PROBLEM KOMIWOJAŻERA – STUDIUM PRZYPADKU

TRAVELLING SALESMAN PROBLEM – CASE STUDY

Jarosław ZIÓŁKOWSKI
jaroslaw.ziolkowski@wat.edu.pl

Angelika MIZIOŁEK
angelika.miziolek@gmail.com

Wojskowa Akademia Techniczna
Wydział Mechaniczny

Dariusz ĆWIK
dariusz.cwik@wat.edu.pl

Wojskowa Akademia Techniczna
Wydział Logistyki

Streszczenie: W artykule przedstawiono problem komiwojażera na przykładzie liczbowym. Celem jest znalezienie trasy łączącej wszystkie miasta, która całościowo jest najkrótsza, najszybsza lub najtańsza i ponadto zaczyna się i kończy się w określonym punkcie. Jest to typowe zagadnienie optymalizacyjne, w którym zadane jest n miast, które komiwojażer musi odwiedzić. Jego rozwiązanie polega na znalezieniu minimalnego cyklu Hamiltona *w pełnym grafie ważonym.*

Abstract: The paper presents the travelling salesman problem (TSP) on a numerical example. The aim of the paper is to find the shortest, the fastest and the cheapest route, which links all the cities and additionally starts and ends at a particular point. This is the typical optimization problem with n number of cities that the travelling salesman has to visit. The solution of the problem is to find the minimum Hamiltonian cycle in a complete weighted graph.

Słowa kluczowe: problem komiwojażera, studium przypadku, optymalizacja.

Key words: travelling salesman problem, case study, optimization.

WSTĘP

Transport jest jedną z podstawowych składowych usług logistycznych. Jego kluczowe zagadnienia stanowią często obszar działalności logistyki informatycznej ze względu na nałożone „z góry” ograniczenia przestrzenne, logiczne i czasowe związane z ruchem i obsługą obiektów w systemie (Ochelska – Mierzejewska, 2016, s. 350). Według danych statystycznych koszty transportu najczęściej mieszczą się w przedziale 30-60[%] (lub więcej) kosztów ogólnologistycznych (Dziamski, 2004, s. 48). Ich względny udział zależy zarówno od branży w jakiej funkcjonuje dany podmiot oraz od rodzaju i profilu działalności danej firmy (Witkowski i Tanona, 2013, s. 411).

Zagadnienie komiwojażera zostało sformułowane w 1759 r. przez Eulera, który zajmował się rozwiązaniem zadania ruchu konika po szachownicy. Optymalne i prawidłowe rozwiązanie polegało, aby konik odwiedził każde pole na szachownicy dokładnie raz w ciągu całego ruchu. W 1932 roku termin „komiwojażer" pojawił się w niemieckiej książce, natomiast problem komiwojażera w obecnym znaczeniu został wprowadzony w 1948 roku przez RAND Corporation (Michalewicz, 2003, s. 245). Problem komiwojażera (j. ang. *Traveling Salesman Problem,* TSP) jest typowym i najbardziej popularnym utrudnieniem pojawiającym się w działalności transportowej. Ma ogromne znaczenie w logistyce, ponieważ dotyczy planowania optymalnych tras przejazdów (Rutczyńska-Wdowiak i Jabłoński, 2016, s. 1523). Zadaniem przewoźnika jest dostarczenie towaru do określonej liczby punktów odbioru i powrót do bazy. Każdy z punktów musi zostać odwiedzony dokładnie raz, ale można to zrobić w dowolnej kolejności (Ignasiak, 2001, s. 119). Dodatkowo znane są odległości pomiędzy węzłami i należy zaplanować trasę w taki sposób, aby sumaryczna droga była najkrótsza. Poprzez odległości można rozumieć długość drogi wyrażoną w kilometrach, czas trwania podróży między danymi punktami lub koszt podróży.

Celem problemu komiwojażera jest znalezienie trasy, która całościowo jest najkrótsza, najszybsza lub najtańsza. Dodatkowo zakłada się, że odległość między dwoma dowolnymi punktami jest nie większa niż długość jakiekolwiek drogi łączącej je a wiodącej przez inne punkty (Ignasiak, 2001, s. 121). Płaszczyznę rozwiązań w zadaniu komiwojażera stanowi zbiór permutacji *p* punktów, a rozwiązaniem jest każda ich permutacja. Optymalne rozwiązanie to takie ustawienie miast, dla którego koszt trasy jest najmniejszy. Z tego wynika, że rozmiar przestrzeni rozwiązań wynosi *p*! (http://www.staff.amu.edu.pl).

Aby rozwiązać problem komiwojażera, należy zbudować graf ważony, którego wierzchołkami są miejsca odbioru, a każda ich para połączona jest krawędzią. Każdej z krawędzi nadana jest waga równa odległości, czasu lub kosztowi przejazdu pomiędzy nimi. W ten sposób utworzony zostaje graf pełny, który ma tyle wierzchołków, ile miejsc musi odwiedzić komiwojażer. Odwiedzenie wszystkich węzłów odpowiada cyklowi, który przechodzi przez każdy wierzchołek danego grafu dokładnie raz. Cykl ten nazywany jest cyklem Hamiltona, w danym przypadku, o minimalnej sumie wag krawędzi. Rozwiązanie danego problemu istnieje zawsze, ponieważ dowolny graf pełny posiada co najmniej jeden cykl Hamiltona. Ze względu na skończoną liczbę wierzchołków występuje taki graf w zbiorze cykli Hamiltona, który posiada minimalną sumę wag krawędzi.

Istnieje wiele algorytmów, które rozwiązują problem komiwojażera. Jednak wszystkie mają podstawową wadę. Wymagają rozważania bardzo dużej liczby przypadków i czas ich działania może być bardzo długi. Niewielki przyrost liczby wierzchołków powoduje znaczny wzrost ilości przypadków do rozważenia oraz czasu działania algorytmu. Jeden z możliwych algorytmów polega na obliczeniu całkowitej długości wszystkich istniejących w danym grafie cykli Hamiltona. Niestety staje się to dosyć skomplikowane dla liczby miast większej od pięciu.

Stosując jeden z algorytmów przybliżonych można zmniejszyć czas rozwiązywania problemu komiwojażera, który nie wymaga rozważania dużej liczby przypadków. Niestety nie zawsze te algorytmy znajdują optymalne rozwiązanie. Stworzona przez nie trasa może być „dłuższa" od najkrótszej. Stosowanie algorytmów przybliżonych zrodziło się z konieczności wyboru pomiędzy „jakością" wyszukanego rozwiązania a szybkością znajdowania. Zakłada się, że wynik działania tego algorytmu nie może być gorszy od optymalnego o więcej niż pewna wartość ustalona z góry (http://www.mini.pw.edu.pl). Wśród najczęściej stosowanych algorytmów przybliżonych można wyróżnić między innymi:

- algorytm najbliższego sąsiada;
- algorytm 2-opt;
- algorytm 3-opt;
- algorytm Lina-Kernighana;
- algorytm Christofidesa;
- algorytm symulowanego wyżarzania;
- algorytm zachłanny (Rutczyńska-Wdowiak i Jabłoński, 2016, s. 1524).

Algorytm najbliższego sąsiada (j. ang. *nearest neighbour algorithm*) wykorzystuje strategię zachłanną. Pierwszym punktem w jego działaniu jest wybranie dowolnego wierzchołka i ustawienie go jako aktualny. Następnie przechodzi się do wierzchołka, który leży najbliżej i kreśli się między nimi krawędź. Teraz ten drugi wierzchołek jest aktualnym i dla niego szukamy wierzchołka leżącego najbliżej, ale takiego, który nie został jeszcze odwiedzony. W momencie, gdy wszystkie punkty zostaną odwiedzone wraca się do punktu wyjścia rysując krawędź od ostatniego do pierwszego wierzchołka. Opisywany algorytm nie daje gwarancji na znalezienie najlepszego rozwiązania w problemie komiwojażera. Znalezione rozwiązania mogą być nawet o około 25% gorsze od optymalnych. Rozwiązania otrzymane
w wyniku zastosowania tego algorytmu mogą się od siebie różnić, w zależności od wyboru węzła wyjściowego (Johnson i McGeoch, 1997, s. 7).

Algorytm 2-optymalny jest algorytmem lokalnej optymalizacji. Uważany jest za szczególny przypadek algorytmu *k*-optymalnego. Jego działanie polega na usunięciu dwóch krawędzi z początkowego cyklu i zastąpienie ich innymi krawędziami w taki sposób, aby utworzyły inny cykl. Dla każdej pary krawędzi czynność jest powtarzana, nie licząc krawędzi sąsiadujących ze sobą, ponieważ ich zmiana nie wniosłaby żadnego postępu. Za każdym razem modyfikowany zostaje cykl początkowy. Gdy wszystkie pary krawędzi zostaną sprawdzone, należy wybrać taką modyfikację, która najbardziej skróciła trasę bądź powrócić do cyklu początkowego (http://algorytmy.ency.pl).

Proces algorytmu 3-optymalnego jest zbliżony do działania poprzedniego. 3-optymalna analiza polega na usunięciu trzech krawędzi z początkowego cyklu i zastąpienie ich innym połączeniem punktów, tak aby utworzyły nowy cykl. Dla wytypowanych krawędzi wybierane jest połączenie optymalne. Następnie trzy pierwsze połączenia wymieniane są na inne i proces jest powtarzany. Za każdym razem przekształcany jest cykl początkowy. Ze wszystkich optymalnych połączeń typowane jest połączenie właściwe i najlepsze (Lin, 44, s. 2245-2269).

Algorytm Lina-Kerninghana posiada charakter heurystyczny. Może pracować na grafach o ujemnych i dodatnich wagach krawędzi. Działanie tego schematu polega na podziale zbioru wierzchołków na dwa podobne podzbiory. Podział ten jest tak dokonany, aby sumaryczna waga krawędzi między wierzchołkami z poszczególnych podzbiorów była jak najmniejsza. Następnie zamieniane jest ułożenie poszczególnych wierzchołków pomiędzy podzbiorami, tak, aby została znaleziona najkorzystniejsza dyslokacja punktów oraz funkcja celu określana jako różnica między zewnętrznym i wewnętrznym kosztem wybranego wierzchołka była zmaksymalizowana (Kernighan i Lin, 49 s. 291).

Proces Christofidesa jest algorytmem aproksymacyjnym i znajduje rozwiązanie w grafach o nieujemnych wagach krawędzi (https://xlinux.nist.gov/dads). Jego działanie polega na określeniu minimalnego drzewa rozpinającego dla grafu, w którym szukana jest właściwa trasa i dołączeniu do niego krawędzi z najmniejszymi odległościami pomiędzy wierzchołkami drzewa o stopniu nieparzystym. W kolejnym kroku wyznacza się cykl Eulera dla utworzonego grafu, a następnie kreuje cykl Hamiltona. Wynik otrzymany za pomocą tego algorytmu nie może być gorszy od optymalnego o więcej niż 50%.

Algorytm symulowanego wyżarzania wywodzi się z rodziny algorytmów heurystycznych, które przeszukują przestrzeń alternatywnych rozwiązań problemu, aby wyszukać najlepsze rozwiązanie. Polega on najpierw na przyjęciu losowej konfiguracji, aby w następnych krokach przeszukać przestrzeń rozwiązań i znaleźć najlepszy wynik, czyli najkrótszą drogę. Proces oparty jest na losowej wymianie połączeń (Rutczyńska-Wdowiak i Jabłoński, 2016, s. 1524).

Algorytm zachłanny (j. ang. *greedy algorithm*) stosowany jest do rozwiązania problemów optymalizacyjnych, gdzie wymagane jest podejmowanie wielu decyzji. Dokonuje on wyboru, który w danej chwili jest najkorzystniejszy. W danym procesie podejmowane są decyzje „lokalnie" optymalne, co oznacza, że dany algorytm nie dokonuje oceny sensu wykonywanych działań w kolejnych krokach (Cormen, Leiserson, Rivest i Stein, 2009, s. 414).

Do rozwiązania problemu komiwojażera wykorzystano arkusz kalkulacyjny program MS Excel oraz dodatek Solver.

1. STUDIUM PRZYPADKU

Rozwiązanie problemu komiwojażera zostało przedstawione na przykładzie firmy kurierskiej. W prezentowanym przypadku kurier wyrusza z oddziału (miejsce oznaczone na rys. 1 jako 0), następnie musi dotrzeć do 15 punktów ParcelShop (punkty od 1 do 15), pozostawić lub odebrać paczki i wrócić do bazy.

Rys. 1. Mapa z zaznaczonymi punktami ParcelShop
Źródło: opracowanie własne.

Aby rozwiązać dany problem na początku stworzono w arkuszu kalkulacyjnym tabelę 1 zawierającą odległości pomiędzy punktami, które musi odwiedzić kurier.

Tabela 1. Odległości pomiędzy punktami

	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
3	**Punkt**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
4	**0**	0,0	11,5	14,1	14,7	10,8	13,4	10,6	13,2	11,0	10,1	15,6	9,6	16,1	11,3	9,5	9,1
5	**1**	11,5	0,0	2,6	2,2	1,8	2,1	3,0	2,0	2,6	2,3	3,3	4,5	3,9	5,5	4,6	5,6
6	**2**	14,1	2,6	0,0	2,7	3,5	2,9	3,2	3,4	3,5	3,8	3,6	4,7	5,4	5,6	6,0	7,0
7	**3**	14,7	2,2	2,7	0,0	3,3	0,5	4,1	1,6	4,7	4,5	1,7	5,5	3,9	6,5	7,0	7,8
8	**4**	10,8	1,8	3,5	3,3	0,0	3,0	4,6	1,7	3,5	1,0	4,2	6,0	2,2	6,5	4,2	5,2
9	**5**	13,4	2,1	2,9	0,5	3,0	0,0	4,4	1,5	4,6	4,4	2,0	5,8	3,8	6,8	6,9	7,9
10	**6**	10,6	3,0	3,2	4,1	4,6	4,4	0,0	5,4	1,7	4,9	5,6	1,7	7,3	2,6	3,7	4,2
11	**7**	13,2	2,0	3,4	1,6	1,7	1,5	5,4	0,0	3,7	2,9	3,3	7,1	2,3	7,6	5,4	6,3
12	**8**	11,0	2,6	3,5	4,7	3,5	4,6	1,7	3,7	0,0	3,2	4,8	2,6	5,3	3,7	2,3	3,2
13	**9**	10,1	2,3	3,8	4,5	1,0	4,4	4,9	2,9	3,2	0,0	5,3	5,8	2,5	6,3	3,3	5,0
14	**10**	15,6	3,3	3,6	1,7	4,2	2,0	5,6	3,3	4,8	5,3	0,0	6,2	5,5	7,2	7,5	8,5
15	**11**	9,6	4,5	4,7	5,5	6,0	5,8	1,7	7,1	2,6	5,8	6,2	0,0	8,1	2,3	3,8	2,5
16	**12**	16,1	3,9	5,4	3,9	2,2	3,8	7,3	2,3	5,3	2,5	5,5	8,1	0,0	9,2	6,0	7,7
17	**13**	11,3	5,5	5,6	6,5	6,5	6,8	2,6	7,6	3,7	6,3	7,2	2,3	9,2	0,0	5,1	4,5
18	**14**	9,5	4,6	6,0	7,0	4,2	6,9	3,7	5,4	2,3	3,3	7,5	3,8	6,0	5,1	0,0	2,1
19	**15**	9,1	5,6	7,0	7,8	5,2	7,9	4,2	6,3	3,2	5,0	8,5	2,5	7,7	4,5	2,1	0,0

Źródło: opracowanie własne.

Następnie opracowano tabelę 2, która przedstawia punkty jako sekwencję liczb od 1 do 15 oraz odległości pomiędzy nimi. Wartości odległości pomiędzy punktem z danego wiersza, a punktem z wiersza kolejnego importowane są z tabeli odległości (tabela 1) za pomocą funkcji INDEX. Zawiera ona oznaczenie numeru punktów z poszczególnych wierszy oraz określenie numeru wiersza i kolumny będącego przecięciem tych punktów w tabeli odległości.

Baza oznaczona jest cyfrą 0 ponieważ, jest punktem początkowym i końcowym, który nie bierze udziału w losowaniu. Reszta punktów może być ustawiona w każdej możliwej kolejności, a więc są one komórkami zmiennymi.

Punktem wyjścia będzie ustawienie punktów w kolejności naturalnej od 0 do 15, co zostało zestawione w tabeli 2. W tak ustawionej kolejności punktów, całkowita trasa wyniosła 84,9 kilometrów i jest efektem przypadkowego uszeregowania punktów zanim uruchomiony został dodatek Solver. Suma wszystkich odległości jest funkcją celu, która ma dążyć do minimum. Ograniczeniem w tym zadaniu jest fakt, że każdy punkt mógł zostać odwiedzony tylko raz.

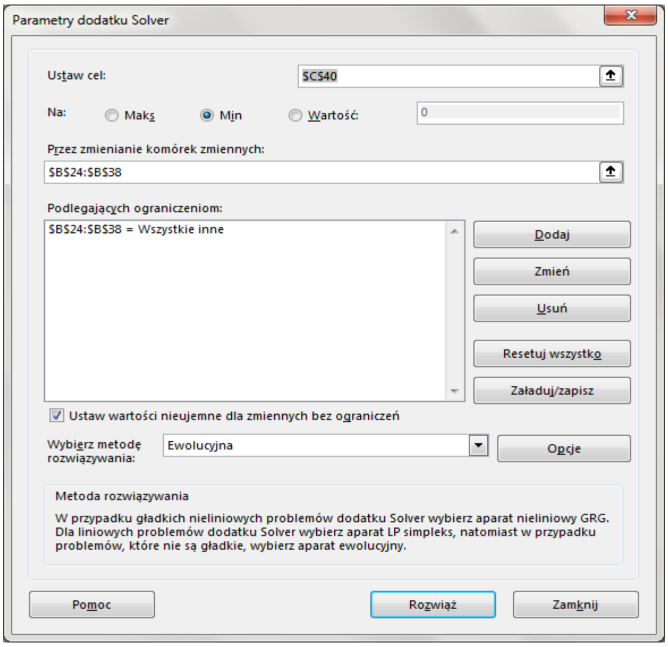
Tabela 2. Relacja pomiędzy punktami ParcelShop wraz ze wskazaniem odległości pomiędzy nimi

	B	C
22	**Punkty**	Odległość
23	**0**	11,5
24	**1**	2,6
25	**2**	2,7
26	**3**	3,3
27	**4**	3
28	**5**	4,4
29	**6**	5,4
30	**7**	3,7
31	**8**	3,2
32	**9**	5,3
33	**10**	6,2
34	**11**	8,1
35	**12**	9,2
36	**13**	5,1
37	**14**	2,1
38	**15**	9,1
39		
40	całkowita długość trasy	**84,9**

Źródło: opracowanie własne.

Najefektywniejszym rozwiązaniem tego typu zadań są algorytmy genetyczne, z tego powodu konieczne jest wybranie metody rozwiązania zadania – metoda ewolucyjna. Dodatkowo zaznaczono, aby komórki zmienne różniły się od siebie oraz żeby szybkość mutacji wyniosła 0,5.

Po wprowadzeniu wszystkich wyżej wymienionych danych do MS Excela i dodatku Solver (rys. 2) oraz zaznaczeniu odpowiednich opcji otrzymano optymalne rozwiązanie.



Rys. 2. Definiowanie parametrów w dodatku Solver
Źródło: opracowanie własne.

Długość najkorzystniejszej trasy wynosi 48,7 kilometra, a punkty układają się w następujący sposób (tabela 3): od punktu 0 poprzez punkty 15 – 14 – 8 – 1 – 4 – 9 – 12 – 7 – 5 – 3 – 10 – 2 – 6 – 13 – 11 i na końcu również występuje punkt 0 (symbolizując powrót).

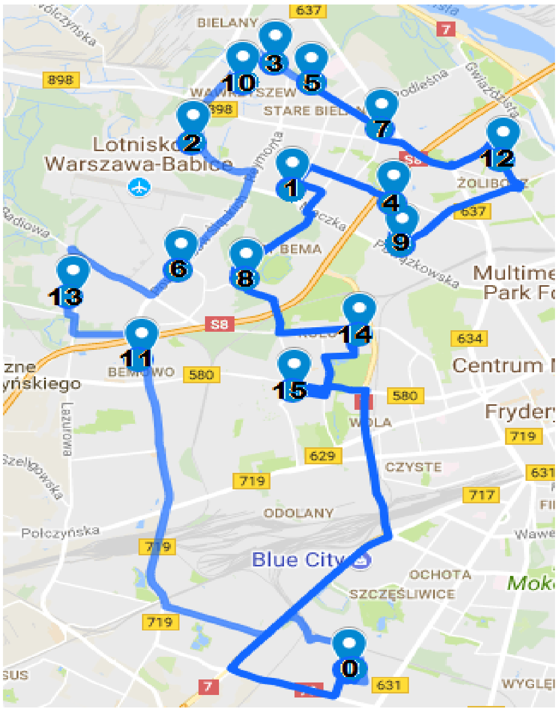
Tabela 3. Relacja pomiędzy punktami ParcelShop, a odległościami pomiędzy nimi

	B	C
22	**Punkty**	Odległość
23	**0**	9,1
24	**15**	2,1
25	**14**	2,3
26	**8**	2,6
27	**1**	1,8
28	**4**	1
29	**9**	2,5
30	**12**	2,3
31	**7**	1,5
32	**5**	0,5
33	**3**	1,7
34	**10**	3,6
35	**2**	3,2
36	**6**	2,6
37	**13**	2,3
38	**11**	9,6
39		
40	całkowita długość trasy	**48,7**

Źródło: opracowanie własne.

2. PODSUMOWANIE

Zoptymalizowaną trasę kierowcy firmy kurierskiej pokazano na rysunku 4. Kurier może wybrać również drogę w odwrotnej pętli, tzn. 0 – 11 – 13 – 6 – 2 – 10 – 3 – 5 – 7 – 12 – 9 – 4 – 1 – 8 – 14 – 15 – 0.



Rys. 4. Zoptymalizowana trasa kierowcy firmy kurierskiej
Źródło: opracowanie własne.

Wybrane przez program Solver rozwiązanie zagwarantowało taki układ przemieszczania z punktu do punktu, dzięki któremu osiągnięto jak najkrótszą trasę. Poprzez taki układ odwiedzania miejsc czas pracy kierowcy, jak również koszty przewozu zostały zminimalizowane (file:///C:/Users/Asus%20Pro/Downloads/artyku_komiwojazer.pdf).

Do rozwiązania problemu komiwojażera również został użyty program MS Excel oraz dodatek Solver. Dzięki nim uzyskano rozwiązanie w postaci kolejności punktów ParcelShop, które powinien odwiedzić kurier, aby jego całkowita trasa była najkrótsza (48,7 km). Przedstawiono mapę z dokładną trasą przejazdu kuriera ze szczegółowym wskazaniem miejsc wraz z kolejnością odbioru lub pozostawienia przesyłki.

LITERATURA:

1. Michalewicz Z., *Algorytmy genetyczne+ struktury danych= programy ewolucyjne*, Wydawnictwo Naukowo-Techniczne, Warszawa 2003.
2. Cormen T.H., Leiserson C.E., Rivest R.L., Stein C., *Introduction to Algorithms,* Massachusetts Institute of Technology, London 2009.
3. Ignasiak E. (red.)., *Badania operacyjne*, PWE, Warszawa 2001.
4. Dziamski O., *Współczesne metody automatycznego planowania dystrybucji towarów*, Logistyka nr 6/2004, ILiM, Poznań 2004.
5. Johnson D. S., McGeoch L. A., *The Traveling Salesman Problem: A Case Study in Local Optimization*, John Wiley and Sons, London 1997.
6. Kernighan B.W., Lin S., *An efficient heuristic procedure for partitioning graphs,* Bell Systems Technical Journal no 49.
7. Lin S., *Computer solutions of the traveling salesman problem*, Bell Systems Technical Journal no 44.
8. Ochelska – Mierzejewska J., *Rozwiązanie problemu komiwojażera przy użyciu algorytmu genetycznego,* Logistyka nr 1/2016, ILiM, Poznań 2016.
9. Rutczyńska-Wdowiak K., Jabłoński N., *Algorytm mrówkowy w problemie komiwojażera,* Autobusy nr 6/2016.
10. Witkowski K., Tanona K., *Analiza kosztów transportu drogowego*, Logistyka nr 5/2013, ILiM, Poznań 2013.
11. http://algorytmy.ency.pl (08.05.2018).
12. http://www.mini.pw.edu.pl (03.05.2018).
13. http://www.staff.amu.edu.pl (03.05.2018).
14. https://xlinux.nist.gov/dads/HTML/christofides.html (09.05.2018).