakowaniu każdego pojemnika trzech prostych metod redukcji o złożoności $O(n)$. Warto zauważyć, że wprowadzone redukcje nie zwiększają złożoności obliczeniowej heurystyk o złożoności $O(n^2)$ lub wyższej.

W pracy [1] zaproponowaliśmy również ulepszoną wersję metody poprawy rozwiązań Perturbation MBS' z pracy [14], nazwaną Perturbation SAWMBS. W nowej metodzie perturbacja jest wykonywana tylko wtedy, gdy sumaryczne miejsce we wszystkich pojemnikach jest nie mniejsze niż c (c = pojemność pojemnika), gdyż tylko wtedy istnieje możliwość poprawy rozwiązania. Jeśli sumaryczne wolne miejsce wynosi c, to wykorzystywana jest oryginalna metoda Perturbation MBS', która z taką sytuacją radzi sobie najlepiej. Jeśli sumaryczne wolne miejsce jest większe niż c, to wykorzystywana jest nowa metoda perturbacji. Wszystkie obiekty sortowane są losowo, przy czym obiekty o większych wagach, jak również obiekty znajdujące się w pojemnikach o stosunkowo dużym wolnym miejscu, mają większe szanse na znalezienie się wcześniej w uporządkowaniu. Następnie dla uzyskanego porządku obiektów metodą FFD wyznaczany jest jeden podzbiór obiektów, które przeniesione zostają do nowego pojemnika. Na koniec, podobnie jak w oryginalnej Perturbation MBS', z rozwiązania usuwane są wszystkie puste pojemniki.

Testy obliczeniowe w pracy [1] pokazały, że zaproponowana zasada wystarczającej średniej wagi znacząco poprawia średnią jakość rozwiązań MBS oraz MBS', jednocześnie zauważalnie skracając ich czas wykonywania. Podobną poprawę, choć w mniejszym stopniu, uzyskaliśmy poprzez zastosowanie technik redukcji. Pokazaliśmy również, że Perturbation SAWMBS znajduje rozwiązania nieco lepsze niż inne wcześniej zaproponowane złożone metody heurystyczne, a jej średni czas obliczeń jest znacznie krótszy niż wcześniej opublikowanych złożonych metod heurystycznych. Dodatkowo pokazaliśmy, że heurystyka SAWB2F z technikami redukcji uzyskuje najlepsze rozwiązania pośród wszystkich znanych heurystyk o złożoności nie wyższej niż $O(n^2)$.

Pakowanie dwuwymiarowe z cięciami gilotynowymi

Problem pakowania dwuwymiarowego (ang. two-dimensional bin packing) jest generalizacją problemu jednowymiarowego. Polega on na zapakowaniu zbioru prostokątnych obiektów o zadanych wymiarach w jak najmniejszej liczbie jednakowych prostokątnych pojemników. Problem ten może być równoważnie sformułowany jako problem rozkroju: zadany zbiór prostokątnych obiektów musi zostać wycięty z jak najmniejszej liczby jednakowych danych prostokątnych obiektów (pojemników). Wymaga się, by obiekty były zapakowane (wycięte) tak, by ich brzegi były równoległe do odpowiednich brzegów pojemników.

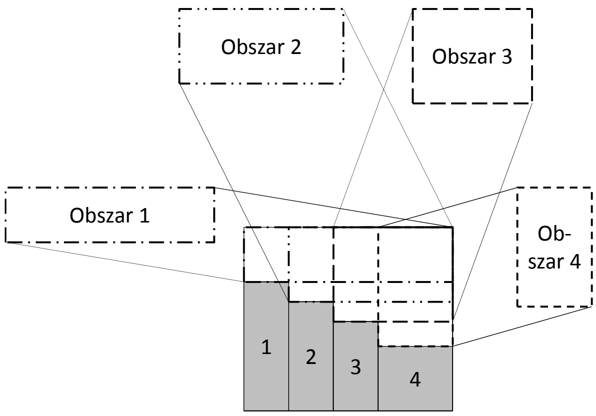
Cztery warianty problemu są typowo rozważane, w zależności od wprowadzenia lub zrelaksowania następujących dwóch ograniczeń [36]. Pierwsze z nich to ograniczenie orientacji przestrzennej obiektów względem pojemników: orientacja obiektów może być niezmienna lub może być dopuszczone obracanie obiektów o 90 stopni. Drugie to ograniczenie wymagające cięć gilotynowych. Jeśli to ograniczenie jest wprowadzone, obiekty muszą być ułożone tak, by możliwe było wycięcie ich z „pojemników" za pomocą tzw. cięć gilotynowych (od brzegu do brzegu).

W pracach [2] i [3] zaproponowane zostały metody dla dwóch wariantów problemu: z cięciami gilotynowymi bez rotacji obiektów oraz z cięciami gilotynowymi i z rotacją obiektów. Praca [2] została wykonana we współpracy z dr Charalambousem w ramach grantu Cypryjskiej Fundacji Promocji Nauki, którego celem było zaproponowanie efektywnych metod cięcia gilotynowego płyt szklanych. Praca [3] wykonana została przeze mnie indywidualnie po zakończeniu grantu.

Typowe szybkie metody rozwiązywania problemu pakowania dwuwymiarowego konstruują rozwiązania dwuetapowo, w każdym z etapów koncentrując się na jednym wymiarze: najpierw obiekty pakowane są w tzw. półki (prostokąty o szerokościach równych szerokości pojemnika i różnych wysokościach), a następnie półki pakowane są do pojemników. Do takich metod należą finite-best-strip i finite-first-fit [35] oraz floor-ceiling i knapsack-problem-based [36]. Metody zaproponowane w pracach [2], [3] odbiegają od tego schematu, gdyż konstruują rozwiązania pakując obiekty bezpośrednio do pojemnika.

W pracy [2] zaproponowaliśmy metodę konstrukcyjną, która buduje rozwiązanie dla pojedynczego pojemnika przyrostowo, w każdej fazie wstawiając *zestaw obiektów* ułożonych obok siebie w jednym z tzw. *dostępnych prostokątnych obszarów*. Dla danego dostępnego prostokątnego obszaru, generowanych jest szereg zestawów obiektów i wybierany jest ten, który najlepiej wypełnia obszar. Każdy z zestawów generowany jest metodą first-fit z innym uszeregowaniem obiektów. Rozważane uszeregowania zależą częściowo od wysokości obiektów (im wyższy obiekt, tym wcześniej na liście), a częściowo od powierzchni obiektów (im większa powierzchnia obiektu, tym wcześniej obiekt jest na liście). W zależności od wag przypisanych powyższym kryteriom uzyskuje się różne uszeregowania i różne zestawy.

Metoda konstrukcyjna zaproponowana w pracy [2] wyznacza najpierw najlepszy zestaw obiektów dla obszaru całego pojemnika. Po zapakowaniu tego zestawu, wyznaczane są największe możliwe dostępne prostokątne obszary, które mogą się częściowo pokrywać. Na rysunku obok przedstawiona jest taka sytuacja z czterema obszarami. Dla każdego z nich wyznaczany jest najlepszy zestaw obiektów. Następnie najlepszy ze wszystkich zestawów jest pakowany do odpowiedniego prostokątnego obszaru, po czym aktualizowana jest lista dostępnych prostokątnych obszarów. Proces ten jest powtarzany aż do chwili, gdy żaden z dostępnych obiektów nie mieści się w żadnym z dostępnych prostokątnych obszarów.



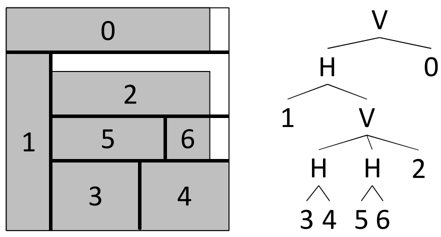
Przy wyborze najlepszego zestawu obiektów do zapakowania wykorzystana została miara najlepszego wypełnienia. Dodatkowo, wykorzystana została również zasada wystarczającej średniej powierzchni, będąca adaptacją zasady wystarczającej średniej wagi z pracy [1]. Zgodnie z tą zasadą, wybór najlepszego zestawu obiektów ogranicza się do takich zestawów, które pozwolą utrzymać średnią powierzchnię zapakowanych obiektów na poziomie nie niższym niż średnia powierzchnia wszystkich obiektów aktualnie dostępnych. Jeśli żaden z zestawów nie spełnia tego warunku, wybierany jest ten, który jest najbliższy jego spełnienia.

Poza opisaną powyżej heurystyką konstrukcyjną, w pracy [2] zaproponowaliśmy dwie metody postoptymalizacji. Jedna z nich polega na wielokrotnym uruchamianiu heurystyki konstrukcyjnej z nieco zaburzonym (zaostrzonym lub zrelaksowanym) warunkiem wystarczającej średniej powierzchni obiektów. Z tak wygenerowanych rozwiązań wybierane jest to, które wykorzystuje najmniej pojemników. Druga metoda postoptymalizacji wykorzystuje informację z pierwszej metody. Wykrywa ona obiekty, które są często pakowane w ostatnich pojemnikach. Dla kilku takich obiektów metoda konstrukcyjna jest uruchamiana ponownie, przy czym w każdym uruchomieniu jeden z tych obiektów jest pakowany w ramach pierwszego zestawu obiektów w pierwszym pojemniku.

Testy obliczeniowe przedstawione w pracy [2] pokazały, że wykorzystanie zasady wystarczającej średniej powierzchni poprawia nieco średnią jakość rozwiązań metody konstrukcyjnej bez wpływu na czas obliczeń. Pokazaliśmy również, że zaproponowana metoda konstrukcji znajduje rozwiązania o bardzo dobrej jakości w porównaniu do innych prostych heurystyk, a gdy wykorzystane zostaną dodatkowo dwie zaproponowane metody postoptymalizacji, uzyskiwane rozwiązania są lepsze niż rozwiązania innych znanych złożonych heurystyk.

Wadą zaprezentowanej heurystyki konstrukcyjnej jest to, że ma ona złożoność obliczeniową $O(n^3 \log n)$, podczas gdy typowe szybkie metody konstrukcji rozwiązań mają złożoność $O(n^2)$. Dodatkowo, wadą zarówno zaproponowanej metody jak i innych wcześniej opublikowanych metod jest fakt, że w trakcie konstrukcji rozwiązania położenie już zapakowanych obiektów nie może być zmienione, co w pewnych przypadkach znacznie utrudnia wstawienie dodatkowych obiektów i uzyskanie dobrego rozwiązania. W pracy [3] starałem się rozwiązać powyższe problemy.

Heurystyki konstrukcyjne, które zaproponowałem w pracy [3], bazują na reprezentacji rozwiązań za pomocą drzew. Zapakowanie każdego pojemnika reprezentowane jest za pomocą pojedynczego drzewa, które składa się z węzłów reprezentujących cięcia gilotynowe oraz z węzłów reprezentujących pakowane obiekty. Przykładowe zapakowanie pojemnika oraz odpowiadające mu drzewo są przedstawione na rysunku obok. Podobna reprezentacja rozwiązań była wykorzystywana wcześniej, np. w algorytmie genetycznym Kroegera [37], lecz nigdy w kontekście szybkich metod konstrukcji rozwiązań.



Heurystyki konstrukcyjne zaproponowane w pracy [3] sortują obiekty według nierosnących powierzchni (w przypadku równych powierzchni według nierosnących długości krótszych boków), a następnie pakują po jednym obiekcie na raz (a więc są zorientowane na obiekt), wybierając dla danego obiektu pojemnik oraz położenie w pojemniku. Za każdym razem, gdy rozważany obiekt nie mieści się w żadnym dostępnym pojemniku, dodawany jest nowy pojemnik, do którego pakowany jest rozważany obiekt.

Zaproponowane heurystyki bazują na dwóch ważnych innowacyjnych elementach. Jednym z nich jest nowa metoda o złożoności O(n) służąca do enumeracji podzbioru dopuszczalnych wstawień rozważanego obiektu w drzewie reprezentującym zapakowanie pojemnika. Drugim jest nowe kryterium dopasowania pozwalające na wybranie najlepszego z wielu możliwych wstawień, bazujące na stopniu dopasowania wielkości wstawianego obiektu do wielkości zajmowanego obszaru oraz do wielkości obiektu sąsiedniego.

Na bazie powyższych elementów, w pracy [3] zaproponowałem trzy metody konstrukcji i jedną metodę poprawy rozwiązań. Pierwsza metoda, nazwana first-fit insertion heuristic (FFIH), pakuje każdy obiekt w najlepszy możliwy sposób w pierwszym pojemniku, w którym ten obiekt się mieści. Druga, best-fit insertion heuristic (BFIH), pakuje obiekt w tym pojemniku, w którym obiekt ten uzyskuje najlepsze dopasowanie. FFIH oraz BFIH są adaptacjami heurystyk pakowania jednowymiarowego FFD oraz BFD do problemu pakowania dwuwymiarowego.

Trzecia metoda, critical-fit insertion heuristic (CFIH), rozważa w każdej fazie wszystkie niezapakowane obiekty, których wielkość jest niezdominowana przez żaden inny niezapakowany obiekt. Obiekt uznaje się za zdominowany, jeśli mieści się on w prostokącie zdefiniowanym przez inny dostępny obiekt. Dla każdego z obiektów niezdominowanych, wyznaczane jest nie tylko najlepsze aktualnie możliwe zapakowanie, ale również liczba pojemników, w których jego zapakowanie jest możliwe. CFIH wybiera do zapakowania obiekt krytyczny, czyli taki obiekt, dla którego liczba pojemników jest najmniejsza. W przypadku równych liczb pojemników wybierany jest obiekt uzyskujący lepsze dopasowanie.

Metoda wyboru obiektu w CFIH bazuje na założeniu, że im mniejsza liczba pojemników do których może być zapakowany dany obiekt, tym ważniejsze jest jak najszybsze zapakowanie obiektu. Na przykład, gdy istnieje obiekt, który nie może być już zapakowany w żadnym dostępnym pojemniku, to natychmiast dodawany jest nowy pojemnik i obiekt ten jest w nim pakowany. Dzięki temu zwiększają się możliwości wyboru przy pakowaniu pozostałych obiektów. W innym wypadku, gdy istnieje niezapakowany obiekt, który mieści się tylko w jednym pojemniku, jest on pakowany przed obiektami, które mieszczą się w dwóch lub więcej pojemnikach. Pozwala to czasem uniknąć dodania niekoniecznego pojemnika.

Czwarta i ostatnia metoda zaproponowana w pracy [3] to metoda poprawy poprzez tzw. justyfikację, będąca adaptacją podobnej metody zastosowanej do problemu harmonogramowania projektu z ograniczonymi zasobami (ang. resource-constraint project scheduling) [38]. Metoda ta próbuje poprawić dane rozwiązanie początkowe poprzez sukcesywne rozpakowywanie pojemników począwszy od ostatniego aż do pierwszego i pakowanie rozpakowanych obiektów w nowym rozwiązaniu opisaną powyżej metodą FFIH. Tym razem jednak metoda FFIH pakuje obiekty w kolejności ich rozpakowywania z rozwiązania początkowego, przy czym dla danego pojemnika obiekty rozpakowywane są w kolejności nierosnących powierzchni (w przypadku równych powierzchni w kolejności nierosnących długości krótszych boków). W dowolnym momencie, oba częściowe rozwiązania mogą być połączone w jedno kompletne rozwiązanie. W ten sposób możliwe jest uzyskanie rozwiązań wykorzystujących mniej pojemników niż rozwiązanie początkowe. Opisana metoda justyfikacji może być powtórzona wielokrotnie, jeśli końcowe rozwiązanie przyjęte zostanie jako nowe rozwiązanie początkowe.

Warto zauważyć, że metody FFIH, BFIH oraz justyfikacja mają bardzo dobrą złożoność obliczeniową $O(n^2)$, a tylko metoda CFIH wymaga obliczeń o złożoności $O(n^3)$.

Testy obliczeniowe przedstawione w pracy [3] pokazały, że zaproponowane heurystyki konstrukcji są wysoce efektywne i bardzo szybkie. FFIH oraz BFIH znajdują rozwiązania znacznie lepsze niż poprzednio opublikowane heurystyki o podobnej złożoności obliczeniowej. Jedyną heurystyką w tej grupie zdolną konkurować z FFIH oraz BFIH jest heurystyka floor-ceiling zmodyfikowana tak, by jej złożoność wynosiła $O(n^2)$ (w oryginale złożoność tej metody jest wykładnicza). Gdy po uruchomieniu FFIH lub BFIH wykonywana jest dodatkowo zaproponowana metoda poprawy za pomocą justyfikacji, uzyskane rozwiązania są znacznie lepsze niż rozwiązania wszystkich poprzednio opublikowanych heurystyk o podobnej złożoności obliczeniowej, nawet gdy do tych heurystyk również dodana jest metoda justyfikacji.

Mimo iż teoretyczna złożoność obliczeniowa CFIH jest znaczenie wyższa niż złożoność FFIH oraz BFIH, czas obliczeń CFIH jest średnio tylko 2-3 krotnie dłuższy niż czasy obliczeń FFIH oraz BFIH. Jednocześnie rozwiązania CFIH są znacznie lepsze. Ponadto CFIH w połączeniu z justyfikacją uzyskuje rozwiązania o jakości lepszej lub porównywalnej z rozwiązaniami wcześniej opublikowanych metaheurystyk, których czas obliczeń jest zazwyczaj znacznie dłuższy niż czas obliczeń CFIH.

W przyszłości na bazie heurystyk zaproponowanych w pracy [3] planuję zaprojektowanie metaheurystyk (np. VNS) do rozwiązywania tego samego problemu. Planuję również adaptację powyższych heurystyk do problemu pakowania dwuwymiarowego w pojemnikach o różnych wielkościach i różnych kosztach.

Problemy dystrybucyjne z ograniczeniami pojemnościowymi

Innymi problemami alokacji, nad którymi pracowałem, są problemy dystrybucyjne. W problemach tych wyróżnia się zwykle zbiór klientów o określonych lokalizacjach i określonych zapotrzebowaniach. Każdy z klientów musi zostać przypisany do pewnego rodzaju obiektu obsługi, który zwykle ma ograniczoną pojemność. W tym sensie, problemy te są podobne do problemów pakowania.

Wyniki moich prac nad dwoma problemami dystrybucyjnymi zostały opisane w artykułach [4] i [5]. Praca [4] wykonana była we współpracy z prof. Hindim i prof. Osmanem, a praca [5] wykonana była we współpracy z prof. Hindim. W obu pracach zaproponowaliśmy algorytmy wykorzystujące metaheurystykę VNS [33], wykorzystaną we wcześniejszej mojej pracy [14] do rozwiązywania jednowymiarowego problemu pakowania.

W pracy [4] zajmowaliśmy się rozwiązywaniem problemu planowania tras dostaw, w którym trasy pojazdów są otwarte (ang. open vehicle routing problem, OVRP). Problem ten jest podobny do problemu planowania dostaw (ang. vehicle routing problem, VRP). Jedyną różnicą jest fakt, że w OVRP trasa każdego pojazdu kończy się w punkcie, w którym znajduje się ostatni obsługiwany klient. Problem ten ma zastosowanie w sytuacjach, w których pojazdy są wynajmowane od firmy trzeciej. W takich sytuacjach opłacane są tylko koszty przejazdu z towarem, a pojazdy nie muszą wracać do centrum obsługi.

Formalnie, w otwartym problemie planowania dostaw dany jest zbiór klientów o określonych lokalizacjach i zapotrzebowaniach oraz centrum dystrybucyjne o określonej lokalizacji. Celem rozwiązania problemu jest znalezienie najmniejszej liczby pojazdów, które mogą obsłużyć wszystkich klientów, a następnie wyznaczenie takich tras dostaw, aby sumaryczny czas przejazdu wszystkich pojazdów był jak najkrótszy. Każdy z klientów powinien zostać odwiedzony tylko przez jeden pojazd, który powinien pokryć całe zapotrzebowanie klienta. Każdy z pojazdów ma ograniczoną pojemność oraz opcjonalnie ograniczony całkowity czas przejazdu.

W artykule [4] do rozwiązywania otwartego problemu planowania dostaw zaproponowaliśmy metodę VNS. W pierwszym kroku algorytmu wyznaczane są dwa oszacowania dolne na liczbę pojazdów, jedno bazujące na zapotrzebowaniach klientów i pojemności pojazdu, a drugie bazujące na minimalnym drzewie rozpinającym oraz maksymalnym czasie trwania trasy pojazdu. Większa z wartości wykorzystywana jest jako wartość startowa liczby pojazdów. Następnie uruchamiana jest metaheurystyka VNS, w której wyszukiwane jest najlepsze możliwe rozwiązanie z zadaną liczbą pojazdów. W trakcie VNS przeszukiwana jest przestrzeń zarówno rozwiązań dopuszczalnych jaki i rozwiązań niespełniających ograniczeń, przy czym VNS minimalizuje w pierwszej kolejności stopień niespełnienia ograniczeń. Jeśli VNS nie jest w stanie znaleźć rozwiązania dopuszczalnego, to liczba pojazdów jest zwiększana o jeden i metoda VNS jest uruchamiana ponownie.

W ramach VNS wykonywane są cyklicznie optymalizacja lokalna oraz losowe zaburzenie rozwiązania w celu ucieczki z minimum lokalnego. Optymalizacja lokalna wykonuje najpierw optymalizację czasu trwania pojedynczych tras, a następnie optymalizację bazującą na przeniesieniach i wymianach klientów pomiędzy parami tras. Losowe zaburzenie polega na przeniesieniu losowo wybranego segmentu trasy losowo wybranego pojazdu do trasy innego losowo wybranego pojazdu.

Rozwiązanie początkowe wyznaczane jest za pomocą prostej heurystyki, która wstawia klientów do tras pojazdów tak, by jak najmniej zwiększać całkowity czas przejazdów pojazdów. Metoda VNS jest uruchamiana wielokrotnie dla każdej zadanej liczby pojazdów. Za pierwszym razem w rozwiązaniu początkowym wykorzystywane jest uszeregowanie według nierosnących zapotrzebowań klientów. W pozostałych uruchomieniach wykorzystywane jest losowo zaburzone uszeregowanie według nierosnących zapotrzebowań klientów.

Testy obliczeniowe pokazały, że zaproponowana metoda VNS znajduje rozwiązania lepsze niż wcześniej zaproponowane metody bądź rozwiązania porównywalne z wcześniej zaproponowanymi metodami, które jednak wymagają większego czasu obliczeń niż VNS.

W pracy [5] zajmowaliśmy się problemem wyznaczania lokalizacji *p* punktów obsługi o ograniczonych pojemościach (ang. capacitated *p*-median problem). W problemie tym dany jest zbiór klientów o określonych lokalizacjach i zapotrzebowaniach. Należy wyznaczyć lokalizacje *p* punktów obsługi (median) i tak przypisać klientów do tych punktów, aby nie przekroczyć pojemności żadnego punktu obsługi oraz aby suma odległości między klientami i punktami obsługi była jak najmniejsza.

W artykule [5] zaproponowaliśmy algorytm rozwiązywania tego problemu bazujący na dekompozycji na problem nadrzędny lokalizacji punktów obsługi i problem podrzędny przypisania klientów do punktów obsługi. Do rozwiązywania problemu nadrzędnego wykorzystaliśmy metodę VNS. Optymalizacja lokalna w VNS bazuje na testowaniu zmian lokalizacji pojedynczych punktów obsługi. Losowe zaburzenie w VNS wyznaczane jest poprzez losową zmianę lokalizacji losowo wybranych punktów obsługi.

W problemie podrzędnym klienci muszą być przypisani do punktów obsługi tak, aby sumaryczna odległość klientów od punktów obsługi była jak najmniejsza oraz aby nie przekroczona została pojemność żadnego z punktów obsługi. Ten problem to uogólniony problem alokacji (ang. generalized assignment problem, GAP) i chociaż jest on łatwiejszy do rozwiązania niż problem oryginalny, to jest on wciąż problemem NP-trudnym. Do jego rozwiązywania wykorzystaliśmy model programowania całkowitoliczbowego, rozwiązywany przez pakiet CPLEX. Aby uniknąć częstego jego rozwiązywania, wykorzystaliśmy dwa znacznie szybsze obliczeniowo oszacowania dolne. Pierwsze oszacowanie, LB1, wykorzystuje relaksację ograniczenia na pojemność punktu obsługi. W tym wypadku rozwiązanie GAP sprowadza się do przypisania każdego klienta do najbliższego punktu obsługi. Drugie oszacowanie, LB2, wykorzystuje relaksację liniową GAP.

Powyższe oszacowania dolne wykorzystywane są w ramach optymalizacji lokalnej w VNS. Gdy testowana jest zmiana lokalizacji jednego punktu obsługi, to zanim uruchomione zostanie rozwiązanie GAP za pomocą pakietu CPLEX, najpierw wyznaczane są wartości LB1, a następnie LB2. Jeśli którakolwiek z tych wartości wskazuje na to, że koszt po zmianie lokalizacji punktu obsługi nie może być lepszy niż przed zmianą, to rozwiązywanie GAP za pomocą pakietu CPLEX nie jest wywoływane.

Eksperymenty obliczeniowe przedstawione w pracy [5] pokazały, że zaproponowana metoda jest bardzo skuteczną metodą rozwiązywania problemu lokalizacji punktów obsługi z ograniczonymi pojemnościami. Pokazaliśmy również, że wykorzystanie oszacowań dolnych pozwala uniknąć rozwiązywania uogólnionego problemu alokacji za pomocą pakietu CPLEX w ponad 99% przypadków, dzięki czemu opracowana metoda jest bardzo szybka.

6. Dodatkowa bibliografia

[6] **Fleszar K**, Hindi KS. Fast, effective heuristics for the 0-1 multi-dimensional knapsack problem. Computers and Operations Research 36(5), 1602-1607, May 2009.

[7] **Fleszar K**, Charalambous C, Hindi KS. A variable neighborhood descent heuristic for the problem of makespan minimisation on unrelated parallel machines with setup times. Journal of Intelligent Manufacturing 23(5), pp. 1949-1958, Oct 2012

[8] Charalambous C, **Fleszar K**. Variable neighborhood descent for the unrelated parallel machine scheduling problem. International Journal on Artificial Intelligence Tools 21(4), Aug 2012

[9] Hindi KS, **Fleszar K**. A constraint propagation heuristic for the single-hoist, multiple-products scheduling problem. Computers and Industrial Engineering 47(1), 91-101, Aug 2004

[10] **Fleszar K**, Hindi KS. Solving the resource-constrained project scheduling problem by a variable neighbourhood search. European Journal of Operational Research 155(2), 402-413, Jun 2004

[11] Hindi KS, **Fleszar K**, Charalambous C. An effective heuristic for the CLSP with set-up times. Journal of the Operational Research Society 54(5), 490-498, May 2003

[12] **Fleszar K**, Hindi KS. An enumerative heuristic and reduction methods for the assembly line balancing problem. European Journal of Operational Research 145(3), 606-620, Mar 2003

[13] Hindi KS, Yang H, **Fleszar K**. An evolutionary algorithm for resource-constrained project scheduling. IEEE Transactions on Evolutionary Computation 6(5), 512-518, Nov 2002

[14] **Fleszar K**, Hindi KS. New heuristics for one-dimensional bin-packing. Computers and Operations Research 29(7), 821-839, Jun 2002

[15] **Fleszar K**, Charalambous C. Hybridizing local search with MIP for unrelated parallel machine scheduling problems with makespan minimization. 5th Multidisciplinary International Scheduling Conference (MISTA), Phoenix, Arizona, 9-12 Aug 2011.

[16] Charalambous C, **Fleszar K**, Hindi KS. A hybrid searching method for the unrelated parallel machine scheduling problem with setup times. 6th IFIP Conference on Artificial Intelligence Applications and Innovations, Cyprus, 5-7 Oct 2010.

[17] **Fleszar K**, Charalambous C, Hindi KS. SAWMBS: A Sufficient-Average-Weight-Minimum-Bin-Slack Heuristic for One-Dimensional Bin-Packing. 23rd European Conference on Operational Research (EURO), Bonn, Germany, 5-8 Jul 2009.

[18] Charalambous C, Hindi KS, **Fleszar K**. A constructive bin-oriented algorithm for the two-dimensional bin packing problem with guillotine cuts. 23rd European Conference on Operational Research (EURO), Bonn, Germany, 5-8 Jul 2009.

[19] Osman IH, Belouadah H, **Fleszar K**, Saffar M. New dispatching rule for the total weighted tardiness single machine scheduling problem with availability constraints. 4th Multidisciplinary International Scheduling Conference (MISTA), Dublin, Ireland, 10-12 Aug 2009.

[20] Osman IH, Belouadah H, **Fleszar K**. Evaluation of Heuristics for the Weighted Tardiness Single Machine Scheduling Problem with Periodic Maintenance. Conference for International Federation of Operational Research Societies (IFORS), South Africa, 13-19 Jul 2008.

[21] **Fleszar K**, Hindi KS. An effective VNS for the capacitated p-median problem. Learning and Intelligent OptimisatioN (LION2) Conference, Trento, Italy, 2007.

[22] **Fleszar K**, Charalambous C, Hindi KS, Pericleous S. Minimum-bin-waste heuristic for the two-dimensional bin-packing problem with guillotine cuts and item rotation. 22nd European Conference on Operational Research (EURO), Prague, Czech Republic, 2007.

[23] **Fleszar K**, Ogryczak W. On Generalized OWA Approach to Support Location and Routing Decisions. In Jaszkiewicz A, Kaczmarek M, Zak J, Kubiak M (eds.). Advanced OR and AI Methods in Transportation. Proceedings of 10th EWGT Meeting and 16th Mini-EURO Conference, Poznań, Poland, 746-751, 2005.

[24] **Fleszar K**. Toczyłowski E, Algorytmy przybliżonego rozwiązywania problemu aukcji kombinatorycznej. Materiały konferencyjne: XV Krajowa Konferencja Automatyki, tom III, 335-340, Warszawa, 2005.

[25] Hindi KS, **Fleszar K**. A Constraint Propagation Heuristic for Solving the Single-Hoist, Multiple-Products Scheduling Problem. EURO/INFORMS Joint International Meeting, Istanbul, Turkey, 6-10, 2003.

[26] **Fleszar K**, Kaleta M, Pieńkosz K, Rogulski M, Toczyłowski E. Uwzględnianie energii i kosztów rozruchów jednostek wytwórczych na godzinowym Rynku Bilansującym energii elektrycznej w Polsce. IX Konferencja Naukowo-Techniczna „Rynek Energii Elektrycznej", Kazimierz, 2002.

[27] **Fleszar K**, Kaleta M, Pieńkosz K, Rogulski M, Toczyłowski E. Analiza systemu rozliczeń na Rynku Bilansującym. IX Konferencja Naukowo-Techniczna „Rynek Energii Elektrycznej", Kazimierz, 2002.

[28] **Fleszar K**, Kaleta M, Pieńkosz K, Rogulski M, Toczyłowski E. Analiza struktury podmiotowej i obiektowej na Rynku Bilansującym. IX Konferencja Naukowo-Techniczna „Rynek Energii Elektrycznej", Kazimierz, 2002.

[29] **Fleszar K**. Nowe heurystyczne metody rozwiązywania jednowymiarowego problemu bin-packing. Zeszyty Naukowe Politechniki Śląskiej, Seria: Automatyka, z. 136, Nr kol. 1556, 45-54, 2002.

[30] Walczak A, Toczyłowski E, Granat J, **Fleszar K**. Wspomaganie decyzji dobowego planowania pracy jednostek wytwórczych na rynku energii elektrycznej. Konferencja Naukowo-Techniczna „Optymalizacja w elektroenergetyce", Jachranka, 33-42, 1999.

[31] Gupta JND, Ho JC, A new heuristic algorithm for the one-dimensional binpacking problem. Production Planning and Control 10 (6), pp. 598-603, 1999.

[32] Martello S, Toth P. Knapsack problems. New York: Wiley, 1990.

[33] Mladenović N, Hansen P, Variable neighbourhood search, Computers and Operations Research 24 (11), pp. 1097-1100, 1997.

[34] Friesen, D., Langston, M., Analysis of a compound bin packing algorithm. SIAM Journal on Discrete Mathematics 4, 61, 1991.

[35] Berkey J,Wang P. Two-dimensional finite bin-packing algorithms. Journal of the Operational Research Society 38(5), pp. 423-429, 1987.

[36] Lodi A, Martello S, Vigo D. Heuristics and metaheuristic approaches for a class of two-dimensional bin packing problems. INFORMS Journal on Computing 11(4), pp. 345-357, 1999.

[37] Kroeger B. Guillotineable bin packing: a genetic approach. European Journal of Operational Research 84(3), pp. 645-61, 1995.

[38] Valls V, Ballestin F, Quintanilla S. Justification and RCPSP: a technique that pays. European Journal of Operational Research 165(2), pp. 375-386, 2005.

Krzysztof Fleszar