

Rozwiązywanie problemu komiwojażera za pomocą LP/Quadratic Solver z Analytic Solver Platform v12.5

Solving the travelling salesman problem using LP/Quadratic Solver from the Analytic Solver Platform v12.5

W artykule przedstawiono wyniki rozwiązań przykładów problemu komiwojażera (TSP). Uzyskano je za pomocą LP/Quadratic Solver wchodzącego w skład Analytic Solver Platform v12.5. LP/Quadratic Solver, zaprojektowany do rozwiązywania problemów LP/MIP, pozwala na rozwiązanie TSP w postaci modelu programowania całkowitoliczbowego. Rozwiązania uzyskano w oparciu o wprowadzony do Excela 2010 model problemu przydziału z warunkami ograniczającymi Millera, Tuckera i Zemlina eliminującymi podcykle. Przedstawiono czasy rozwiązań symetrycznych i asymetrycznych przykładów TSP z TSPLIB o małych rozmiarach, ograniczonych przez maksymalną liczbę zmiennych całkowitoliczbowych w LP/Quadratic Solver.

Słowa kluczowe:

arkusz kalkulacyjny, problem komiwojażera, oprogramowanie optymalizacyjne, programowanie całkowitoliczbowe.

The solutions of results of Travelling Salesperson Problem (TSP) samples are presented in this article. They were received using LP/Quadratic Solver included in Analytic Solver Platform V12.5. LP/Quadratic Solver designed for solutions of LP/MIP problems allow to solve TSP as integer programming model. Solutions were received based on Assignment Problem with Miller, Tucker, Zemlin subtour eliminating constraints model introduced to Excel 2010. Solved times of symmetric and asymmetric TSP samples from TSPLIB with small size of problems, limited by max integer variables of LP/Quadratic Solver are presented.

Key words:

spreadsheet, Travelling Salesperson Problem, optimisation software, integer programming.

Wprowadzenie

Problem komiwojażera (w skrócie TSP, od nazwy problemu w j. ang. *Travelling Salesman Problem* lub *Travelling Salesperson Problem*) jest ważnym obiektem zainteresowania badaczy, zajmując ich uwagę od ponad pół wieku. Wynika to z wielu powodów, poczynając od możliwości bezpośredniego zastosowania TSP do rozwiązywania wielu problemów (Applegate, Bixby, Chvátal i Cook, 2007), poprzez pełnienie roli punktu odniesienia do problemów podobnych, a na metodyce nauczania zagadnień PCL kończąc (Pataki, 2003). W wyniku rozwoju zarówno specjalnie opracowywanych metod, jak i wzrostu mocy obliczeniowej komputerów współcześnie istnieje bardzo wiele możliwości podejścia do rozwiązywania TSP.

Znany jest szeroki zakres sformułowań TSP w postaci określonych modeli matematycznych. Jak się wydaje wśród typowych sformułowań można wymienić sformułowanie oparte na modelu problemu przydziału. Do kluczowych elementów formułowanych modeli należą warunki ograniczające eliminujące

podcykle (ang. *Subtour Elimination Constraints* — SEC's). Wśród częściej stosowanych można wymienić np. Dantzig, Fulkersona i Johnsona (DFJ; Dantzig, Fulkerson i Johnson, 1954) i Millera, Tuckera i Zemlina (MTZ; Miller, Tucker i Zemlin, 1960) SEC's.

Wcześniej Premium Solver Platform (PSP), a obecnie Analytic Solver Platform (ASP) (<http://www.solver.com/analytic-solver-platform>; Frontline Systems Inc., 2013) lub Risk Solver Platform (RSP) oprócz korzystania z solverów bezpośrednio wbudowanych posiadają opcję instalowania solverów do rozwiązywania problemów o większych rozmiarach (ang. *Large-scale Solvers*). Dla tych ostatnio wymienionych pojawiły się wyniki badań zawierające informacje dotyczące możliwości rozwiązywania z wykorzystaniem arkusza kalkulacyjnego. Natomiast dla solverów bezpośrednio wbudowanych do ASP autorowi nie udało się znaleźć informacji dotyczących czasu rozwiązywania, w tym wpływu postaci modelu na czas rozwiązywania, czy w ogóle na możliwość jego uzyskania.

W praktyce często rejestruje się czasy uzyskania rozwiązań optymalnych lub określonego stopnia

osiągnięcia wartości takich rozwiązań na tym samym sprzecznie (np. <http://www.tsp.gatech.edu/concorde/>; Rasmussen, 2011b). Daje to, co najmniej z grubsza, orientacyjne informacje o poglądowym charakterze co do efektywności badanych metod i algorytmów, czy trudności wynikających ze struktury badanego problemu. Badania takie często przeprowadza się poprzez wykorzystanie pewnych wzorcowych przykładów problemów, dla których są znane ich wartości rozwiązań optymalnych. Takim ogólnie dostępnym zbiorem TSP jest TSPLIB (<http://comopt.ifl.uni-heidelberg.de/software/TSPLIB95/>). Bywa, że również w tym celu rozwiązuje się przykłady TSP, gdzie odległości między miastami obliczane są z losowo generowanych współrzędnych na dwuwymiarowej siatce (np. Baker, 2011).

Standardowy Solver z Excela 2010 jest ograniczony do 200 zmiennych i 100 warunków ograniczających. Limit rozmiaru TSP, który można rozwiązać za pomocą standardowego Solvera Excela 2010, jest ograniczony dopuszczalnymi liczbami zmiennych i ograniczeń, co pozwala na rozwiązanie problemu TSP o rozmiarze $n = 13$. W Analytic Solver Platform (<http://www.solver.com/analytic-solver-platform>; licencjonowanym komercyjnie lub dostępnym jako dwutygodniowa wersja próbna) ograniczenia są wielokrotnie podwyższone, np. dla LP/Quadratic Solver 8000 zmiennych decyzyjnych, 8000 warunków ograniczających i 2000 całkowitoliczbowych warunków ograniczających (Frontline Systems Inc., 2013).

W odniesieniu do solverów (standardowego z Excela 2010 i wbudowanych do ASP) pojawiły się możliwości rozwiązywania wielu problemów, wcześniej nieosiągalne. W szczególności dotyczy to TSP, np. w wielu pracach (Pataki, 2003; Patterson i Harmel, 2005; Changbing, 2010; Rasmussen, 2011a; Rasmussen, 2011b; Baker, 2011; Węgrzyn, 2013) przedstawiono niektóre możliwości rozwiązywania problemów TSP i TSP-podobnych w arkuszu kalkulacyjnym.

W artykule przedstawiono wyniki badań możliwości rozwiązywania TSP z wykorzystaniem arkusza Excel 2010 wyposażonego w Analytic Solver Platform v12.5. Wykorzystano sformułowanie modelu w oparciu o problem przydziału z warunkami ograniczającymi eliminującymi podcykle Millera, Tuckera, Zemlina. Do obliczeń wykorzystano przykłady problemu komiwojażera symetryczne (sTSP) i asymetryczne (aSTP) z TSPLIB (<http://comopt.ifl.uni-heidelberg.de/software/TSPLIB95/>).

TSP — określenie problemu

Problem komiwojażera może być opisany jak następuje: komiwojażer zamierza odwiedzić klientów w n różnych miejscowościach, odwiedzając każdą z nich jeden raz. Znane są odległości (koszty przejazd-

du) pomiędzy każdą parą miejscowości. W jakiej kolejności powinien komiwojażer odwiedzać miejscowości, aby łącznie przebyta droga pomiędzy nimi była najkrótsza i aby na koniec powrócił do miejscowości, z której rozpoczął podróż?

W podejściu sieciowym do problemu TSP (Sysło, Deo i Kowalik, 1999; Rasmussen, 2011a; Rasmussen, 2011b) miejscowości są węzłami, a drogi łączące miasta — łukami. Występuje zestaw węzłów $N = \{1, 2, \dots, n\}$, gdzie n jest całkowitą liczbą węzłów i zestaw łuków $A = \{(i, j) : i, j \in N, i \neq j\}$, gdzie (ij) wskazuje łuk pomiędzy węzłem i a j . Liczba węzłów n jest używana jako miara rozmiaru problemu. W standardowym TSP zakłada się bezpośrednie połączenia pomiędzy każdą parą węzłów (taka sieć jest nazwana siecią pełną). Komiwojażer może odwiedzić każde miasto tylko jeden raz. Typowe określenie zestawu zmiennych decyzyjnych jest $X = \{(i, j) : i, j \in N, i \neq j\}$, gdzie $x_{ij} = 1$, jeśli komiwojażer podróżuje bezpośrednio z węzła i do węzła j , oraz $x_{ij} = 0$ w przeciwnym przypadku. W TSP jest określona macierz kosztów $C = \{(i, j) : i, j \in N, i \neq j\}$, w której każdy jej element c_{ij} przedstawia koszt podróży (odległość) z węzła i do węzła j . W standardowym TSP najczęściej zakłada się, że kwadratowa macierz kosztów jest symetryczna, tj. $c_{ij} = c_{ji}$, co znaczy, że koszt jest taki sam w obu kierunkach. Kolejnym typowym założeniem jest to, by elementy macierzy kosztów spełniały tzw. warunek trójkąta (nierówność trójkąta). Dla dowolnych trzech węzłów i, j, k powinna być spełniona naturalna własność: $c_{ij} + c_{jk} \geq c_{ik}$, co znaczy, że bezpośrednie połączenie $i - k$ jest nie droższe niż połączenie poprzez węzeł pośredni j . Ponadto komiwojażer powinien wrócić do węzła, z którego rozpoczął podróż (Sysło, Deo i Kowalik, 1999, s. 282): „W języku teorii sieci problem polega na znalezieniu najkrótszego cyklu długości n (zwanego cyklem Hamiltona) w n -wierzchołkowej sieci pełnej”.

Modele LP/MIP

Istnieje szereg możliwych sformułowań TSP, które były analizowane w wielu pracach (np. Dantzig, Fulkerson i Johnson, 1954; Miller, Tucker i Zemlin, 1960; Langevin i Suomis, 1990; Gutin i Punnen, 2004, s. 15–21; Orman i Williams, 2005; Chauhudri i De, 2010; Rasmussen, 2011a; Rasmussen, 2011b). TSP może być postrzegany jako problem permutacji (Gutin i Punnen, 2004, s. 3–4; Chauhudri i De, 2010; Rasmussen, 2011a). Większość sformułowań programowania całkowitoliczbowego aTSP bazuje na problemie przydziału. Takie sformułowanie wymaga dodatkowych ograniczeń eliminujących podcykle. Ograniczenia te nazywają się warunkami ograniczającymi eliminacji podcykli (ang. *Subtour Elimination Constraints* — SEC's). Pierwsze sformułowanie DFJ SEC's (Dantzig, Fulkerson i Johnson, 1954; Gutin i Punnen, 2004,

s. 15–21; Pataki, 2003) posiada eksponentyjną liczbę SEC's. Wielomianową liczbę SEC's posiada sformułowanie MTZ (Miller, Tucker i Zemlin, 1960; Gutin i Punnen, 2004, s. 15–21; Orman i Williams, 2005). Jeszcze inne sformułowania wykorzystują warunki ograniczające opisujące przepływ towaru w sieci (ang. *Network Flow Constraints*; Gutin i Punnen, 2004, s. 15–21; Langevin i Suomis, 1990; Orman i Williams, 2005), gdzie można wymienić warunki opisujące przepływ pojedynczego towaru, przepływ dwóch towarów i przepływ wielu towarów w sieci.

Sformułowanie oparte na problemie przydziału

Poniżej przedstawiono sformułowanie oparte na problemie przydziału (ang. *Assignment Problem*):

$$f(x_{ij}) = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \rightarrow \min \quad (1)$$

$$\sum_{i=1}^n x_{ij} = 1 \quad (j = 1, 2, \dots, n) \quad (2)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad (i = 1, 2, \dots, n) \quad (3)$$

$$x_{ij} \in \{0, 1\} \quad (i, j = 1, 2, \dots, n) \quad (4)$$

Dodatkowo niezbędne są warunki ograniczające eliminujące podcykle (SEC's).

Sformułowanie wyrażone wzorami (1)–(3) zakłada sieć pełną i jeśli dane zorganizować w postaci macierzy kwadratowych, elementy leżące na przekątnej głównej (c_{ii} , x_{ii}) również są włączone do modelu. W sieci pełnej jedyne łuki, jakie nie istnieją, znajdują się właśnie na przekątnej głównej. Są to zmienne x_{ii} . Ze względu na prostotę organizacji arkusza i łatwość wprowadzania danych w przypadku sieci pełnej często stosuje się duże wartości kosztów $c_{ii} = M$, co zapewnia niewprowadzanie nieistniejących łuków x_{ii} do rozwiązania (nie musi tak być w przypadku sieci niepełnej, ale takich przykładów w niniejszej pracy nie badano). Takie podejście powoduje zwiększenie rozmiaru modelu o n zmiennych.

Warunki ograniczające eliminujące podcykle (SEC's)

Pierwotnie nieco szerzej sformułowane MTZ SEC's (Miller, Tucker i Zemlin, 1960) w przypadku oznaczenia miasta bazowego indeksem 1 dla jednego cyklu obejmującego wszystkie miasta są następujące:

$$u_i - u_j + 1 \leq (n - 1)(1 - x_{ij}) \quad \forall (i, j) \in A, : i, j, \quad (5)$$

z uzupełnieniem (Pataki, 2003):

$$u_1 = 1, \quad (6)$$

$$2 \leq u_i \leq n - \forall i \neq 1 \quad (7)$$

gdzie:

x_{ij} — nieujemne liczby całkowite,
 u_i ($i = 1, \dots, n$) — arbitralne liczby rzeczywiste, wobec których wymogi mogą być ograniczone do liczb nieujemnych całkowitych; reprezentują sekwencję węzłów, które są odwiedzane.

Nierówności zawierające u_i służą do eliminowania cykli, które nie rozpoczynają się i nie kończą się w mieście bazowym 1 oraz u_i mogą być tak dostosowane, że $u_i = j$ dla j -tego miasta odwiedzanego w cyklu, który zawiera miasto i .

Warunki określone wzorem (5) można zapisać równoważnie:

$$u_i - u_j + (n - 1)x_{ij} \leq n - 2 \quad \forall (i, j) \in A, : i, j, \neq 1 \quad (8)$$

Biorąc pod uwagę tylko zmienne występujące w rzeczywistości, przy rozmiarze problemu n (miast) wystąpi $n(n - 1)$ zmiennych binarnych x_{ij} , n zmiennych u_i , $2n$ warunków ograniczających przydziału, $n(n - 1)$ warunków ograniczających eliminacji podcykli. W przyjętej w niniejszych badaniach macierzowej organizacji arkusza występuje n^2 zmiennych binarnych x_{ij} . Powoduje to, że największy rozmiar problemu, który może być rozwiązany za pomocą LP/Quadratic (limit 2000 całkowitoliczbowych warunków ograniczających) wynosi $n = 44$. Stąd, przy $n = 44$, $x_{ij} : n^2 = 1936$, $u_i : n = 44$, razem zmiennych: 1980, warunków ograniczających przydziału plus eliminacji podcykli: $2n + n(n - 1) = 1980$. (Z kolei przy $n = 45$, $x_{ij} : n^2 = 2025$, $u_i : n = 45$, razem zmiennych: 2070, warunków ograniczających przydziału plus eliminacji podcykli: $2n + n(n - 1) = 2070$).

Sformułowanie i rozwiązanie problemu z wykorzystaniem arkusza kalkulacyjnego

Przykład z TSPLIB — burma14

Przykłady TSP w TSPLIB są zamieszczone w formie plików wymagających rozpakowania. Dane w plikach występują w postaci kilku różnych typów zapisu i wymagają odpowiedniego przetworzenia w zależności od typu w celu uzyskania pełnej macierzy odległości c_{ij} . Sposoby przetwarzania danych są opisane w odpowiedniej dokumentacji TSPLIB (<http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>).

Zapis w arkuszu i użycie Solvera

Oryginalny plik z danymi po rozpakowaniu `bays29.tsp` zaimportowano do arkusza, gdzie po odpowiednim przeorganizowaniu danych uzyskane wartości c_{ij} rozmieszczono w formie macierzy w zakresie komórek `AJ9:BL37` arkusza przedstawionego na rysunku 2.

Na głównej przekątnej dla odległości c_{ij} wprowadzono duże wartości =1000. Wykaz formuł użytych do obliczeń w arkuszu przedstawiono w tabeli 1. Wpisy do pól klasycznego okna dialogowego Parametry dodatku solver przedstawiono na rysunku 1, a okna dialogowe Specyfikacji Modelu i Opcji Solvera (Model i Output) Analytic Solver Platform v12.5 na rysunku 3.

W problemie występują dwa rodzaje zmiennych decyzyjnych: binarne x_{ij} i całkowitoliczbowe u_i . Zmienne decyzyjne x_{ij} rozmieszczono w zakresie komórek AJ41:BL69. Ich wartości startowe ustalano na zero. Lewe strony (LHS) warunków ograniczających dla tych zmiennych rozmieszczono w zakresach BM41:BM69 i AJ70:BL70. Z kolei na zmienne u_i przewidziano zakres komórek AG41:AG69, gdzie wartości startowe tych zmiennych (przed uruchomieniem Solvera), były ustalone w kolejności rosnącej liczb oznaczających miasta.

Funkcje celu wprowadzono do komórki AG71. LHS SEC's rozmieszczono w zakresie komórek AJ75:BL103 z wyjątkiem elementów na głównej przekątnej, gdzie wpisano zera. Zakres komórek AJ75:BL75 nie wchodzi do obliczeń, dlatego LHS SEC's obejmuje zakres AJ76:BL103.

Na rysunku 2 przedstawiono istotną część arkusza po uzyskaniu rozwiązania. Wartości u_i rozwiązania wyznaczają kolejność odwiedzanych miast, ale ich właściwe odczytanie nie jest oczywiste. Mianowicie wartość $u_i = 1$ znajduje się w komórce AG61, a po prawej i lewej stronie znajdują się komórki AI61 i AF61 zawierające numer miasta, do którego rozpoczyna się marszruta. Jest to numer miasta 21. Wartość $u_i = 2$ znajduje się w komórce AG53 na wysokości komórek AI53 i AF53, wskazując na numer miasta 13, itd. Łatwo to sprawdzić, badając rozmieszczenie zmiennych x_{ij} . Rozpoczynając podróż z miasta o numerze 1 (komórka AI41), $x_{ij} = 1$ znajduje się w komórce BD41, wskazując numer miasta 21 (w komórce BD42). Kontynuując podróż z miasta o numerze 21 (komórka AI61), $x_{ij} = 1$ znajdziemy w komórce AV61, wskazującej na numer miasta 13 (w komórce AV40) itd. Przy założeniu, że startujemy z miasta o numerze 1, zgodnie z wzorem (7) u_i

nie jest równe 1. Dlatego zakres komórek AJ75:BL75, odpowiadający miastu numer 1, jest wyłączony z SEC's.

Uzyskane rozwiązanie optymalne przedstawione na rys. 2 wykazuje następujący porządek odwiedzanych miast: 1 – 21 – 13 – 16 – 24 – 8 – 27 – 23 – 7 – 25 – 19 – 11 – 22 – 17 – 14 – 18 – 15 – 4 – 10 – 20 – 2 – 3 – 29 – 26 – 5 – 9 – 12 – 6 – 28 – 1 (lub w kolejności odwrotnej) z całkowitą długością marszruty wynoszącą 2022.

Ze względu na to, że w problemie są podane współrzędne węzłów, możliwe jest sporządzenie wykresu przedstawiającego położenie węzłów i optymalnej marszruty stanowiącej cykl Hamiltona o minimalnej długości (tab. 2; rys. 4). W tym celu zorganizowano wartości u_i w kolejności odwiedzanych miast w zakresie komórek AC41:AC69. Funkcja arkusza WYSZUKAJ. PIONOWO (rozmieszczona w zakresie komórek AC41:AC49) wyszukuje wartości z zakresu B41:B69 (nr miasta) w zakresie AE41:AE69 (powtórzone wartości u_i rozwiązania) i odczytuje wartości z drugiego zakresu AF41:AF69. Dane do wykresu rozmieszczono w zakresach C41:D69 i F41:G70. W pierwszym zakresie znajdują się współrzędne węzłów oznaczających miasta. W drugim współrzędne węzłów w kolejności ich odwiedzania (marszruty). Pierwsze na wykresie wykreślono tylko jako punkty, a drugie tylko jako linie łączące. Funkcje arkusza INDEKS wyszukują w zakresie B41:D69 współrzędne $x[i]$ i $y[i]$ węzłów z zakresu AC41:AC69 i umieszczają ich wartości odpowiednio w zakresach F41:F69 i G41:G69. Komórki F70 i G70 równają się odpowiednio komórkom F41 i G41, jako że komiwojażer powinien powrócić do miasta, z którego rozpoczął podróż.

Komputer, oprogramowanie i Opcje Solvera

Badania wykonano na komputerze wyposażonym w procesor Intel® Core™ i5-2520M CPU 2,50GHz, RAM: 4 GB z systemem Windows 7 Professional

Tabela 1

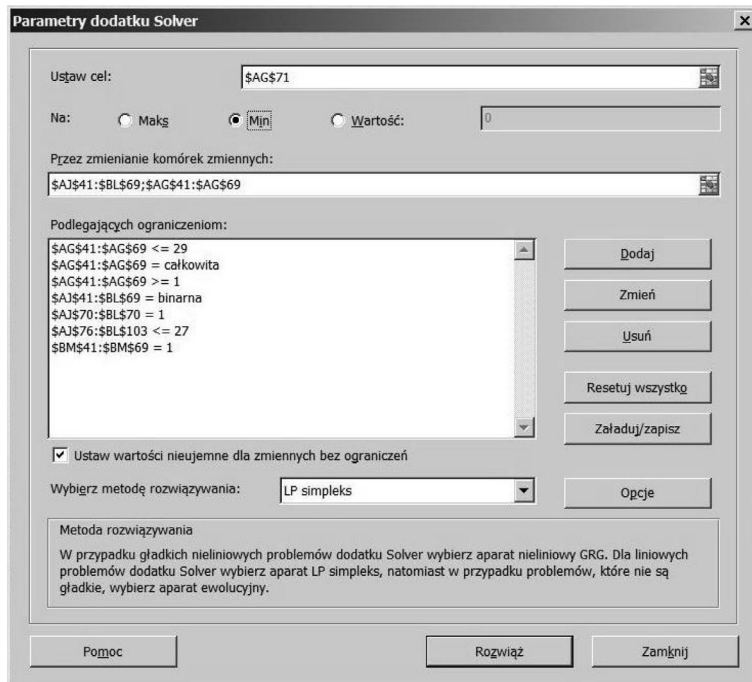
Wykaz formuł w modelu problemu bays29

Komórka	Formuła	Skopiowana do
AJ70	=SUMA (AJ41: AJ69)	AK70:BL70
BM41	=SUMA (AJ41: BL41)	BM42:BM69
AJ76	=WYSZUKAJ. PIONOWO (\$AI76; \$AF\$41: \$AG\$69; 2; FAŁSZ) WYSZUKAJ. PIONOWO (AJ\$74; \$AF\$41: \$AG\$69; 2; FAŁSZ) + \$AG\$76*AJ42	pozostałych komórek zakresu AJ75:BL103 z wyjątkiem głównej przekątnej, gdzie każda komórka=0
AG71	=SUMA. ILOCZYNÓW (AJ9: BL37; AJ41: BL69)	

Źródło: opracowanie własne.

Rysunek 1

Klasyczne Okno dialogowe Parametry dodatku Solver problemu bays29



Źródło: opracowanie własne.

Rysunek 2

Rozwiązanie optymalne problemu bays29; elementy c_{ij} , x_{ij} , u_i trzech macierzy

AG71		-SUMA.ILOCZYNÓW(AJ9:BL37;AJ41:BL69)																																
AF	AG	AH	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY	AZ	BA	BB	BC	BD	BE	BF	BG	BH	BI	BJ	BK	BL	BM		
7			Do																															
8			Do																															
9			Do																															
10			Do																															
11			Do																															
12			Do																															
13			Do																															
14			Do																															
15			Do																															
16			Do																															
17			Do																															
18			Do																															
19			Do																															
20			Do																															
21			Do																															
22			Do																															
23			Do																															
24			Do																															
25			Do																															
26			Do																															
27			Do																															
28			Do																															
29			Do																															
30			Do																															
31			Do																															
32			Do																															
33			Do																															
34			Do																															
35			Do																															
36			Do																															
37			Do																															
38			Do																															
39			Do																															
40	Miasto	u_i	x_{ij}	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29		
41	1	29	Z	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
42	2	20		0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
43	3	21		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
44	4	17		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
45	5	24		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
46	6	27		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
47	7	8		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
48	8	5		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
49	9	26		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
50	10	18		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
51	11	11		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
52	12	28		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
53	13	2		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
54	14	14		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
55	15	16		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
56	16	3		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
57	17	13		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
58	18	16		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
59	19	10		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	20	19		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
61	21	1		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
62	22	12		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
63	23	7		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
64	24	4		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
65	25	9		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
66	26	23		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
67	27	6		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
68	28	28		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
69	29	22		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
70				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
71				2022 f(xij)->min																														

cd. rysunku 2

	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY	AZ	BA	BB	BC	BD	BE	BF	BG	BH	BI	BJ	BK	BL	BM	
73																																			
74	U=uj+(n-1)xj=-cn-2			SEQ's	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29		
75	n	29			0	9	8	12	5	2	21	24	4	11	18	3	27	15	13	26	16	14	19	10	56	17	22	25	20	6	23	1	7		
76	n-1	28			2	-9	0	27	3	-4	-7	12	15	-5	2	9	-6	18	6	4	17	7	5	10	1	19	8	13	16	11	-3	14	-8	-2	
77	n-2	27			3	-8	1	0	4	-3	-6	13	16	-4	3	10	-5	19	7	5	18	8	6	11	2	20	9	14	17	12	-2	15	-7	27	
78					4	-12	-3	-4	0	-7	-10	9	12	-8	27	6	-9	15	3	1	14	4	2	7	-2	16	5	10	13	8	-6	11	-11	-5	
79					5	-5	4	3	7	0	-3	16	19	27	6	13	-2	22	10	8	21	11	9	14	5	23	12	17	20	15	1	18	-4	2	
80					6	-2	7	6	10	3	0	19	22	2	9	16	1	25	13	11	24	14	12	17	8	26	15	20	23	18	4	21	27	5	
81					7	-21	-12	-13	-9	-16	-19	0	3	-17	-10	-3	-18	6	-6	-8	5	-5	-7	-2	-11	7	-4	1	4	27	-15	2	-20	-14	
82					8	-24	-15	-16	-12	-19	-22	-3	0	-20	-13	-6	-21	3	-9	-11	2	-8	-10	-5	-14	4	-7	-2	1	-4	-18	27	-23	-17	
83					9	-4	5	4	8	1	-2	17	20	0	7	14	27	23	11	9	22	12	10	15	6	24	13	18	21	16	2	19	-3	3	
84					10	-11	-2	-3	1	-6	-9	10	13	-7	0	7	-8	16	4	2	15	5	3	8	27	17	6	11	14	9	-5	12	-10	-4	
85					11	-18	-9	-10	-6	-13	-16	3	6	-14	-7	0	-15	9	-3	-5	8	-2	-4	1	-8	10	27	4	7	2	-12	5	-17	-11	
86					12	-3	6	5	9	2	27	18	21	1	8	15	0	24	12	10	23	13	11	16	7	25	14	19	22	17	3	20	-2	4	
87					13	-27	-18	-19	-15	-22	-25	-6	-3	-23	-16	-9	-24	0	-12	-14	27	-11	-13	-8	-17	1	-10	-5	-2	-7	-21	-4	-26	-20	
88					14	-15	-6	-7	-3	-10	-13	6	9	-11	-4	3	-12	12	0	-2	11	1	27	4	-5	13	2	7	10	5	-9	8	-14	-8	
89					15	-13	-4	-5	27	-8	-11	8	11	-9	-2	5	-10	14	2	0	13	3	1	6	-3	15	4	9	12	7	-7	10	-12	-6	
90					16	-26	-17	-18	-14	-21	-24	-5	-2	-22	-15	-8	-23	1	-11	-13	0	-10	-12	-7	-16	2	-9	-4	27	-6	-20	-3	-25	-19	
91					17	-16	-7	-8	-4	-11	-14	5	8	-12	-5	2	-13	11	27	-3	10	0	-2	3	-6	12	1	6	9	4	-10	7	-15	-9	
92					18	-14	-5	-6	-2	-9	-12	7	10	-10	-3	4	-11	13	1	27	12	2	0	5	-4	14	3	8	11	6	-8	9	-13	-7	
93					19	-19	-10	-11	-7	-14	-17	2	5	-15	-8	27	-16	8	-4	-6	7	-3	-5	0	-9	9	-2	3	6	1	-13	4	-18	-12	
94					20	-10	27	-2	2	-5	-8	11	14	-6	1	8	-7	17	5	3	16	6	4	9	0	18	7	12	15	10	-4	13	-9	-3	
95					21	-28	-19	-20	-16	-23	-26	-7	-4	-24	-17	-10	-25	27	-13	-15	-2	-12	-14	-9	-18	0	-11	-6	-3	-8	-22	-5	-27	-21	
96					22	-17	-8	-9	-5	-12	-15	4	7	-13	-6	1	-14	10	-2	-4	9	27	-3	2	-7	11	0	5	8	3	-11	6	-16	-10	
97					23	-22	-13	-14	-10	-17	-20	27	2	-18	-11	-4	-19	5	-7	-9	4	-6	-8	-3	-12	6	-5	0	3	-2	-16	1	-21	-15	
98					24	-25	-16	-17	-13	-20	-23	-4	27	-21	-14	-7	-22	2	-10	-12	1	-9	-11	-6	-15	3	-8	-3	0	-5	-19	-2	-24	-18	
99					25	-20	-11	-12	-8	-15	-18	1	4	-16	-9	-2	-17	7	-5	-7	6	-4	-6	27	-10	8	-3	2	5	0	-14	3	-19	-13	
100					26	-6	3	2	6	27	-4	15	18	-2	5	12	-3	21	9	7	20	10	8	13	4	22	11	16	19	14	0	17	-5	1	
101					27	-23	-14	-15	-11	-18	-21	-2	1	-19	-12	-5	-20	4	-8	-10	3	-7	-9	-4	-13	5	-6	27	2	-3	-17	0	-22	-16	
102					28	27	8	7	11	4	1	20	23	3	10	17	2	26	14	12	25	15	13	18	9	27	16	21	24	19	5	22	0	6	
103					29	-7	2	1	5	-2	-5	14	17	-3	4	11	-4	20	8	6	19	9	7	12	3	21	10	15	18	13	27	16	-6	0	

Źródło: opracowanie własne.

Rysunek 3

Okna dialogowe Opcji Solvera i Specyfikacji Modelu (Model i Output)
Analytic Solver Platform v12.5 dla problemu bays29

Solver Options and Model Specifications

Model | Platform | Engine | Output

Recourse

Constraints

- Normal
 - \$AJ\$70:\$BL\$70 = 1
 - \$AJ\$76:\$BL\$103 <= 27
 - \$BM\$41:\$BM\$69 = 1
- Chance
- Recourse
- Bound
 - \$AG\$41:\$AG\$69 <= 29
 - \$AG\$41:\$AG\$69 >= 1

Model Diagnosis

Model Type: LP/MIP

Variables - Functions - Dependencies	Vars	Fcns	Dpns
All	870	871	4875
Smooth	870	871	4875
Linear	870	871	4875
Recourse	0	0	0
Uncertain	0	0	0

Other Model Elements

Bounds	1740
Integers	870

Model Type

If Unknown, press the 'Analyze without Solving' button to diagnose the model.

Structure Check and Convexity Test Completed

Solver Options and Model Specifications

Model | Platform | Engine | Output

Diagnosis started...
Convexity testing started...
Model diagnosed as "LP/MIP".
Automatic engine selection: Standard
LP/Quadratic
Model: [bays29 perm ASP.tsp.xlsx]bays29
AP-0x
Using: Psi Interpreter
Parse time: 0,59 Seconds.

Engine: Standard LP/Quadratic
Setup time: 0,05 Seconds.

Engine Solve time: 2 Minutes, 45 Seconds.

Integer solution found within tolerance.
Solve time: 2 Minutes, 52 Seconds.

Best Integer Objective	2022
Current Objective	2002,35714285714
Nodes	33970
Iterations	0
Relaxed Objective	1775,07
Best Possible Objective	2001,79
Integer Gap	0,0100981

Integer solution found within tolerance.

Źródło: opracowanie własne.

Tabela 2

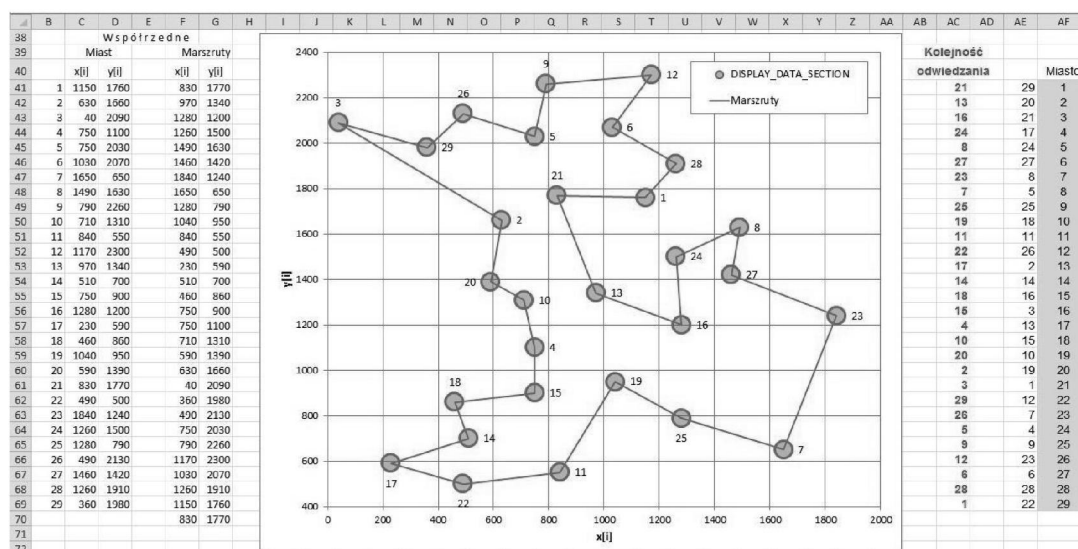
Wykaz formuł do wykresu problemu bays29

Komórka	Formuła	Skopiowana do
AE41	=AG41	AE42:AE69
AC41	=WYSZUKAJ. PIONOWO (B41; \$AE\$41: \$AF\$69; 2; FAŁSZ)	AC42:AC69
F41	=INDEKS (\$B\$41: \$D\$69; AC41; 2)	F42:F69
G41	=INDEKS (\$B\$41: \$D\$69; AC41; 3)	G42:G69
F70	=F41	
G70	=G41	

Źródło: opracowanie własne.

Rysunek 4

Marszruta rozwiązania optymalnego (w ramach optymalności całkowitoliczbowej = 1%) problemu bays29



Źródło: opracowanie własne.

64-bitowy Service Pack 1. Do obliczeń wykorzystano Analytic Solver Platform v12.5 (dostępny również w bezpłatnej dwutygodniowej wersji próbnej) (<http://www.solver.com/analytic-solver-platform>). ASP zawiera pięć wbudowanych różnych solverów, wśród nich LP/Quadratic Solver (Frontline Systems Inc., 2013) przeznaczony do rozwiązywania zadań programowania liniowego i mieszanego programowania całkowitoliczbowego (ang. *Linear Programming/Mixed Integer Programming* — LP/MIP).

Po wprowadzeniu modelu problemu szczegółowy rodzaj metody/solvera (Engine — w oryginale) jest zarówno dobierany automatycznie przez ASP, jak i może być wybrany przez użytkownika. Wybór Standard LP/Quadratic (w oknie Output ASP) odpowiada Metodzie LP Simpleks (w klasycznym oknie Parametry dodatku Solver). Opcje Solvera pozostawiono w standardowych ustawieniach producenta, które wynosiły: dokładność ograniczeń 0,00000001, skalowanie automatyczne i optymalność całkowitoliczbową 1%.

Zestawienie rozwiązań przykładów problemów z TSPLIB

Oprócz bays29 rozwiązano jeszcze inne przykłady problemów symetrycznych i niesymetrycznych TSP. Rodzaj problemu, jego nazwę, rozmiar, ustaloną wartość rozwiązania optymalnego (Minimum) w TSPLIB oraz uzyskane za pomocą LP/Quadratic Solver z ASP wartości rozwiązania $f(x_i)$ (minimalna długość marszrut), odchylenie od wartości optymalnej rozwiązania (Odch., % = $\{f(x_i) - \text{Minimum}\} / \text{Minimum} * 100\%$) i czas rozwiązywania problemu przedstawiono w tabeli 3.

Zarówno dla siedmiu przykładów sTSP, jak i czterech aTSP uzyskano dokładne wartości rozwiązania i z niewielkim odchyleniem od wartości Minimum z TSPLIB. W jednym przypadku nie uzyskano rozwiązania. Czas rozwiązań przykładów symetrycznych burma14, gr17, gr21, gr24, fri26 osiągnął rząd kilkunastu sekund, a w przypadku bays29 i swiss42 wyraźnie wzrósł, osiągając 172 i 1230 s. W przypadku problemów asymetrycznych do rozmiaru 39 wynosi kilkadziesiąt sekund, natomiast

w przypadku p43 nie uzyskano rozwiązania w czasie 1860 s. Obydwe wartości rozwiązania (bays29 i swiss42) mieszczą się w zakresie <1%. Ze względu na ustawienie tolerancji całkowitoliczbowej na 1% można przypuszczać, że w przypadku jej zmniejszenia uzyskano by dokładniejsze wartości rozwiązań, chociaż prawdopodobnie skutkowałyby to wzrostem czasu uzyskania rozwiązania.

Ogólnie można stwierdzić, że w uzyskanych wynikach widać pewien wpływ rozmiaru problemu (n) i jego struktury (symetryczny, asymetryczny). Z grubsza trudność uzyskania rozwiązania rośnie ze wzrostem rozmiaru problemu. Asymetryczne TSP trudniej rozwiązać niż symetryczne TSP.

Dla przykładów z TSPLIB (w tym znacznie większych rozmiarów niż wymienione w tabeli 3) przedstawiono wyniki obliczeń, np. dla rozwiązywanych za pomocą solverów większej skali z Analytic Solver Platform w pracach R. Rasmussena i K.R. Bakera (Rasmussen, 2011b; Baker, 2011) oraz Solverów CONCORDE (<http://www.tsp.gatech.edu/concorde/>). Biorąc pod uwagę różne komputery użyte w badaniach, dane te nie mogą być podstawą bezpośrednich porównań. Niemniej dają informacje, o poglądowym charakterze, co do możliwości badanego oprogramowania (Solvency CONCORDE — <http://www.tsp.gatech.edu/concorde/> — nie są predysponowane do arkusza kalkulacyjnego, z kolei w obliczeniach, m.in. Baker, 2011, wykorzystano inny typ przykładów testowych).

Obliczenia wykonane w tym artykule i szeroko zakrojone obliczenia z pracy R. Rasmussena (Rasmussen, 2011b) dotyczą kilku tych samych przykładów: sy-

metrycznego fri26 i asymetrycznych br17, ftv33 i p43. W pracy tego autora (Rasmussen, 2011b) obliczenia 20 oryginalnych przykładów TSPLIB (plus przykłady zmodyfikowane) wykonano w Excelu używając dołączalnych do ASP Solverów Gurobi i Xpress oraz korzystając z serwera NEOS z Solverem Scip, stosując „szersze” i „węższe” sformułowanie problemu w postaci modelu przepływu. Uzyskane rezultaty kilku tych samych przykładów zestawiono w tabeli 4.

Cztery przykłady z tabeli 4 nie mogą być dobrą podstawą do dalej idących uogólnień, ale widać, że solwery użyte w pracy (Rasmussen, 2011b) działają szybciej, chociaż w przypadku dość małego problemu br17 wykazują zaskakująco większy czas rozwiązywania. Ciekawostką jest również to, że w przypadku p43 nie uzyskano rozwiązania w niniejszych badaniach i w pracy R. Rasmussena (Rasmussen, 2011b), podczas gdy uzyskano rozwiązanie (z Odch. = 0,07%) stosując Metodę Ewolucyjną ze standardowego Solvera z Excela 2010 (Węgrzyn, 2013).

Podsumowanie

W artykule przedstawiono przykład rozwiązania problemu komiwojażera w oparciu o wprowadzony do arkusza kalkulacyjnego Excel 2010 model problemu przydziału z warunkami ograniczającymi Millera, Tuckera i Zemlina eliminującymi podcykle (MTZ SEC's). Przedstawiono uzyskane czasy rozwiązań optymalnych 7 przykładów symetrycznych

Tabela 3

Wartości rozwiązań i czasu rozwiązywania Solvera za pomocą Analytic Solver Platform v12.5 w Excelu 2010 (optymalność całkowitoliczbowa = 1%)

Lp	Problem z TSPLIB			Sformułowania w oparciu o problem przydziału			
	Nazwa	n	Minimum	f(x _{ij})	Odch., %	Solve time, s	Uwagi
Symmetric							
1	burma14	14	3323	3323		8,6	
2	gr17	17	2085	2085		14,2	
3	gr21	21	2707	2707		16,7	
4	gr24	24	1272	1272		17,4	
5	fri26	26	937	937		18,8	
6	bays29	29	2020	2022	0,10	172,0	
7	swiss42	42	1273	1285	0,94	1 230,0	*1), *2)
Asymmetric							
1	br17	17	39	39		30,0	*3)
2	ftv33	34	1286	1286		49,3	
3	ftv38	39	1530	1530		36,8	
4	p43	43	5620				Brak rozwiązania w ciągu 1860 s

*1 Przybliżony Solve time.

*2 Solver zatrzymano na żądanie użytkownika po 2043 s.

*3 Pomimo uzyskania najlepszej całkowitoliczbowej wartości funkcji celu proces rozwiązywania był kontynuowany dopóki Solver nie zatrzymał się z powodu osiągnięcia limitu maksymalnego czasu.

Źródło: opracowanie własne.

Tabela 4

Czas rozwiązywania (w sekundach) za pomocą różnych solverów:
ASPLP/Quadratic i Gurobi, Xpress oraz NEOS/Scip

Problem z TSPLIB		ASP, Solver LP/Quadratic	RSP Solvery Gurobi i Xpress (Rasmussen, 2011b)	Serwer NEOS, Solver Scip (Rasmussen, 2011b)
Nazwa	n	Sformułowanie problemu		
		problem przydziału	problem przepływu (Rasmussen, 2011b)	
fri26 (sTSP)	26	18,8	2,78 ÷ 4,60	3,77 ÷ 9,81
br17 (aTSP)	17	30	66 ÷ 665	636 ÷ 1786
ftv33 (aTSP)	34	49,3	3,92 ÷ 5,15	3,65 ÷ 15,26
p43 (aTSP)	43	Brak rozwiązania w czasie 1860 s	Przekroczenie pamięci	Rozwiązywanie przerwane

Źródło: opracowanie własne dotyczące wyników z niniejszych badań i z pracy R. Rasmussena (Rasmussen, 2011b).

i 4 asymetrycznych z TSPLIB, przy tolerancji całkowitoliczbowej 1%. W odróżnieniu od innych metod, w tym np. Metody Ewolucyjnej, gdzie rozwiązując problem o nieznannej wartości rozwiązania optymalnego, nie wiadomo, w jakiej odległości od niego znajduje się rozwiązanie uzyskane, podejście programowania całkowitoliczbowego zasadniczo daje gwarancje znalezienia rozwiązania optymalnego.

Pomimo, że rozmiar badanych problemów w niniejszym artykule w odniesieniu do praktycznych problemów wydaje się być rozmiarem, który może być określany jako mały, to jednak względna prostota organizacji arkusza jest pewnym atutem przedstawionego podejścia. *Analytic Solver Platform* (ASP) jest pakietem licencjonowanym komercyjnie, chociaż istnieje

możliwość skorzystania z dwutygodniowej wersji próbnej. Pewnym minusem ASP są ograniczenia dotyczące rozmiarów liczby zmiennych i ograniczeń, niemniej są one kompensowane możliwością włączenia do ASP innych solverów umożliwiających rozwiązywanie problemów o większych rozmiarach. *Analytic Solver Platform* pozwala łączyć łatwość korzystania z Excela i doświadczenie w budowaniu modelu arkuszowego z większą funkcjonalnością i mocą obliczeniową.

Arkusz Excel jako platforma do zbudowania modelu matematycznego zyskał znaczne możliwości w postaci gotowego oprogramowania do rozwiązywania problemów kombinatorycznych, do niedawna o nieosiągalnych rozmiarach.

Literatura

- Applegate, D.L., Bixby, R.E., Chvátal, V. and Cook, W.J. (2007). *The traveling salesman problem: a computational study*. NJ, USA: Princeton University Press.
- Baker, K.R. (2011). Solving sequencing problems in spreadsheets. *Int. J. Planning and Scheduling*, 1(1/2), 3–18.
- Changbing, J. (2010). A Reliable Solver of Euclidean Traveling Salesman Problems with Microsoft Excel Addin Tools for Smallsize System. *Journal of Software*, 5(7), 761–768.
- Chaudhuri, A., De K. (2010). A study of Traveling Salesman Problem Using Fuzzy Self Organizing Map. In: D. Davendra (edited) *Traveling Salesman Problem, Theory and Applications*. InTech 2010; <http://www.intechopen.com> (2013–04–15).
- Dantzig, G., Fulkerson, R., Johnson, S. (1954). Solution of a Large-Scale Traveling-Salesman Problem. Reviewed work(s): Source: *Journal of the Operations Research Society of America*, 2(4), 393–410. Published by: INFORMS Stable URL: <http://www.stor.org/stable/166695> (Accessed: 11/04/2012 17: 52).
- Gutin, G., Punnen, A.P. (edited). (2004). *The Traveling Salesman Problem And Its Variations*. Kluwer Academic Publishers.
- Langevin, A., Suomis, F. (1990). Classification of Travelling Salesman Problem formulations. *Operations Research Letters*, (9), 127–132.
- Miller, C.E., Tucker, A.W., Zemlin, R.A. (1960). Integer Programming Formulation of Traveling Salesman Problems. *Journal of ACM*, (7), 326–329.
- Orman, A.J. and Williams, H.P. (2004). A survey of different integer programming formulations of the travelling salesman problem. LSE Research Online, March 09 2004 Revised July 2005; http://eprints.lse.ac.uk/9349/1/WP67_A_Survey_of_DifferentFormulationsoftheTSPJuly20051LSEROVERSION.pdf (2013–04–25).
- Pataki, G. (2003). Teaching integer programming formulations using the traveling salesman problem. *SIAM Review*, 45 (1), 116–123.
- Patterson, M.C., Harmel, B. (2005). Solving the Traveling Salesman Problems using Premium Solver Platform Software. *International Journal of Management*, 22 (4), 532–541.
- Rasmussen R. (2011a). TSP in Spreadsheets — a Guided Tour. 2011. *International Review of Economic Education*, 10 (1), 94–116; <http://www.economicsnetwork.ac.uk/iree/v10n1/rasmussen.pdf> (2012–08–30).
- Rasmussen R. (2011b). TSP inspreadsheets — A fast and flexible tool. *Omega* 39 (2011) pages 51–63; <http://www.elsevier.com/locate/omega> (2013–04–04).
- Sysło M., Deo N., Kowalik J. (1999). *Algorytmy optymalizacji dyskretnej*. Warszawa: PWN.
- Węgrzyn J. (2013). Rozwiązywanie problemu komiwojażera za pomocą Metody Ewolucyjnej z Excela 2010. *Gospodarka Materialowa i Logistyka*, (12), 11–17.
- Frontline Systems Inc. (2013). *Frontline Solvers User Guide version 12.5 For Use With Excel 2003–2013*. <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/> (2013–01–29). <http://www.solver.com/analytic-solver-platform> (2013–02–15). <http://www.tsp.gatech.edu/concorde/> (2013–04–04).