Available online at www.sciencedirect.com

ScienceDirect

Procedia Economics and Finance 26 (2015) 1026 – 1034

Procedia Economics and Finance

www.elsevier.com/locate/procedia

4th World Conference on Business, Economics and Management, WCBEM

Formulations for Minimizing Tour Duration of the Traveling Salesman Problem with Time Windows

Imdat Kara[a], Tusan Derya[a]*

[a]*Baskent University, Faculty of Engineering, Department of Industrial Engineering, Baglica Campus, Ankara 06530, Turkey*

Abstract

Traveling Salesman Problem with Time Windows (TSPTW) serves as one of the most important variants of the Traveling Salesman Problem (TSP). The main objective functions expressed in the literature of the TSPTW consist of the following: (1) to minimize total distance travelled (or to minimize total travel time spent on the arcs), (2) to minimize total cost of traveling on the arcs and cost of waiting before services, or (3) to minimize total time passed from depot to depot (tour duration). When the unit cost of traveling and unit cost of waiting appear equal, then one may solve the problem just minimizing tour duration. According to our best knowledge, only one formulation with nonlinear constraints considers the aim of minimizing tour duration for symmetric case. However, for just minimizing tour duration, we haven’t seen any linear formulation for symmetric and any general formulation for asymmetric TSPTW. In this paper, for minimizing tour duration, we propose polynomial size integer linear programming formulation for symmetric and asymmetric TSPTW separately. The performances of our proposed formulations are tested on well-known benchmark instances used to minimize total travel time spent on the arcs. For symmetric TSPTW, our proposed formulation sufficiently and capably finds optimal solutions for all instances (the biggest are with 400 customers) in an extremely short time with CPLEX 12.4. For asymmetric TSPTW, we used instances generated from a real life problem which were used for minimizing total arc cost. We used these instances for minimizing tour duration and observe that, optimal solutions of the most of the instances up to 232 nodes are obtained within seconds. In addition, the proposed formulation for asymmetric case obtained 7 new best known solutions.

Keywords: Travelling Salesman Problem; Routing; Total Duration

* Tusan Derya. Tel.: +90-312-2466666; fax: +90-312-2466660.
E-mail address: tderya@baskent.edu.tr

doi:10.1016/S2212-5671(15)00926-0

1. Introduction

The Traveling Salesman Problem (TSP) lies at the heart of routing problems and serves as one of the most attractive combinatorial optimization problems. One may state the TSP simply as finding the shortest route for a salesman (traveler) who has to visit a number of cities, by starting and ending at the depot and visiting each one of the other cities exactly once. In the Traveling Salesman Problem with Time Windows (TSPTW), an important variant of TSP, the traveler must visit each customer within a time interval, named as the time windows indicating the earliest and latest times this customer will permit the start of the service.

TSPTW may arise in a distribution logistic, such as bank deliveries, postal deliveries, industrial refuse collection, school bus routing, transportation of perishable goods, job-shop scheduling, military operations, etc., where no capacity restrictions apply.

TSPTW has not received as much attention as TSP or as the Vehicle Routing Problem (VRP). For an earlier overview of TSPTW, see Desrosiers *et al*. (1988). A brief literature review of TSPTW may be found in recent papers by Lopez-Ibanez and Blum (2010) and Kara *et al*. (2013). Also, Tsitsiklis (1992) studied special cases of the TSPTW.

The first attempt to find an optimal solution to the TSPTW appeared in an article by Christofides *et al*. (1981). They developed a branch-and-bound algorithm using dynamic programming and solving instances up to 50 nodes to optimality. Dumas *et al*. (1995) extended earlier dynamic programming approaches and solved instances up to 200 nodes.

TSPTW reduces to TSP when we take time windows as $[0, \infty]$ for all nodes. Thus, TSPTW is NP-hard. Savelsbergh (1985) has proved even finding a feasible solution for the TSPTW is NP-Complete. The limitation of the exact algorithm to find an optimal solution and the complexity of the problem lead to researchers developing effective heuristics (for recent research, see for example Favaretto *et. al*. (2006), Ohlmann and Thomas (2007). More recently, some meta-heuristics approaches Lopez-Ibanez and Blum (2010), Sarhadi and Ghoseiri (2010), Silva and Urrutia (2010) have used integer programming formulations to obtain high-quality solutions.

The goal of the TSPTW may be: (1) to minimize total tour duration (completion time of the tour) (2) to minimize the sum of arcs traversal time (or total cost occurred on travelling the arcs) (3) to minimize the sum of arcs traversal time (travel cost) plus waiting time (waiting cost) due the waiting in front of the nodes.

The case where one to minimize tour duration is also called “Makespan Problem with Time Windows” (Langevien *et al*., 1993). When the unit cost of travelling and unit cost of waiting appear equal, then one may solve the problem simply by minimizing tour duration. Savelsbergh (1992) first defined the variant of the TSPTW named as the Vehicle Routing Problem with Time Windows by minimizing tour duration. He did not propose a mathematical model, but rather an edge-exchange improvement method to solve the problem. As far as we know, there is only one formulation for symmetric TSPTW proposed by Baker (1983) dealing with tour duration, but it is a nonlinear programming model. Baker (1983) developed a branch-and-bound procedure and solved instances up to 50 nodes to optimality. We could not recognize at any paper for asymmetric TSPTW that minimize tour duration. Special case of the formulation recently appeared in the literature may be used for minimizing tour duration (Kara *et al*., 2013).

The remarkable improvement in hardware and software technologies and widespread availability of commercial optimization software will allow us to solve user friendly mathematical models easily in the near future. Furthermore, increasing interest in e-commerce and internet facilities will necessitate solving routing problems more easily and rapidly by using any optimizer. The main objective of this paper is to present new formulations to minimize tour duration of the symmetric and asymmetric TSPTW useable to solve small- and moderate-sized TSPTW directly by using any optimizer. This is the main motivation of this paper. Our contributions are three folds: (1) We present new integer linear programming formulations for symmetric and asymmetric TSPTW with $O(n^2)$ constraints and $O(n^2)$ binary variables. Both formulations are useable directly by any optimizer. (2) We computationally show that, with a suitable mathematical model, we can obtain optimal solution of the problems by using any optimizer. (3) We demonstrate that, all the benchmark instances for symmetric case solved so far by a special algorithm and/or a heuristic are solved with our formulations optimally by using CPLEX 12.4. Thus, researchers must emphases to develop accurate mathematical model for the problems before beginning to construct special algorithms and/or heuristics.

No benchmark instances exist for tour duration; the instances and their optimal solutions appearing in e-libraries are for minimizing total time spent on the arcs (or distance traveled or cost of traveling on the arcs). For symmetric TSPTW, we solved all the test instances of TSPTW up to 200 nodes generated by Dumas *et al.* (1995); the same instances with large time windows generated by Gendreau *et al.* (1998); and all the instances (200-400 nodes) generated by Silva and Urrutia (2010) by CPLEX 12.4 to optimality with the proposed model. The performance of the formulation cannot be over-stated; it solves all the symmetric instances to optimality within seconds. For asymmetric TSPTW, we solved the instances generated by Ascheuer (1995).

In section 2, we propose an integer linear programming formulation for symmetric TSPTW and summarize the result of computational analysis of the proposed formulation. An integer linear programming formulation for asymmetric TSPTW and its computational analysis are presented in section 3. The conclusion and further remarks appear in section 4.

2. Formulation For Symmetric TSPTW

In this section, we first define the notations used in this paper and then identify symmetric and asymmetric cases of TSPTW, and we present integer programming formulation for symmetric TSPTW.

2.1. Notations

We define the following notations:

Index Sets: Let G = (V, A) be a complete graph where, $V = \{0, 1, 2, \ldots, n\}$ set of the nodes (vertices, cities, customers, etc.), 0 is the depot and $A = \{(i, j) \mid i \neq j;\ i, j \in V\}$ is the set of undirected arcs.

Parameters:
t_{ij}: The travel time from node i to node j.
a_i: The earliest visit (service begins) time of the node i.
b_i: The latest visit time of the node i.
$[a_i, b_i]$: Time window of the node i.
For simplicity, if a service time is necessary at a node i, this time will be included in the travel time t_{ij}'s. TSPTW is named as *symmetric* if $t_{ij} = t_{ji}$ and *asymmetric* if $t_{ij} \neq t_{ji}$.

Decision Variables:
t_i : Time passed until visiting the node i.
x_{ij}: The auxiliary binary variable, *i.e.*, $x_{ij} \in \{0,1\}$.

2.2. Baker's Formulation

As mentioned before, Baker (1983) presented the first mathematical model for symmetric TSPTW where total tour duration is minimized. Baker's model stands as the only formulation for simply minimizing completion time. With the above notations, Baker's model appears as follows:

$$\textit{Minimize}\ \ t_{n+1} - t_0 \tag{1}$$

subject to

$$t_i - t_0 \geq t_{0i}, \qquad i = 1, 2, \ldots, n \tag{2}$$

$$\left| t_i - t_j \right| \geq t_{ij}, \qquad i = 2, 3, \ldots, n,\ 1 \leq j < i \tag{3}$$

$$t_{n+1} - t_i \geq t_{i0}, \qquad i = 1, 2, \ldots, n \tag{4}$$

$$t_i \geq a_i, \qquad i = 1, 2, \ldots, n \tag{5}$$

$$t_i \leq b_i, \qquad i = 1, 2, \ldots, n \tag{6}$$

$$t_i \geq 0, \qquad i = 0, 1, \ldots, n+1 \tag{7}$$

This model contains n+1 decision variables and O(n^2) constraints and assumes a symmetric and complete distance matrix for which the triangle inequality holds. One must also assume time spent on the arc is a scalar transformation of the distance traveled, so time and distance may be used interchangeably. The validity of the above formulation has been shown by proofing a theorem given in Baker (1983). The model is nonlinear in nature, *i.e*. it cannot be used directly by a linear programming solver. Baker (1983) developed a branch-and-bound procedure and solved randomly generated instances with up to 50 nodes to optimality.

2.3. Proposed model for symmetric TSPTW

In this section, we propose an integer linear programming formulation for symmetric TSPTW minimizing tour duration.

Proposition 1: The inequalities given in the constraints (3) of the above formulation may be lineralized as:

$$t_i - t_j \geq t_{ij} - M\, y_{ij} \quad \forall i, j = 0,1,\ldots,n \qquad (8)$$

$$t_j - t_i \geq t_{ij} - M(1 - y_{ij}) \quad \forall i, j = 0,1,\ldots,n \qquad (9)$$

where $M > 0$ and a sufficiently large number and y_{ij}'s are binary variables for all arcs of A.

Proof: Since y_{ij} is a binary variable, if $y_{ij} = 0$, the inequality (8) becomes a bounding constraint, and if $y_{ij} = 1$, the inequality (9) becomes a bounding constraint. Hence, the constraints given by (3) hold in any case. □

New Formulation: We propose our formulation as follows:

$$\textit{Minimize}\ t_{n+1} - t_0 \qquad (1)$$

subject to

$$t_i - t_0 \geq t_{0i}\,, \qquad i = 1,2,\ldots,n \qquad (2)$$

$$t_{n+1} - t_i \geq t_{i0}\,, \qquad i = 1,2,\ldots,n \qquad (4)$$

$$t_i \geq a_i\,, \qquad i = 1,2,\ldots,n \qquad (5)$$

$$t_i \leq b_i\,, \qquad i = 1,2,\ldots,n \qquad (6)$$

$$t_i \geq 0\,, \qquad i = 0,1,\ldots,n+1 \qquad (7)$$

$$t_i - t_j + M\, y_{ij} \geq t_{ij}\,, \qquad \forall i, j = 0,1,\ldots,n \qquad (8)$$

$$t_j - t_i - M\, y_{ij} \geq t_{ij} - M\,, \qquad \forall i, j = 0,1,\ldots,n \qquad (9)$$

$$y_{ij} \in \{0,1\} \quad \forall i, j = 0,1,\ldots,n\ ,\ i \neq j \qquad (10)$$

$$t_0 = 0 \qquad (11)$$

where $M > 0$ and a sufficiently large number. This formulation has n+1 continuous decision variables while having $n(n-1)$ binary decision variables, and the number of constraints is $n(n-1)+5n+1$. So, the formulation is linear and has O(n^2) integer variables and O(n^2) constraints. Hence, we can use this formulation directly for solving symmetric TSPTW problems by using an optimizer.

In this new formulation, any additional restrictions may be easily adopted to the model. For example, if the node j must be visited just after i, it is sufficient to add the constraints $t_i + t_{ij} = t_j$ to the model. If j must be visited before i then we add the constraints $t_j \leq t_i + t_{ij}$ to the model.

2.4. Computational analysis of symmetric TSPTW

As mentioned before, no test instances for minimizing tour duration appear for TSPTW; the instances given on literature are all for symmetric TSPTW generated for minimizing time spent on the arcs (or distance traveled). We used these instances, and in a computational experiment, we tested the performance of the new formulation on three different sets of data (380 instances) taken from the literature (http://homepages.dcc.ufmg.br/~rfsilva/tsptw/).

These sets are as follows:

- Data set 1: 135 instances proposed and solved to optimality with a special purpose algorithm based on dynamic programming by Dumas *et al*. (1995). The Dumas instances include problems considering between 20 and 200 customers with time window widths ranging from 20 to 100 time units.
- Data set 2: 120 instances proposed by Gendreau *et al*. (1998). These instances are the same as the instances proposed by Dumas *et al*. (1995), but time window widths are relatively large.
- Data set 3: 125 instances proposed by Silva and Urrutia (2010). These instances include problems considering between 200 and 400 customers with time window widths ranging from 100 to 500 time units.

In each data set, for each number of the node, five symmetric Euclidean instances exist with the same number of customers and the same range of time windows, where customer coordinates are uniformly distributed between 0 and 50, and travel times equal distances. No travel time matrix is directly available; between customers, it is computed from x-y coordinates by using the Euclidean distance. The proposed formulation assumes symmetric, non-negative travel times, satisfying the triangle inequality. Therefore, all travel times are truncated with the floor operator and then adjusted to obey the triangle inequality. The method proposed by Ohlmann and Thomas (2007) is used to maintain the validity of the triangular inequality.

Each of the symmetric TSPTW test instances (providing the triangular inequality) was solved by the proposed formulation to optimality by using a CPLEX 12.4 solver on two Intel Pentium IV processors of 3.00 Ghz clock speed with 2.00 GB of RAM running Microsoft Windows XP operating system.

Optimal solutions for all benchmark instances proposed by Dumas *et al*. (1995) up to 200 customers are achieved within seconds. According to the test results on benchmark instances proposed by Dumas *et al*. (1995), our proposed formulation sufficiently and capably find optimal solutions for problems with up to 200 customers and narrow time windows with a short amount of CPU time. We calculated the waiting times of each given route of all Dumas *et al*. (1995) instances and the obtained tour durations. As it is expected, optimal tour duration obtained by the new formulation is always less than or equal to the calculated duration. To avoid from extra pages, we do not give all these results within paper, one can achieve them from corresponding author.

Gendreau *et al.* (1998) proposed new larger widths for Dumas *et al*. (1995) instances up to 100 nodes. They solved these instances for minimization of travel times on the arcs by a generalized insertion heuristic. We calculated the waiting times of each route of all instances and the obtained tour durations of the Gendreau *et al*. (1998) instances and solved all these instances optimally with our formulation. The obtained results are reported as averages over five instances for each set of problems. In Table 1, we present a summary of the average values. According to the test results on benchmark instances proposed by Gendreau *et al.* (1998), our proposed formulation sufficiently and capably find optimal solutions for problems with larger widths within a short amount of CPU time.

Silva and Urrutia (2010) proposed new instances up to 400 nodes and solved these instances with a general variable neighborhood search heuristic by minimizing the sum of the travel time on the path. We calculated the waiting times of each route of all instances and the obtained calculated tour durations of the Silva and Urrutia's instances and then we solved all these instances optimally with the proposed formulation. The obtained results are reported as averages over five instances for each set of problems. In Table 1, we present a summary of the average values. According to the test results on benchmark instances proposed by Silva and Urrutia (2010), our proposed formulation sufficiently and capably find optimal solutions for problems with larger widths within a short amount of CPU time.

As expected, for each group of instances, the average total tour duration obtained by the new formulation is equal to or less than the average total tour duration calculated based on the given tour. Thus, for the case when the cost of traveling on the arcs and the cost of waiting in front of the nodes must both be considered and have equal weights; the proposed formulation will give the optimal solution.

Table 1. Computational results for symmetric TSPTW benchmark instances proposed by Gendreau *et al*. (1998) and Silva and Urrutia (2010).

Data set		Gendreau *et al.* (1998)	Proposed formulation		Data set		Silva & Urrutia (2010)	Proposed formulation	
# nodes	**width**	**ACD**	**AOD**	**CPU sec.**	**# nodes**	**width**	**ACD**	**AOD**	**CPU sec.**
20	120	330,8	**319,6**	0,27	200	100	10577,0	10577,0	0,16
	140	289,6	**286,2**	0,60		200	10707,6	**10697,6**	0,24
	160	315,4	**311,4**	0,27		300	10539,6	**10527,6**	0,25
	180	341,4	**311,2**	0,61		400	10487,2	**10477,6**	0,41
	200	309,4	**281,1**	0,91		500	10435,8	**10431,6**	0,55
40	120	503,0	**470,6**	1,25	250	100	13393,6	**13390,6**	0,26
	140	483,6	**458,2**	1,23		200	13043,2	**13041,6**	0,32
	160	446,4	**426,8**	3,86		300	13545,4	**13542,4**	0,36
	180	453,6	**427,4**	4,36		400	13252,8	**13248,6**	0,98
	200	426,6	**412,0**	6,47		500	13245,2	**13220,0**	0,62
60	120	588,0	**573,8**	2,51	300	100	15920,8	**15918,8**	0,35
	140	612,4	**600,0**	3,95		200	16038,0	16038,0	0,42
	160	651,2	**619,6**	7,79		300	15585,0	**15579,8**	0,55
	180	606,8	**576,0**	14,6		400	15747,6	**15726,8**	0,71
	200	586,0	**570,2**	763,8		500	16087,4	**16047,8**	1,23
80	100	719,0	**711,2**	4,42	350	100	18587,0	**18585,0**	0,51
	120	727,6	**697,4**	16,3		200	18355,8	**18353,8**	0,57
	140	687,0	**672,8**	14,0		300	18382,4	**18362,8**	0,81
	160	680,0	**653,6**	2008		400	18350,4	**18335,2**	0,97
	180	811,4	**805,8**	6,43		500	19029,6	**19024,2**	0,87
100	100	808,6	**795,8**	16,9	400	100	20744,0	**20742,6**	0,63
	120	906,4	**895,4**	1,77		200	21488,6	**21480,2**	0,70
	140	924,6	**906,6**	13,9		300	21338,2	**21325,2**	0,89
	160	879,4	**865,0**	24,7		400	21035,2	21035,2	1,30
						500	21068,0	**21035,6**	1,90

ACD: Average of the calculated duration, AOD: Average of the optimal duration

3. Formulation For Asymmetric TSPTW

Aschuer *et al*. (2000, 2001) are studied asymmetric TSPTW in detail and present three formulation for minimizing time spent on the arcs. As we mentioned before, we could not arrive any formulation for minimizing tour duration for the TSPTW.

In this section, we consider the case where $t_{ij} \neq t_{ji}$, i.e., the asymmetric TSPTW. With the notations defined in previous section, let us define a new decision variables x_{ij} is equal to 1 if the traveler goes from node i to node j, 0 otherwise. We present an integer programming formulation for this case.

3.1. Mathematical model for asymmetric TSPTW

We first state some propositions and present valid inequalities for the formulation and then present the mathematical model. When the traveler goes from the depot to a node i, then $t_i = max\{a_i, t_{oi}\}$ must hold. To do this, by considering earliest service times of each node, we state the following proposition.

Proposition 2: The following inequalities initialize the arrival time of the first node of a tour and the time necessary to travel from origin to that node.

$$t_i - t_{0i} x_{ij} \geq 0 \quad , \quad i = 1, 2, ..., n \tag{12}$$

$$t_i \geq a_i \quad , i = 1, 2, \ldots, n \tag{13}$$

Proof: If the traveler goes from depot to the node i then $x_{oi} = 1$, so, from (12) and (13), we obtain $t_i = max\{a_i, t_{oi}\}$. If the node i is an intermediate node, then $x_{0i} = 0$ so then (12) and (13) implies $t_i \geq a_i$. □

Definition of x_{ij}'s and t_i's implies that, if x_{ij} = 1then $t_j \geq t_i + t_{ij}$ must hold. To do this, we give the following proposition.

***Proposition 3*:** The conditional statement "if $x_{ij} = 1$ then $t_j \geq t_i + t_{ij}$" may be linearized as:

$$t_i - t_j + (b_i - a_j + t_{ij})x_{ij} \le b_i - a_j \quad , \quad i \ne j;\ i, j = 1, 2, ..., n \tag{14}$$

Proof: Let $x_{ij} = 1$ for a given arc (i, j). If we substitute "1" instead of x_{ij} in the above inequalities, we get $t_j \ge t_i + t_{ij}$. If the arc (i, j) is not on the tour, *i.e.*, if $x_{ij} = 0$ then we get $t_i - t_j \le b_i - a_j$ which is a valid inequality for TSTW since $max\{t_i - t_j\} = max\{t_i\} - min\{t_j\} \le b_i - a_j$. □

The inequalities given in (14) guarantee that arrival times to the customer on a tour constitute an increasing step function, which means that these inequalities prohibit to form any illegal subtour, *i.e.*, they are the subtour elimination constraints of our formulation.

In accordance with the explanations and derived constraints given above and also adding nonnegativity and binary restrictions, we state new integer programming formulation for asymmetric TSPTW as follow:

$$Minimize\ t_{n+1} \tag{1}$$

subject to

$$t_i - t_{0i}x_{0i} \ge 0 \quad , \quad i = 1, 2, ..., n \tag{12}$$

$$t_i \ge a_i \quad , \quad i = 1, 2, ..., n \tag{13}$$

$$t_i - t_j + (b_i - a_j + t_{ij})x_{ij} \le b_i - a_j \quad , \quad i \ne j;\ i, j = 1, 2, ..., n \tag{14}$$

$$\sum_{i=0}^{n} x_{ij} = 1 \quad , \quad j = 1, 2, ..., n \tag{15}$$

$$\sum_{j=0}^{n} x_{ij} = 1 \quad , \quad i = 1, 2, ..., n \tag{16}$$

$$t_i \le b_i \quad , \quad i = 1, 2, ..., n \tag{17}$$

$$t_i + t_{i0} \le t_{n+1} \quad , \quad i = 1, 2, ..., n \tag{18}$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \tag{19}$$

In this formulation, we have $2n^2 + 6n + 2$ constraints, $n^2 - n$ binary variables and $n+1$continuous variables. So proposed formulation has $O(n^2)$ binary variables and $O(n^2)$ constraints.

The objective function minimize the tour duration. The constraints given in (12), (13) and (14) initialize t_i's and constitute an increasing step function of t_i's, so then no illegal tour will be formed. Constraints (15) and (16) are the asignment constraints. Time window constraints are given in (13) and (17). Constraints given in (18) guarantee that the tour duration is less than or equal to visit time plus the time necessary for return to the depot for all nodes. Nonnegativity and binary restrictions are given by the constraints (13) and (19).

This formulation may be easily rewritten for the case when we have multiple travellers. In that case, if there are m travellers, it is sufficient to rewrite the constraints given in (17) and (18) as:

$$\sum_{i=0}^{n} x_{ij} = 1 \quad , \quad j = 1, 2, ..., n \tag{17a}$$

$$\sum_{j=0}^{n} x_{ij} = 1 \quad , \quad i = 1, 2, ..., n \tag{18a}$$

$$\sum_{i=1}^{n} x_{i0} = m \tag{17b}$$

$$\sum_{j=1}^{n} x_{0j} = m \tag{18b}$$

If there are additional restrictions for the order of the visiting the node, as demonstrated for the symmetric case, they may be easily inserted as new constraints to this model also.

3.2. Computational analysis of asymmetric TSPTW

A computational study on the performance of the proposed formulation for asymmetric TSPTW was conducted using 30 asymmetric TSPTW problems introduced by Ascheuer (1995). These are real-world instances derived from an industry project with the aim to minimize the unloaded travel time of a stacker crane within an automated storage system. These asymmetric TSPTW instances had customer sizes varying 11 to 232. The data set can be downloaded from the "homepage for Benchmark Instances for the TSPTW" http://iridia.ulb.ac.be/~manuel/tsptw-instances.

For the solution of the linear programming models the LP–solver CPLEX 12.4 was used. All computational results were obtained on two Intel Pentium IV processors of 3.00 Ghz clock speed with 2.00 GB of RAM under Microsoft Windows XP operating system. For all runs we allowed a maximum CPU–time of 2 hours.

In Table 2, the optimal solutions for all of the asymmetric TSPTW instances obtained by the proposed formulation are compared with the best known solutions reported in Lopez-Ibanez *et. al* (2013) in terms of the total tour duration.

Thirty of all asymmetric TSPTW instances are solved to optimality in the given time limit of 2 hours. For all other instances the obtained feasible solution values found after 2 hours of computing time equal or are close to the value of the best feasible solution known so far. The proposed formulation obtained 7 new best known solutions and 21 solutions that equaled previously best known solutions.

Table 2. Computational results for asymmetric TSPTW benchmark instances proposed by Ascheuer (1995).

Data set			**Proposed formulation**		**Data set**			**Proposed formulation**	
Problem	***n***	**Best-known**	**OTD**	**CPU sec.**	**Problem**	***n***	**Best-known**	**OTD**	**CPU sec.**
rbg010a	11	-	3840	0.00	rbg041a	42	3793	3793	0.72
rbg016a	17	-	2596	0.06	rbg050a	51	-	12050	0.75
rbg016b	17	2094	2094	0.56	rbg055a	56	6929	6929	3.99
rbg017.2	16	2351	2351	0.05	rbg067a	68	10331	10331	2.76
rbg017	16	-	2351	0.00	rbg086a	87	16899	16899	5.77
rbg019a	20	2694	2694	0.00	rbg092a	93	-	12501	2.25
rbg019b	20	-	3840	0.02	rbg125a	126	14214	14214	48.4
rbg019d	20	3479	3479	0.26	rbg132.2	131	18524	18524	105
rbg031a	32	3498	3498	0.03	rbg132	131	18524	18524	13.4
rbg033a	34	3757	3757	1.36	rbg152.3	151	-	17455	215
rbg034a	35	3314	3314	0.53	rbg152	151	17455	17455	0.33
rbg035a.2	36	3325	3325	2.79	rbg193.2	192	21401	21401	700
rbg035a	36	3388	3388	10.4	rbg193	192	21401	21401	0.79
rbg038a	39	5699	5699	4.57	rbg201a	202	21380	21380	496
rbg040a	41	5679	5679	0.50	rbg233	232	26143	26143	3495

OTD: Optimal tour duration

4. Conclusion and Further Direction

In this study, we address the symmetric and asymmetric travelling salesman problem with time windows, where the objective function is to minimize total time passed from depot to depot. When the unit cost of travelling and the unit cost of waiting appear equal, then one may solve the problem just minimizing tour duration. For this case, two polynomial size integer linear programming formulation having $O(n^2)$ binary variables and $O(n^2)$ constraints are developed to solve the symmetric and asymmetric TSPTW instances optimally by a standard optimization solver.

Though the TSPTW is NP-hard, large-scale problems are quite hard to solve optimally. In the case of minimizing tour duration for symmetric TSPTW, our proposed model can achieve optimal solutions in an extremely short time. All TSPTW benchmark instances involving large numbers of customers (with up to 400 nodes) are solved to optimality with the proposed formulation by using CPLEX 12.4 solver within a few seconds. For asymmetric TSPTW, computational results demonstrate our proposed formulation sufficiently and capably finds optimal solutions for all instances with up to 232 customers in an extremely short time with CPLEX 12.4. So, these formulations would be a miles stone for solving routing problems to optimality by an optimizer and researchers must emphases to develop accurate mathematical model for the problems before beginning to construct special algorithms and/or heuristics.

These formulations may be extended for Vehicle Routing Problems with Time Windows (VRPTW).

References

Ascheuer, N., (1995). Hamiltonian Path Problems in the On-line Optimization of Flexible Manufacturing Systems. PhD thesis. Technische Universitat, Berlin, Germany.

Ascheuer, N., Fischetti, M., Grotschel, M., (2000). A polyhedral study of the asymmetric travelling salesman problem with time windows. *Networks* 36 (2), 69 – 79.

Ascheuer, N., Fischetti, M., Grotschel, M., (2001). Solving the asymmetric travelling salesman problem with time windows by branch-and-cut. Mathematical Programming Series A. 90, 475 – 506.

Baker, E.K., 1983. An exact algorithm for the time-constrained traveling salesman problem. Operations Research 31(5), 938 – 945.

Cheng, C-B., Mao, C-P., (2007). A modified ant colony system for solving the traveling salesman problem with time windows. Mathematical and computer modeling 46, 1225 – 1235.

Christofides, N., Mingozzi, A., Toth, P., (1981). State-space relaxation procedures for the computation of bounds to routing problems. Networks 11, 145 – 164.

Desrosiers, J., Sauve, M., Soumis, F., (1988). Lagrangian relaxation methods for solving the minimum fleet size multiple traveling salesman problem with time windows. *Management Science* 34(8), 1005 – 1022.

Dumas, Y., Desrosiers, J., Gelinas, E., Solomon, M.M., (1995). An optimal algorithm for the traveling salesman problem with time windows. Operations Research 43 (2), 367 – 371.

Favaretto, D., Moretti, E., Pellegrini, P., 2006. An ant colony system approach for variants of the traveling salesman problem with time windows. Journal of Information and Optimization Sciences 27 (1), 35 – 54.

Gendreau, M., Hertz, A., Laporte, G., Stan, M., (1998). A generalized insertion heuristic for the traveling salesman problem with time windows. Operations Research 46 (3), 330 – 335.

Kara, I., Koc, G.N., Altıparmak, F., Dengiz, B., (2013). New Integer Linear Programming Formulation for Traveling Salesman Problem with Time Windows: Minimizing Tour Duration with Waiting Times, Optimization. 62 (10), 1309 – 1319.

Langevin, A., Desrochers, M., Desrosiers, J., Gelinas, S., Soumis, F., (1993). A two-commodity flow formulation for the traveling salesman and the makespan problems with time windows. *Networks* 23 (7), 631 – 640.

Lopez-Ibanez, M., Blum, C., (2010). Beam-ACO for the traveling salesman problem with time windows. Computers and Operations Research 37, 1570 – 1583.

Lopez-Ibanez, M., Blum, C., Ohlmann, J.W., Thomas, B.W., (2013). The traveling salesman problem with time windows: Adapting algorithms from travel-time to makespan optimization. *Applied Soft Computing* 13, 3806 – 3815.

Ohlmann, J.W., Thomas, B.W., 2007. A compressed-annealing heuristic for the traveling salesman problem with time windows. *INFORMS Journal on Computing* 19 (1), 80 – 90.

Sarhadi, H., Ghoseiri, K., (2010). An ant colony system approach for fuzzy traveling salesman problem with time windows. *International Journal of Advanced Manufacturing Technology* 50 (9), 1203 – 1215.

Savelsbergh, M.W.P., (1985). Local search in routing problem with time windows. *Annals of Operations Research* 4, 285 – 308.

Savelsberg, M.W.P., (1992). The Vehicle Routing with time windows: Minimizing route duration. *ORSA Journal of Computing* 4 (2), 146 – 154.

Silva, R.F., Urrutia, S., (2010). A general VNS heuristic for the traveling salesman problem with time windows. *Discrete Optimization* 7, 203 – 211.

Tsitsiklis, J.N., 1992. Special Cases of the Traveling salesman and Repairman Problems with time windows. *Networks* 22, 263 – 282.