Submitted to *INFORMS Journal on Computing*
manuscript (JOC-2016-03-OA-049)

Integer Programming Formulations and Benders Decomposition for the Maximum Induced Matching Problem

Betül Ahat, Tınaz Ekim, Z. Caner Taşkın
Department of Industrial Engineering, Boğaziçi University, {betul.ahat@boun.edu.tr, tinaz.ekim@boun.edu.tr, caner.taskin@boun.edu.tr}

We investigate the Maximum Induced Matching problem (MIM), which is the problem of finding an induced matching having the largest cardinality on an undirected graph. The problem is known to be NP-hard for general graphs. We first propose a vertex-based integer programming formulation for MIM, which is more compact compared to an edge-based formulation found in the literature. We also introduce the Maximum Weight Induced Matching problem (MWIM), which generalizes MIM so that vertices and edges have weights. We adapt the edge-based formulation to MWIM, and propose a quadratic programming formulation of MWIM based on our vertex-based formulation. We then linearize our quadratic programming formulation, and devise a Benders decomposition algorithm that exploits a special structure of the linearized formulation. We also propose valid inequalities and formulation tightening procedures to improve the efficiency of our approach. Our computational tests on a large suite of randomly generated graphs show that our vertex-based formulation and decomposition approach significantly improve the solvability of MIM and MWIM, especially on dense graphs.

Key words: Integer programming, Benders decomposition, Maximum induced matching, Distance-2 matching, Strong matching
History:

1. Introduction

For a given graph $G=(V,E)$, an induced matching is a matching such that no two edges in the matching are joined by an edge of G. In other words, an induced matching is a matching that forms an induced subgraph. In Figure 1, the edge set $\{\{1,5\},\{6,7\},\{4,8\}\}$ is an induced matching since no two edges in the set are joined by another edge of G. Each induced matching in G corresponds to an independent set of vertices in the square of the

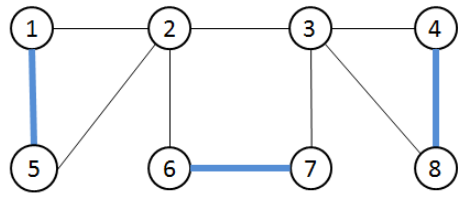

Figure 1 **An example of induced matching**

line graph of G, denoted by $[L(G)]^2$. Similarly, each independent set of vertices in $[L(G)]^2$ corresponds to an induced matching in G (Cameron 1989).

The size (or cardinality) of an induced matching is defined as the number of edges in the induced matching. The Maximum Induced Matching problem (MIM) aims to find an induced matching having the largest cardinality. MIM is also related with strong matching (Golumbic and Laskar 1993), maximum distance-2 matching (Balakrishnan et al. 2004) and risk-free marriage problems (Stockmeyer and Vazirani 1982). It is used as a sub-task in finding a strong edge coloring, an edge-coloring in which each color class is an induced matching (Duckworth et al. 2002). In practice, induced matchings are heavily used in communication industry to obtain secure communication channels (Christou and Vassilaras 2013). It is also applied to model the problem of determining the maximum number of concurrent transmissions at media access (MAC) layer in an ad-hoc wireless network (Balakrishnan et al. 2004).

MIM is NP-hard for general graphs (Cameron 1989, Stockmeyer and Vazirani 1982). However, it is polynomial-time solvable for some special graphs such as trees (Fricke and Laskar 1992), chordal and interval graphs (Cameron 1989), weakly chordal graphs (Cameron et al. 2003), and circular arc graphs (Golumbic and Laskar 1993). Various approximation algorithms for solving MIM in general graphs have been proposed (see, e.g., Duckworth et al. (2005), Orlovich et al. (2008)). Moser and Sikdar (2009), Dabrowski et al. (2013), Chang et al. (2015) investigate the parametrized complexity of the problem for some restricted graph classes. Vassilaras and Christou (2011) develop an integer programming-based algorithm to solve the problem in unit disk graphs and compare their exact solutions to solutions obtained by a greedy algorithm by Balakrishnan et al. (2004). We review their edge-based integer programming formulation in Section 2. Christou and Vassilaras (2013) show that this formulation can be equivalently converted into maximum 2-packing problem on the line graph of G. In their formulation, the number of decision variables depends on the number of edges, which is $O(|V|^2)$ for dense graphs.

In this paper, we first give a vertex-based binary integer programming formulation for MIM, in which the number of decision variables and constraints depends on the number of vertices in the graph ($O(|V|)$). We then generalize the problem to graphs in which each vertex has a weight, and the objective is to maximize the total weight of saturated vertices in an induced matching. We call the resulting problem Maximum Vertex-Weighted Induced Matching problem (MVWIM). Similarly, in the Maximum Edge-Weighted Induced Matching problem (MEWIM), we assume that edges have weights and the total edge weight in an induced matching is to be maximized. We adapt Vassilaras and Christou (2011)'s edge-based formulation and our vertex-based formulation for MIM to solve MVWIM and MEWIM.

In the Maximum Weight Induced Matching problem (MWIM), we consider graphs having both edge and vertex weights. The aim is to maximize the total weight of saturated vertices and selected edges in an induced matching. We formulate MWIM as a quadratic programming problem. We then linearize our formulation, and develop a Benders decomposition approach to solve the problem to optimality. Our decomposition strategy explicitly considers vertex weights and seeks an optimal induced matching by solving a master problem, and calculates the corresponding edge weight by solving a subproblem. We investigate the subproblem's structure and devise an efficient algorithm to generate Benders cuts.

The remainder of this paper is organized as follows. In Section 2, we introduce Vassilaras and Christou (2011)'s edge-based formulation and our vertex-based formulation for MIM. We then adapt these formulations to solve MVWIM, MEWIM and MWIM. Section 3 describes our Benders decomposition approach for solving MWIM. We develop valid inequalities and formulation tightening procedures to improve the efficiency of our decomposition algorithm in Section 4. We compare the efficacy of these algorithms in Section 5 on a suite of randomly generated graphs. Finally, we conclude our paper in Section 6.

2. Problem Formulations

2.1. Maximum Induced Matching Problem (MIM)

Vassilaras and Christou (2011) introduced a binary integer programming formulation for MIM. Let $G = (V, E)$ be an undirected graph with vertex set V and edge set E. In Vassilaras and Christou (2011)'s formulation, each edge is represented by a binary decision variable y_{ij}, which takes value 1 if edge $\{i, j\} \in E$ is selected in a maximum induced matching, and

0 otherwise. Let $N_{ij} \subseteq E$ be the set of edges that are adjacent to edge $\{i,j\}$. Then, their formulation for MIM is as follows:

$$\textbf{(VC2011):} \quad max \quad \sum_{\{i,j\}} \in E y_{ij} \tag{1a}$$

$$s.t. \quad y_{ij} + \sum_{(k,l) \in N_{ij}} y_{kl} \leq 1 \quad \forall \ \{i,j\} \in E \tag{1b}$$

$$y_{ij} \in \{0,1\} \quad \forall \{i,j\} \in E \tag{1c}$$

The objective function (1a) maximizes the number of selected edges. Constraints (1b) enforce the condition that if an edge $\{i,j\} \in E$ is selected ($y_{ij} = 1$), none of its adjacent edges can be selected (all $y_{kl} = 0$ for $\{k,l\} \in N_{ij}$), and if it is not selected ($y_{ij} = 0$), at most one of its adjacent edges can be selected. These constraints guarantee that the selection is an induced matching in G. The number of binary variables in this formulation is proportional to the number of edges in the graph, which is $O(|V|^2)$ for dense graphs. Therefore, as we demonstrate in Section 5, this formulation cannot be solved efficiently for graphs having a large number of edges.

Our key observation s that the set of saturated vertices uniquely identifies an induced matching. Therefore, instead of formulating MIM based on edges, one can focus on the vertices and decide which vertices will be saturated in an optimal induced matching. Selected edges in the induced matching can easily be deduced from the set of saturated vertices.

In our vertex-based formulation, a binary variable x_i takes value 1 if the corresponding vertex $i \in V$ is saturated by the induced matching, and 0 otherwise. Let $N(i) \subset V$ represent the set of vertices that are adjacent to vertex $i \in V$. Then, MIM can be formulated as:

$$\textbf{(MIM):} \quad max \quad \sum_{i \in V} x_i/2 \tag{2a}$$

$$s.t. \quad x_i \leq \sum_{j \in N(i)} x_j \quad \forall i \in V \tag{2b}$$

$$\sum_{j \in N(i)} x_j \leq (|N(i)| - 1)(1 - x_i) + 1 \quad \forall i \in V \tag{2c}$$

$$x_i \in \{0,1\} \quad \forall i \in V. \tag{2d}$$

The objective function (2a) maximizes the number of saturated vertices (divided by 2 to obtain the number of edges in the induced matching). Constraints (2b) and (2c) ensure that if vertex $i \in V$ is saturated ($x_i = 1$), exactly one of its adjacent vertices is also saturated. Otherwise ($x_i = 0$), (2b) and (2c) are redundant. Note that the number of binary variables and the number of constraints depend on the number of vertices in the graph ($O(|V|)$), and are not affected by the density of the graph in our formulation.

2.2. Maximum Vertex-Weighted Induced Matching Problem (MVWIM)

In this section we extend previous formulations for MIM to solve the maximum vertex-weighted induced matching problem (MVWIM), in which each vertex $i \in V$ has a weight c_i and the objective is to maximize the total weight of saturated vertices in an induced matching. Note that MIM is a special case of MVWIM where $c_i = 1/2$ for all $i \in V$. Our vertex-based model (MIM) can be reformulated to solve MVWIM simply by changing the objective function (2a) as follows:

$$max \quad \sum_{i \in V} c_i x_i. \tag{3}$$

Vassilaras and Christou's formulation (VC2011) has no decision variable representing the saturated vertices. To reformulate their model to solve MVWIM, we define $w_{ij} = c_i + c_j$ for all $\{i, j\} \in E$, where w_{ij} represents weight of edge $\{i, j\}$. With this transformation, an instance of MVWIM becomes an instance of MEWIM, and an optimal solution can be found using the models given in the following section.

2.3. Maximum Edge-Weighted Induced Matching Problem (MEWIM)

In the maximum edge-weighted induced matching problem (MEWIM), we assume that each edge $\{i, j\} \in E$ has a weight w_{ij} and the sum of edge weights in an induced matching is maximized. Note that MIM is a special case of MEWIM where $w_{ij} = 1$ for all $\{i, j\} \in E$. Since Vassilaras and Christou (2011) define a decision variable to represent an edge, we can reformulate their model (VC2011) by replacing the objective function (1a) with (4). The constraint set remains the same.

$$max \quad \sum_{\{i,j\} \in E} w_{ij} y_{ij} \tag{4}$$

In our formulation (MIM), there is no decision variable representing the selected edges. To reformulate our model to solve MEWIM instances, we observe that an edge $\{i, j\} \in E$ is selected if and only if both of its end-vertices are saturated ($x_i = x_j = 1$). We replace the objective function (2a) with (5), which maximizes the total weight of selected edges. With this transformation, our formulation becomes a quadratic programming problem.

$$max \quad \sum_{\{i,j\}\in E} w_{ij}x_ix_j \tag{5}$$

2.4. Maximum Weight Induced Matching Problem (MWIM)

In the Maximum Weight Induced Matching problem (MWIM) we consider graphs having weights on both edges and vertices. As before, let w_{ij} represent the weight of edge $\{i,j\} \in E$, and c_i represent the weight of vertex $i \in V$. MWIM seeks an induced matching that maximizes the total weight of selected edges and saturated vertices. We can reformulate Vassilaras and Christou's model (VC2011) as:

$$\textbf{(VC2011 MWIM):} \quad max \quad \sum_{\{i,j\}\in E} (w_{ij}+c_i+c_j)\; y_{ij} \tag{6a}$$

$$s.t. \quad y_{ij} + \sum_{(k,l)\in N_{ij}} y_{kl} \le 1 \quad \forall \{i,j\} \in E \tag{6b}$$

$$y_{ij} \in \{0,1\} \quad \forall \{i,j\} \in E \tag{6c}$$

Our vertex-based formulation for MWIM can be obtained by modifying the objective function of (MIM):

$$\textbf{(MWIM QP):} \quad max \quad \sum_{\{i,j\}\in E} w_{ij}x_ix_j + \sum_{i\in V} c_ix_i \tag{7a}$$

$$s.t. \quad x_i \le \sum_{j\in N(i)} x_j \quad \forall i \in V \tag{7b}$$

$$\sum_{j\in N(i)} x_j \le (|N(i)|-1)(1-x_i)+1 \quad \forall i \in V \tag{7c}$$

$$x_i \in \{0,1\} \quad \forall i \in V \tag{7d}$$

Note that (MWIM QP) is a quadratic programming problem. However, we can linearize it by defining a new binary decision variable y_{ij}, which takes value 1 if edge $\{i,j\} \in E$ is selected in an optimal solution, and 0 otherwise. Our linearized formulation for MWIM is as follows:

$$\textbf{(MWIM):} \quad max \quad \sum_{\{i,j\}\in E} w_{ij}y_{ij} + \sum_{i\in V} c_ix_i \tag{8a}$$

$$s.t. \quad x_i \leq \sum_{j \in N(i)} x_j \quad \forall\, i \in V \tag{8b}$$

$$\sum_{j \in N(i)} x_j \leq (|N(i)| - 1)(1 - x_i) + 1 \quad \forall\, i \in V \tag{8c}$$

$$y_{ij} \leq x_i \quad \forall\, \{i,j\} \in E \tag{8d}$$

$$y_{ij} \leq x_j \quad \forall\, \{i,j\} \in E \tag{8e}$$

$$y_{ij} \geq x_i + x_j - 1 \quad \forall\, \{i,j\} \in E \tag{8f}$$

$$x_i \in \{0,1\} \quad \forall\, i \in V \tag{8g}$$

$$y_{ij} \in \{0,1\} \quad \forall\, \{i,j\} \in E \tag{8h}$$

The objective function (8a) maximizes the sum of total edge and vertex weights. Constraints (8b) and (8c) guarantee that the selection is an induced matching in G. Constraints (8d)–(8f) ensure that an edge $\{i,j\} \in E$ is selected in an optimal solution ($y_{ij} = 1$) if and only if both of its end-vertices are saturated by the matching ($x_i = x_j = 1$). We observe that our formulation is valid even if w_{ij} or c_i values are negative. We also observe that y-variables can be relaxed as $y \geq 0$ since (8d)–(8g) ensure that y-variables will take on binary values in an optimal solution.

3. Benders Decomposition for MWIM

Note that model (MWIM) contains $|V| + |E|$ binary decision variables. Even though the y-variables can be relaxed as $y \geq 0$, the number of constraints is $O(|V| + |E|)$. Thus, it may be computationally difficult to solve (MWIM) for large or dense graphs. In this section, we focus on deriving a Benders decomposition algorithm to solve (MWIM).

Our decomposition approach first seeks a feasible induced matching using a master problem, which contains the x-variables. We then check whether the induced matching found by the master problem provides an optimal selection of edges using a subproblem, which only contains the y-variables. Let us first reformulate the problem in terms of x-variables and an additional continuous variable t, which predicts the maximum edge weight that can be obtained with the selection of x-variables, as follows:

$$(MP)\textbf{:} \quad max \quad \sum_{i \in V} c_i x_i + t \tag{9a}$$

$$s.t. \quad x_i \leq \sum_{j \in N(i)} x_j \quad \forall\, i \in V \tag{9b}$$

$$\sum_{j \in N(i)} x_j \leq (|N(i)|-1)(1-x_i)+1 \quad \forall\, i \in V \tag{9c}$$

$$t \leq UB \tag{9d}$$

$$x_i \in \{0,1\} \quad \forall\, i \in V, \tag{9e}$$

where UB is an upper bound on the weight of any induced matching. This formulation contains significantly fewer decision variables and constraints than the original model (MWIM), which is advantageous in terms of computational effort.

In (MP), constraints (9b) and (9c) provide a feasible induced matching. Since the decision variable t predicts the maximum total edge weight in the induced matching, a naive value for UB can be calculated by summing up all positive edge weights. We discuss ways of calculating tighter values for UB, and propose some valid inequalities in Section 4.1.

After solving (MP) and finding a feasible vertex selection $\hat{x}$, the corresponding edge selection and the total edge weights of these edges can be found by solving the following subproblem $(SP(\hat{x}))$:

$$\textbf{(}\boldsymbol{SP(\hat{x})}\textbf{):} \quad max \sum_{\{i,j\} \in E} w_{ij} y_{ij} \tag{10a}$$

$$s.t \quad y_{ij} \leq \hat{x}_i \quad \forall\, \{i,j\} \in E \qquad (\alpha_{ij}) \tag{10b}$$

$$y_{ij} \leq \hat{x}_j \quad \forall\, \{i,j\} \in E \qquad (\beta_{ij}) \tag{10c}$$

$$y_{ij} \geq \hat{x}_i + \hat{x}_j - 1 \quad \forall\, \{i,j\} \in E \qquad (\gamma_{ij}) \tag{10d}$$

$$y_{ij} \in \{0,1\} \quad \forall\, \{i,j\} \in E \tag{10e}$$

In this formulation, y-variables can be relaxed as $y \geq 0$, which makes the subproblem a linear programming (LP) problem. Furthermore, we do not need to solve it as an LP since the solution can be obtained trivially by inspection for any given $\hat{x}$-vector. Note that $SP(\hat{x})$ is always feasible and its feasible region is bounded. Hence, the dual of $SP(\hat{x})$ is always feasible and bounded. Let α_{ij}, β_{ij} and γ_{ij} be dual multipliers associated with the constraints (10b), (10c) and (10d) in $SP(\hat{x})$, respectively. The dual formulation of the subproblem for a given $\hat{x}$ is given below:

$$\textbf{(}\boldsymbol{DSP(\hat{x})}\textbf{):} \quad min \sum_{\{i,j\} \in E} (\alpha_{ij}\hat{x}_i + \beta_{ij}\hat{x}_j + \gamma_{ij}(\hat{x}_i + \hat{x}_j - 1)) \tag{11a}$$

$$s.t \quad \alpha_{ij} + \beta_{ij} + \gamma_{ij} \geq w_{ij} \quad \forall\, \{i,j\} \in E \tag{11b}$$

$$\alpha_{ij} \geq 0 \quad \forall \{i,j\} \in E \tag{11c}$$

$$\beta_{ij} \geq 0 \quad \forall \{i,j\} \in E \tag{11d}$$

$$\gamma_{ij} \leq 0 \quad \forall \{i,j\} \in E \tag{11e}$$

Note that the feasible region of $DSP(\hat{x})$ does not depend on the value of $\hat{x}$, since $\hat{x}$ only appears in the objective function. We observe that $DSP(\hat{x})$ is separable over E. Let $0 \leq \theta_{ij} \leq 1$ be a constant that represents the proportion of w_{ij} that is allocated to α_{ij} in an optimal solution. Then, an optimal solution for $SP(\hat{x})$ and $DSP(\hat{x})$ can be found using Algorithm 1 for a given $\hat{x}$-vector:

Algorithm 1 Solution of $SP(\hat{x})$ and $DSP(\hat{x})$

Require: A graph $G = (V, E)$, a binary vector $\hat{x}$ of size $|V|$ and $0 \leq \theta_{ij} \leq 1$ for all $\{i,j\}$

Ensure: A solution of $SP(\hat{x})$ and $DSP(\hat{x})$

```
For each edge {i,j} ∈ E,
if  w_ij ≥ 0  then
    if  x̂_i < x̂_j  then
        set y_ij = x̂_i,  α_ij = w_ij,  β_ij = 0,  γ_ij = 0
    else if  x̂_i > x̂_j  then
        set y_ij = x̂_j,  α_ij = 0,  β_ij = w_ij,  γ_ij = 0
    else
        set y_ij = x̂_i, α_ij = θ_ij w_ij , β_ij = (1 − θ_ij) w_ij and γ_ij = 0
    end if
else
    if x̂_i + x̂_j < 1 then
        set y_ij = 0, α_ij = β_ij = γ_ij = 0
    else
        set y_ij = x̂_i + x̂_j − 1, α_ij = β_ij = 0 and γ_ij = w_ij
    end if
end if
```

Proposition 1. *A solution of $SP(\hat{x})$ and $DSP(\hat{x})$ obtained using Algorithm 1 satisfies linear programming optimality conditions.*

Proof Note that for each (α,β,γ) solution of Algorithm 1, $\alpha_{ij}+\beta_{ij}+\gamma_{ij}\geq w_{ij}$ and they satisfy non-negativity/non-positivity conditions. Therefore, given an $\hat{x}$-vector, Algorithm 1 always produces dual feasible (α,β,γ) solutions.

For an edge $\{i,j\}\in E$, assume $w_{ij}\geq 0$. If $\hat{x}_i\leq \hat{x}_j$, $y_{ij}=\hat{x}_i$ satisfies (10b)–(10d), hence it is primal feasible. In this case, both $SP(\hat{x})$ and $DSP(\hat{x})$ have the same objective function value, namely $w_{ij}\hat{x}_i$. If $\hat{x}_i>\hat{x}_j$, then $y_{ij}=\hat{x}_j$ is a primal feasible solution and objective function values of $SP(\hat{x})$ and $DSP(\hat{x})$ are both equal to $w_{ij}\hat{x}_j$.

Assume $w_{ij}<0$. If $\hat{x}_i+\hat{x}_j<1$, $y_{ij}=0$ satisfies constraints (10b)–(10d). In this case, both $SP(\hat{x})$ and $DSP(\hat{x})$ have objective function value equal to 0. Else, $y_{ij}=\hat{x}_i+\hat{x}_j-1$ satisfies primal feasibility conditions, and objective function values of $SP(\hat{x})$ and $DSP(\hat{x})$ are both $w_{ij}(\hat{x}_i+\hat{x}_j-1)$. Thus, all linear programming optimality conditions are satisfied. Q.E.D.

Our Benders decomposition strategy first solves (MP) to optimality, yielding a feasible $(\hat{x},\hat{t})$. We then solve $DSP(\hat{x})$ using Algorithm 1 and calculate the total edge weight, which we denote by t^*. If $\hat{t}=t^*$, then $\hat{x}$ corresponds to an optimal induced matching. On the other hand, if $\hat{t}>t^*$, we need to add a constraint to (MP). Let $(\hat{\alpha},\hat{\beta},\hat{\gamma})$ be optimal dual multipliers obtained by solving $DSP(\hat{x})$. We add the following constraint to (MP), and re-solve in the next iteration to obtain a new candidate optimal solution:

$$t\leq \sum_{\{i,j\}\in E}(\hat{\alpha}_{ij}x_i+\hat{\beta}_{ij}x_j+\hat{\gamma}_{ij}(x_i+x_j-1)) \tag{12}$$

Constraints (12) are called "Benders optimality cuts" since they are based on optimality conditions of the subproblem. Since $DSP(\hat{x})$ can have infinitely many solutions, the selection of θ_{ij} parameters results in different cuts. Proposition 2 investigates the relationship between θ_{ij} and quality of the corresponding cut.

PROPOSITION 2. *For any value of θ_{ij} and the corresponding $(\hat{\alpha},\hat{\beta},\hat{\gamma})$ vector obtained by Algorithm 1, violation of optimality cut* (12) *is the same.*

Proof For any optimality cut (12), the violation is

$$t-\sum_{\{i,j\}\in E}(\hat{\alpha}_{ij}\hat{x}_i+\hat{\beta}_{ij}\hat{x}_j+\hat{\gamma}_{ij}(\hat{x}_i+\hat{x}_j-1))=t-t^*, \tag{13}$$

where t^* is the current optimal objective function value for $DSP(\hat{x})$. Thus, all optimality cuts generated in this way have the same violation regardless of the value of $(\hat{\alpha},\hat{\beta},\hat{\gamma})$. Q.E.D.

In (MP), we can create a decision variable t_i for each vertex $i \in V$ instead of a single decision variable t. These t_i-variables are used to predict the maximum edge weight that can be obtained if we saturate vertex i. To incorporate this change, we simply replace the objective function (9a) and constraint (9d) with the following objective function (14) and constraints (15), respectively:

$$max \quad \sum_{i \in V} (t_i \ / \ 2 \ + \ c_i x_i) \tag{14}$$

$$t_i \leq UB_i, \tag{15}$$

where UB_i is an upper bound on the value of t_i. The maximum edge weight emanating from vertex i constitutes a valid upper bound for t_i. We discuss other valid inequalities for t_i-variables in Section 4.1. Our Benders decomposition procedure also needs to be adjusted accordingly. Optimality cut (12) can also be decomposed so that multiple cuts can be added in each iteration (one for each vertex $i \in V$):

$$t_i \leq \sum_{j \in N(i)} (\hat{\alpha}_{ij} x_i + \hat{\beta}_{ij} x_j + \hat{\gamma}_{ij}(x_i + x_j - 1)) \quad \forall i \in V \tag{16}$$

Our solution procedure for MWIM using Benders decomposition is summarized in Algorithm 2.

4. Algorithmic and Modeling Improvements

4.1. Valid Inequalities

The initial optimal solution to the relaxation of (MP), in which none of the Benders optimality cuts (12) have yet been added, will set the t-variable equal its upper bound UB. By tightening the value of UB and adding valid inequalities that relate x- and t-variables, we can have smaller upper bound values for the objective function during initial iterations of Algorithm 2, which can result in faster convergence.

To obtain a tighter upper bound for the t-variable, we will use the following observation: for a graph $G = (V, E)$, a matching can contain at most $\lfloor |V|/2 \rfloor$ edges. This is also valid for any induced matching since an induced matching is also a matching in G. Based on this observation, we can find an upper bound on the value of the t-variable. Since it is used to predict the maximum edge weight of an induced matching in G, a numerical bound, denoted by UB_1, can be found by sorting all w_{ij} values and summing up the greatest

Algorithm 2 MWIM Benders Decomposition

Require: A graph $G=(V,E)$ having edge weights w_{ij} and vertex weights c_i

Ensure: A maximum weight induced matching

1: Set $LB=-\infty$ and $UB=\infty$

2: Solve (MP). Let $(\hat{x},\hat{t})$ be a candidate optimal solution. Set $UB=\hat{t}\ +\sum_{i\in V} c_i\hat{x}_i$ (or set $UB=\sum_{i\in V}(\hat{t}_i\ /\ 2\ +\ c_i\hat{x}_i)$ for multiple t_i version)

3: Obtain sum of weights of selected edges by $DSP(\hat{x})$ using Algorithm 1, denoted by t^*. Set $LB=t^*+\sum_{i\in V} c_i\hat{x}_i$.

4: **if** $UB=LB$ **then**

5: Current edge selection is a maximum weight induced matching of G, STOP.

6: **else**

7: Let $(\hat{\alpha},\hat{\beta},\hat{\gamma})$ denote an optimal solution of $DSP(\hat{x})$.

8: Generate optimality cut(s) using (12) (or (16) for multiple t_i version) with current optimal dual solution $(\hat{\alpha},\hat{\beta},\hat{\gamma})$, add it (them) to (MP). Go to step 2.

9: **end if**

(nonnegative) $\lfloor |V|/2\rfloor$ of them. Therefore an initial bound on the t-variable can be written as:

$$t\le UB_1 \tag{17}$$

Another key observation is that since any induced matching is also a matching, the sum of edge weights of an induced matching is bounded by the maximum weight matching in the graph. Therefore, we can find a tighter numerical bound on the t-variable, denoted by UB_2, by finding a maximum weight matching in the graph using Edmonds' blossom algorithm (Edmonds 1965). Then, the following inequality is valid for the t-variable and it dominates (17) :

$$t\le UB_2 \tag{18}$$

Similarly, if we use multiple t_i-variables in (MP) instead of a single variable t, a numerical bound for the sum of t_i-variables can be written as:

$$\sum_{i\in V} t_i\ /\ 2\le UB_2 \tag{19}$$

Another approach to find a valid inequality for the t-variable is as follows: for any vertex $i \in V$, if $x_i = 1$ (i.e., if it is saturated by an induced matching), it can increase the objective function value by at most by the maximum edge weight emanating from it and if $x_i = 0$ (i.e., if it is not saturated by an induced matching), there is no increase in the objective function value. Thus, the following inequality is valid for t:

$$t \leq \sum_{i \in V} \max_{j \in N(i)} \{w_{ij}\} \; x_i \; / \; 2 \tag{20}$$

Similarly, in the multiple t_i version of (MP), the only bound for t_i-variable is UB_i, which is the maximum edge weight emanating from vertex i. In the formulation, t_i-variables represent the maximum edge weight that can be obtained by saturating vertex i. With a similar argument, we observe that the following inequality is valid for t_i:

$$t_i \leq \max_{j \in N(i)} \{w_{ij}\} \; x_i \tag{21}$$

4.2. Formulation Tightening

In (MIM) model, constraints (2c) are based on the observation that if a vertex $i \in V$ is not saturated ($x_i = 0$), we can saturate at most $|N(i)|$ of its neighbors. However, it may not be possible to saturate all of its neighbors in an induced matching. To obtain a tighter upper bound for these constraints, we will use the following method to calculate the maximum number of neighbors that can be saturated in a matching:

For any vertex $i \in V$, let $S = N(i)$ and $S' = N(S) - \{i\}$. We assign weights to the edges of G such that all edges $\{\{i,j\} : i \in S, j \in S\}$ will have weight 2, $\{\{i,j\} : i \in S, j \in S'\}$ will have weight 1 and other edges will have weight 0. Figure 2 shows assignment of edge weights for vertex i. Then, finding the maximum number of vertices in S saturated by a matching is equivalent to finding a maximum weight matching in G, which can be found by weighted version of Edmonds' blossom algorithm (Edmonds 1965). If for each vertex $i \in V$, we find a maximum weight matching M_i and denote the sum of edge weights of this matching by $w(M_i)$, we can replace $|N(i)|$ in constraints (2c) with $w(M_i)$ and rewrite them as:

$$\sum_{j \in N(i)} x_j \leq (w(M_i) - 1)(1 - x_i) + 1 \quad \forall \, i \in V(G) \tag{22}$$

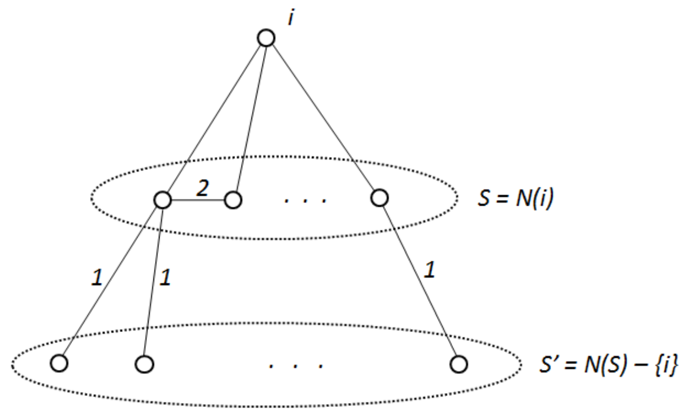

Figure 2 **An example of construction of a maximum weighted induced matching problem for vertex i**

4.3. Single Branch-and-Bound Tree

In our Benders decomposition approach, we solve (MP) to optimality at each iteration and check whether it is an optimal solution or not using a subproblem. If not, we add an optimality cut and re-solve it. Although new cuts may change the structure of the branch-and-bound tree, we may need to revisit candidate solutions that were discarded earlier. Hence, this process can be very expensive from a computational point of view. Instead, we can interrupt the branch-and-bound solution process of (MP) each time the solver finds an integer solution $\hat{x}$ (and $\hat{t}$) and check whether an optimality cut (12) (or (16) for multiple t_i version) that is violated by the current integer solution can be generated. If we can generate such an optimality cut, we reject the current solution, add the newly generated cuts to the problem and resume the solution process. Otherwise, we accept the current solution as the new incumbent and again resume the solution process. In our computational tests, this approach consistently outperformed solving (MP) to optimality at each iteration and re-optimizing it. With this approach, we can solve the problem using a single branch-and-bound tree that is tightened as necessary as opposed to repeatedly generating a branch-and-bound tree in each iteration. Therefore, we avoid considerable rework by never revisiting a branch-and-bound node and overlooking a truly superior solution. A similar approach was also used in (Bodur et al. 2013, Taşkın and Cevik 2013).

5. Computational Results

To test the efficiency of formulations and improvements mentioned in previous sections, we conducted a series of experiments. We executed all integer programming formulations using CPLEX 12.6.3 running on a Windows Server 2012 with a 2 GHz Intel Xeon CPU and 46 GB RAM. We also used CPLEX's callback functions to solve our model in a single branch-and-bound tree as described in Section 4.3. We used LEMON Graph Library 1.2.4 (Dezső et al. 2011) to efficiently implement graph related data structures and algorithms. Our base test data set contains randomly generated graph instances having expected edge density (measured as $D = \frac{2|E|}{|V|\times(|V|-1)}$) of 0.05, 0.2, 0.5 and 0.8, where $|V|$ is within the range [25–400]. To generate weighted instances we first generated random graphs as in the unweighted case, and then assigned an integer weight uniformly distributed between 1 and 10 to each vertex and/or edge. We generated five problem instances for each problem size, determined by the expected edge density and the number of vertices. Data sets used in our tests are available online at `www.ie.boun.edu.tr/~taskin/data/mwim_graphs.zip`

For each problem size, we report the following statistics calculated over five random instances:

- "Solved:" the number of problem instances solved to optimality within the allowed time limit of 1800 seconds.
- "Gap:" the average final percentage optimality gap for all instances (calculated as $(UB-LB)/LB$ where UB denotes the upper bound and LB denotes the lower bound).
- "LP Gap:" the average percentage gap between the objective function value of LP relaxation solution and the optimal objective function value (measured as $\frac{Z_{LP}-Z^*}{Z^*}$).
- "Time:" the average amount of time in seconds spent by each algorithm.
- "BB Node:" the average number of nodes processed within the given time limit in the active branch-and-bound search.

In Table 1, we compare the performance of Vassilaras and Christou (2011)'s formulation (VC2011) with our vertex-based formulation (MIM) given in Section 2.1. Also, we investigate the effect of formulation tightening approach described in Section 4.2 (set of columns titled "(MIM)-modified"). Here, "Time" column in (MIM)-modified also includes the required time to calculate maximum weight matchings in the graph. In the table, "-" shows that the corresponding method is unable to solve instances within the given time limit.

We observe that for low-density instances ($D = 0.05$ and $D = 0.2$), (VC2011) has smaller CPU time than (MIM) and (MIM)-modified. On the other hand, for moderate and high densities ($D = 0.5$ and $D = 0.8$, respectively), (MIM) performs significantly faster and

Table 1 Comparison of formulations for solving MIM

		(VC 2011)					(MIM)					(MIM)-modified				
D	$\lvert V\rvert$	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes
	25	5	0%	0%	0.1	0	5	0%	11%	0.1	0	5	0%	5%	0.2	0
	50	5	0%	5%	0.2	0	5	0%	24%	0.5	18	5	0%	18%	0.6	0
	75	5	0%	14%	1.6	2	5	0%	32%	3.7	962	5	0%	29%	3.9	973
0.05	100	5	0%	25%	5.1	153	5	0%	38%	18.9	35421	5	0%	37%	23.7	36900
	125	5	0%	28%	47.8	5086	5	0%	34%	715.3	432856	5	0%	32%	786.8	386531
	150	5	0%	38%	1187.1	155685	0	14%	-	1800.0	329828	0	15%	-	1800.0	398863
	175	0	22%	-	2899.1	282031	-	-	-	-	-	-	-	-	-	-
	25	5	0%	26%	0.1	0	5	0%	26%	0.1	0	5	0%	26%	0.3	0
	50	5	0%	63%	0.4	63	5	0%	70%	1.1	2936	5	0%	70%	1.5	2936
0.2	75	5	0%	96%	12.7	5343	5	0%	96%	23.4	50346	5	0%	96%	33.7	50346
	100	5	0%	129%	848.1	150336	2	5%	117%	1740.0	3442803	2	7%	117%	1729.6	3442930
	125	0	22%	-	1800.0	370826	0	42%	-	1800.0	5435342	0	45%	-	1800.0	5435342
	25	5	0%	118%	0.2	0	5	0%	117%	0.1	0	5	0%	117%	0.2	0
	50	5	0%	219%	4.1	2992	5	0%	220%	0.6	4468	5	0%	220%	1.0	4468
	75	5	0%	280%	116.0	14702	5	0%	280%	7.3	27508	5	0%	280%	8.3	27508
0.5	100	4	2%	-	1640.8	129672	5	0%	321%	44.1	121631	5	0%	321%	47.3	121631
	125	0	423%	-	1800.0	82062	5	0%	425%	172.0	345995	5	0%	425%	192.1	345995
	150	-	-	-	-	-	5	0%	529%	986.8	1369073	5	0%	529%	1054.8	1369073
	175	-	-	-	-	-	0	26%	-	1800.0	1643480	0	27%	-	1800.0	1643480
	25	5	0%	221%	0.2	86	5	0%	221%	0.1	0	5	0%	221%	0.3	0
	50	5	0%	328%	2.0	903	5	0%	428%	0.3	255	5	0%	428%	0.6	255
	75	5	0%	430%	221.6	4249	5	0%	530%	0.9	2017	5	0%	530%	1.5	2017
	100	0	538%	-	1800.0	2575	5	0%	739%	3.3	5026	5	0%	739%	4.4	5026
0.8	125	-	-	-	-	-	5	0%	947%	12.5	8309	5	0%	947%	14.0	8309
	150	-	-	-	-	-	5	0%	1155%	19.6	23272	5	0%	1155%	22.5	23272
	175	-	-	-	-	-	5	0%	1364%	42.0	37706	5	0%	1364%	48.4	37706
	200	-	-	-	-	-	5	0%	1572%	81.4	78074	5	0%	1572%	93.7	78074
	225	-	-	-	-	-	5	0%	1623%	150.2	92436	5	0%	1623%	172.7	92436
	250	-	-	-	-	-	5	0%	1795%	317.4	104562	5	0%	1795%	343.0	104562
	300	-	-	-	-	-	5	0%	1949%	846.2	136585	5	0%	1949%	880.4	136585
	350	-	-	-	-	-	3	43%	-	1702.7	126408	3	53%	-	1750.2	126408
	400	-	-	-	-	-	0	313%	-	1800.0	146838	0	313%	-	1800.0	146838

solves more instances within the given time limit. Another observation is that for $D = 0.05$, (MIM)-modified method provides better LP relaxation bounds, but for higher densities, it does not have a positive effect on the bound. It can be seen that there is no significant difference between (MIM) and (MIM)-modified. Therefore, usage of the formulation tightening approach suggested in Section 4.2 is not justified. In addition, we implemented a cutting-plane algorithm to solve (VC2011) in which we add (1b) as lazy constraints. However, according to our computational tests, this approach does not provide better results compared to adding all constraints to the formulation.

Recall that in our Benders decomposition approach proposed in Section 3, $DSP(\hat{x})$ can have infinitely many solutions. By changing θ_{ij} values, we can obtain different $(\hat{\alpha}, \hat{\beta}, \hat{\gamma})$, resulting in different cuts in (12) and (16). To test the impact of θ_{ij} selection, we solved all instances by setting all θ_{ij} values to 0.2, 0.5 and 0.8. We also tried to assign random values between 0 and 1 to θ_{ij} in each iteration. Note that regardless of θ_{ij} values, all possible optimality cuts (12) or (16) have the same violation (see Proposition 2), therefore there is no significant difference between these options. Our computational results showed that θ_{ij} selection has no clear effect on the total running time and the optimality gap. Therefore, we set $\theta_{ij} = 0.5$ in the remaining parts for simplicity.

Our second experiment compares the performances of MWIM formulations given in Section 3 and investigates the effect of proposed improvements suggested in Section 4.1.

The results of this experiment are summarized in Tables 2–6. In Table 2, we compare the performances of Vassilaras and Christou (2011)'s formulation (VC2011 MWIM) with our formulation quadratic programming formulation (MWIM QP) and its linearization (MWIM). Our first observation is that (VC2011 MWIM) and (MWIM) significantly outperform (MWIM QP). (MWIM QP) is unable to solve even instances with small number of vertices to optimality within the given time limit. We also observe that (VC2011 MWIM) can solve more instances to optimality within the given time limit whereas our (MWIM) model has higher total running times and optimality gaps for all densities.

In Table 3, we measure the effect of the suggested improvements on the decomposition approach where we have a single t-variable in (MP). Here, we report the following results: "Single t" Algorithm 2 with a single t-variable in (MP), "Single t + (17)" Algorithm 2 where we use (17) as an initial bound on the t-variable in (MP), "Single t + (18)" Algorithm 2 with (18) as an initial bound on the t-variable in (MP), "Single t + (20)" Algorithm 2 augmented with valid inequality (20) in (MP). For the model with a single t-variable in (MP), it can be observed that inequality (17) yields a slight improvement in total running time. However, it is unable to solve instances that could not be solved by (MWIM); it can only decrease the optimality gap for these instances. Inequality (18), in which we calculate an upper bound on the t-variable by finding a maximum weight matching in the graph, does not provide better solutions. On the other hand, inequality (20) has a significant impact on the performance in terms of computational time and allows us to solve more instances in the enforced time limit.

In Table 4, we investigate the effect of valid inequalities (19) and (21) on the decomposition approach where we have multiple t_i-variables in (MP). We report the following results: "Multiple t" Algorithm 2 with multiple t_i-variables in (MP), "Multiple t + (21)" Algorithm 2 improved with valid inequality (21) in (MP), "Multiple t + (19) + (21)" Algorithm 2 with valid inequalities (19) and (21) in (MP). Here, we note that valid inequality (21) yields a significant improvement in terms of computational effort. However, adding inequality (19) into (MP) slightly increases the total running time. As a result, our improved algorithm contains the valid inequality (21) but not (19). We also tried to separate Benders cuts for integer and fractional solutions, but we observed that separating Benders cuts only for integer solutions is better in terms of computational effort.

Table 2 Comparison of formulations for solving MWIM

		(VC2011 MWIM)					(MWIM QP)					(MWIM)				
D	$\lvert V\rvert$	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes
0.05	25	5	0%	0%	0.1	0	5	0%	12%	7.3	-	5	0%	20%	0.3	0
	50	5	0%	3%	1.3	0	0	17%	-	1800.0	-	5	0%	59%	1.3	0
	75	5	0%	10%	1.9	0	-	-	-	-	-	5	0%	106%	8.9	959
	100	5	0%	20%	75.5	221	-	-	-	-	-	5	0%	154%	878.6	8596
	125	5	0%	24%	187.7	3021	-	-	-	-	-	0	44%	-	1800.0	16214
	150	5	0%	33%	419.2	47750	-	-	-	-	-	-	-	-	-	-
	175	3	9%	-	1774.4	74232	-	-	-	-	-	-	-	-	-	-
0.2	25	5	0%	18%	0.2	0	0	1%	-	1800.0	-	5	0%	145%	0.4	0
	50	5	0%	59%	0.4	153	-	-	-	-	-	5	0%	454%	6.7	13917
	75	5	0%	98%	9.9	4635	-	-	-	-	-	5	0%	817%	1231.2	55539
	100	5	0%	126%	234.3	66082	-	-	-	-	-	0	164%	-	1800.0	90722
	125	0	33%	-	1800.0	276523	-	-	-	-	-	-	-	-	-	-
0.5	25	5	0%	94%	0.2	0	0	5%	-	1800.0	-	5	0%	692%	0.5	0
	50	5	0%	190%	5.0	101	-	-	-	-	-	5	0%	1955%	12.0	3418
	75	5	0%	288%	70.8	6556	-	-	-	-	-	5	0%	3880%	93.0	49988
	100	5	0%	394%	1293.8	19423	-	-	-	-	-	5	0%	5452%	1463.5	83631
	125	0	139%	-	1800.0	31329	-	-	-	-	-	0	745%	-	1800.0	153238
0.8	25	5	0%	191%	0.3	0	0	15%	-	1800.0	-	5	0%	1560%	0.7	0
	50	5	0%	385%	3.6	0	-	-	-	-	-	5	0%	5276%	12.0	420
	75	5	0%	580%	47.6	0	-	-	-	-	-	5	0%	10957%	51.6	4255
	100	5	0%	686%	199.9	1270	-	-	-	-	-	5	0%	16724%	263.7	8783
	125	5	0%	824%	909.5	7080	-	-	-	-	-	5	0%	24120%	934.8	16193
	150	0	82%	-	1800.0	12999	-	-	-	-	-	0	6128%	-	1800.0	17180

In Table 5, we summarize the best implementations for solving MWIM. If we compare the running times of (MWIM), in which we have all decision variables and constraints, and our Benders decomposition approach, it can be observed that our decomposition approach outperforms (MWIM) for all instances. Furthermore, comparing the two decomposition approaches, we note that the use of multiple t_i-variables in (MP) along with valid inequalities (21) significant improve solution times and optimality gaps for all instances. By comparing its performance against (VC2011 MWIM), we note that directly solving (VC2011 MWIM) is faster than our decomposition approach for low density graphs. However, our algorithm performs better for medium and large densities. In our preliminary analysis, we observed that we can find close-optimal solutions in the first iterations, but it takes longer to improve the upper bound. Therefore, primal side of the solution process is not problematic, and there is no need to use a heuristic within our decomposition algorithm. Detailed results (lower bounds, upper bounds, optimality gaps and solutions times) for each instance are given in the Appendix A.

In our last experiment, we increase the range of the weights from 10 to 100. In addition, to incorporate the case where we have some node / edge weights are negative, we kept range as 100 and assigned an integer weight uniformly distributed in [-20,80]. The result of this experiment is given in Table 6. If we increase the range of the weights from 10 to 100, objective function coefficients of variables in the model can be differentiated more easily, thus reducing symmetry effect. Hence, it gives better results for almost all instances.

Table 3 **Effect of valid inequalities on the decomposition approach with a single t-variable in (MP)**

		Single t					Single t + (17)					Single t + (18)					Single t + (20)				
D	$\|V\|$	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes
0.05	25	5	0%	-	0.5	7	5	0%	49%	0.5	8	5	0%	16%	0.5	9	5	0%	7%	0.3	2
	50	5	0%	-	7.3	2427	5	0%	80%	7.1	2383	5	0%	46%	7.8	3344	5	0%	27%	3.9	678
	75	1	12%	-	1734.8	40502	1	11%	-	1729.1	361658	0	13%	-	1800.0	439859	5	0%	37%	230.4	41007
	100	0	93%	-	1800.0	19044	0	91%	-	1800.0	157573	-	-	-	-	-	0	13%	-	1800.0	329228
0.2	25	5	0%	-	0.5	740	5	0%	101%	0.7	803	5	0%	74%	0.6	888	5	0%	45%	0.4	170
	50	3	11%	-	1433.6	195075	3	11%	-	1226.4	229565	3	16%	-	1498.2	22019	5	0%	79%	7.6	10879
	75	0	409%	-	1800.0	46166	0	183%	-	1800.0	49012	0	166%	-	1800.0	49748	3	7%	-	1552.8	536950
	100	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	29%	-	1800.0	2950217
0.5	25	5	0%	-	1.2	440	5	0%	222%	1.1	512	5	0%	94%	1.3	509	5	0%	110%	0.3	424
	50	5	0%	-	237.1	1140	5	0%	296%	240.3	1263	5	0%	138%	238.7	1260	5	0%	149%	8.2	5083
	75	0	315%	-	1800.0	9476	0	231%	-	1800.0	9955	0	224%	-	1800.0	9819	5	0%	196%	39.2	7345
	100	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	0%	214%	153.2	13174
	125	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	0%	272%	614.7	18423
	150	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	30%	-	1800.0	19535
0.8	25	5	0%	-	2.5	251	5	0%	257%	1.5	261	5	0%	229%	1.6	279	5	0%	202%	0.5	185
	50	5	0%	-	227.5	551	5	0%	459%	167.3	572	5	0%	438%	166.9	571	5	0%	292%	4.5	349
	75	5	0%	-	814.9	3134	5	0%	635%	829.5	3257	5	0%	646%	889.7	3250	5	0%	492%	12.8	4744
	100	0	1179%	-	1800.0	4466	0	619%	-	1800.0	4790	0	606%	-	1800.0	4706	5	0%	528%	25.7	5724
	125	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	0%	665%	83.9	6154
	150	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	0%	828%	257.7	8435
	175	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	0%	965%	414.3	10349
	200	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	0%	1023%	1151.8	12856
	225	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	70%	-	1800	13962

Table 4 **Effect of valid inequalities on the decomposition approach with multiple t_i-variables in (MP)**

		Multiple t					Multiple t + (21)					Multiple t + (19) + (21)				
D	$\|V\|$	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes
0.05	25	5	0%		0.6	5	5	0%	7%	0.4	0	5	0%	7%	0.4	0
	50	5	0%		2.3	873	5	0%	27%	1.6	110	5	0%	27%	1.9	105
	75	5	0%		36.2	102334	5	0%	37%	4.4	2981	5	0%	37%	5.0	3831
	100	0	21%		1800.0	6085709	5	0%	51%	118.3	224166	5	0%	51%	184.0	171968
	125	-	-		-	-	0	12%	-	1800.0	1055447	0	15%	-	1800.0	538861
0.2	25	5	0%		0.6	425	5	0%	45%	0.3	120	5	0%	45%	0.4	142
	50	5	0%		22.3	61726	5	0%	79%	2.5	9963	5	0%	79%	4.2	10033
	75	0	68%		1800.0	1470049	5	0%	113%	424.3	371791	5	0%	113%	476.1	380668
	100	-	-		-	-	0	12%	-	1800.0	5649446	0	18%	-	1800.0	3933845
0.5	25	5	0%		0.7	774	5	0%	110%	0.6	383	5	0%	110%	0.6	372
	50	5	0%		89.9	14340	5	0%	199%	3.5	4450	5	0%	199%	4.1	4444
	75	0	414%		1800.0	51362	5	0%	246%	23.3	26959	5	0%	246%	29.6	27962
	100	-	-		-	-	5	0%	296%	59.4	82384	5	0%	296%	93.5	83484
	125	-	-		-	-	5	0%	352%	513.7	113043	5	0%	352%	685.8	109456
	150	-	-		-	-	0	14%	-	1800.0	107242	0	19%	-	1800.0	99437
0.8	25	5	0%		2.1	489	5	0%	122%	0.4	161	5	0%	122%	0.7	163
	50	5	0%		121.6	2782	5	0%	292%	3.4	1400	5	0%	292%	3.4	1289
	75	0	142%		1800.0	4612	5	0%	386%	7.6	4580	5	0%	386%	8.4	4792
	100	-	-		-	-	5	0%	476%	12.9	9434	5	0%	476%	13.1	9672
	125	-	-		-	-	5	0%	592%	39.5	16973	5	0%	592%	45.9	18297
	150	-	-		-	-	5	0%	668%	71.4	32648	5	0%	668%	80.7	33665
	175	-	-		-	-	5	0%	728%	96.4	57453	5	0%	728%	148.2	53972
	200	-	-		-	-	5	0%	858%	229.0	89962	5	0%	858%	245.5	82593
	225	-	-		-	-	5	0%	945%	458.3	103481	5	0%	945%	502.7	105942
	250	-	-		-	-	5	0%	1034%	600.6	146259	5	0%	1034%	792.9	145295
	300	-	-		-	-	2	11%	-	1643.7	154641	2	13%	-	1713.9	129325

Table 5 Summary of best implementations for solving MWIM

		(VC2011 MWIM)					(MWIM)					Single t + (20)					Multiple t + (21)				
D	$\|V\|$	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes
0.05	25	5	0%	0%	0.1	0	5	0%	20%	0.3	0	5	0%	7%	0.3	2	5	0%	7%	0.4	0
	50	5	0%	3%	1.3	0	5	0%	59%	1.3	0	5	0%	27%	3.9	678	5	0%	27%	1.6	110
	75	5	0%	10%	1.9	0	5	0%	106%	8.9	959	5	0%	37%	230.4	41007	5	0%	37%	4.4	2981
	100	5	0%	20%	75.5	221	5	0%	154%	878.6	8596	0	13%	-	1800.0	329228	5	0%	51%	118.3	224166
	125	5	0%	24%	187.7	3021	0	44%	-	1800.0	16214	-	-	-	-	-	0	12%	-	1800.0	1055447
	150	5	0%	33%	419.2	47750	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	175	3	9%	-	1774.4	74232	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0.2	25	5	0%	18%	0.2	0	5	0%	145%	0.4	0	5	0%	45%	0.4	170	5	0%	45%	0.3	120
	50	5	0%	59%	0.4	153	5	0%	454%	6.7	13917	5	0%	79%	7.6	10879	5	0%	79%	2.5	9963
	75	5	0%	98%	9.9	4635	5	0%	817%	1231.2	55539	3	7%	-	1552.8	536950	5	0%	113%	424.3	371791
	100	5	0%	126%	234.3	66082	0	164%	-	1800.0	90722	0	29%	-	1800.0	2950217	0	12%	-	1800.0	5649446
	125	0	33%	-	1800.0	276523	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0.5	25	5	0%	94%	0.2	0	5	0%	692%	0.5	0	5	0%	110%	0.3	424	5	0%	110%	0.6	383
	50	5	0%	190%	5.0	101	5	0%	1955%	12.0	3418	5	0%	149%	8.2	5083	5	0%	199%	3.5	4450
	75	5	0%	288%	70.8	6556	5	0%	3880%	93.0	49988	5	0%	196%	39.2	7345	5	0%	246%	23.3	26959
	100	5	0%	394%	1293.8	19423	5	0%	5452%	1463.5	83631	5	0%	214%	153.2	13174	5	0%	296%	59.4	82384
	125	0	139%	-	1800.0	31329	0	745%	-	1800.0	153238	5	0%	272%	614.7	18423	5	0%	352%	513.7	113043
	150	-	-	-	-	-	-	-	-	-	-	0	30%	-	1800.0	19535	0	14%	-	1800.0	107242
0.8	25	5	0%	191%	0.3	0	5	0%	1560%	0.7	0	5	0%	202%	0.5	185	5	0%	122%	0.4	161
	50	5	0%	385%	3.6	0	5	0%	5276%	12.0	420	5	0%	292%	4.5	349	5	0%	292%	3.4	1400
	75	5	0%	580%	47.6	0	5	0%	10957%	51.6	4255	5	0%	492%	12.8	4744	5	0%	386%	7.6	4580
	100	5	0%	686%	199.9	1270	5	0%	16724%	263.7	8783	5	0%	528%	25.7	5724	5	0%	476%	12.9	9434
	125	5	0%	824%	909.5	7080	5	0%	24120%	934.8	16193	5	0%	665%	83.9	6154	5	0%	592%	39.5	16973
	150	0	82%	-	1800.0	12999	0	6128%	-	1800.0	17180	5	0%	828%	257.7	8435	5	0%	668%	71.4	32648
	175	-	-	-	-	-	-	-	-	-	-	5	0%	965%	414.3	10349	5	0%	728%	96.4	57453
	200	-	-	-	-	-	-	-	-	-	-	5	0%	1023%	1151.8	12856	5	0%	858%	229.0	89962
	225	-	-	-	-	-	-	-	-	-	-	0	70%	-	1800.0	13962	5	0%	945%	458.3	103481
	250	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	0%	1034%	600.6	146259
	300	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2	11%	-	1643.7	154641

Table 6 Effect of changing weight range as [-20,80]

		(VC2011 MWIM)					(MWIM)					Single t + (20)					Multiple t + (21)				
D	$\|V\|$	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes	Solved	Gap	LP Gap	Time	BB Nodes
0.05	25	5	0%	2%	0.2	0	5	0%	36%	0.2	0	5	0%	16%	0.2	0	5	0%	68%	0.2	0
	50	5	0%	9%	1.2	0	5	0%	82%	1.2	0	5	0%	34%	3.5	526	5	0%	104%	1.5	145
	75	5	0%	34%	1.7	0	5	0%	140%	8.2	856	5	0%	64%	218.5	53732	5	0%	147%	4.0	1406
	100	5	0%	73%	72.6	184	5	0%	215%	819.6	6584	0	11%	-	1800.0	242948	5	0%	169%	116.3	28957
	125	5	0%	89%	160.9	2863	0	31%	-	1800.0	14346	-	-	-	-	-	0	10%	-	1800.0	104272
	150	5	0%	103%	403.5	45238	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	175	4	4%	-	1742.9	69234	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0.2	25	5	0%	79%	0.2	0	5	0%	675%	0.3	0	5	0%	121%	0.4	145	5	0%	221%	0.3	92
	50	5	0%	262%	0.4	109	5	0%	2003%	6.1	12587	5	0%	271%	7.1	18859	5	0%	371%	2.2	10673
	75	5	0%	473%	9.0	3718	5	0%	3780%	1185.3	53550	5	0%	283%	1490.0	558444	5	0%	546%	418.7	400601
	100	5	0%	535%	226.6	179686	0	161%	-	1800.0	84636	0	17%	-	1800.0	2941822	0	9%	-	1800.0	4904526
	125	0	28%	-	1800.0	597346	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0.5	25	5	0%	426%	0.2	0	5	0%	3077%	0.6	0	5	0%	519%	0.3	311	5	0%	519%	0.4	279
	50	5	0%	975%	4.4	0	5	0%	9621%	11.8	4047	5	0%	1036%	8.1	5342	5	0%	1036%	3.5	4770
	75	5	0%	1412%	62.6	7437	5	0%	18455%	85.5	51854	5	0%	1458%	32.6	42964	5	0%	1458%	21.1	37211
	100	5	0%	1907%	1127.4	98370	5	0%	29033%	1303.9	205680	5	0%	1950%	155.8	186475	5	0%	1950%	52.5	171567
	125	0	83%	-	1800.0	145112	0	692%	-	1800.0	394224	5	0%	2464%	602.0	188459	5	0%	2457%	466.7	253792
	150	-	-	-	-	-	-	-	-	-	-	0	25%	-	1800.0	263683	0	12%	-	1800.0	207452
0.8	25	5	0%	819%	0.3	0	5	0%	6592%	0.8	0	5	0%	885%	0.4	164	5	0%	885%	0.4	147
	50	5	0%	1752%	3.5	0	5	0%	23562%	11.8	999	5	0%	1801%	4.1	1204	5	0%	1101%	3.1	1321
	75	5	0%	2913%	41.5	0	5	0%	52935%	48.5	4066	5	0%	2963%	12.7	4384	5	0%	1863%	7.1	4811
	100	5	0%	3644%	178.6	2973	5	0%	86168%	213.4	9291	5	0%	3689%	22.8	9013	5	0%	2389%	11.6	8889
	125	5	0%	3368%	835.3	11908	5	0%	70917%	761.0	15740	5	0%	3899%	80.4	11418	5	0%	2555%	34.2	19605
	150	0	59%	-	1800.0	29524	0					5	0%	4927%	237.5	15908	5	0%	2957%	69.3	35837
	175	-	-	-	-	-	-	-	-	-	-	5	0%	5283%	411.8	20573	5	0%	3286%	94.2	58206
	200	-	-	-	-	-	-	-	-	-	-	5	0%	5926%	1103.9	24891	5	0%	3689%	203.4	72853
	225	-	-	-	-	-	-	-	-	-	-	0	61%	-	1800.0	23954	5	0%	4035%	424.0	89281
	250	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	0%	4723%	584.9	106343
	300	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	3	7%	-	1656.3	119454

6. Conclusions and Future Research

In this paper, we studied the Maximum Induced Matching problem (MIM), where the aim is to find an induced matching with the largest cardinality. Induced matchings can be used to determine the maximum capacity of MAC layer in wireless ad-hoc networks.

The problem is known to be NP-hard for general graphs, even for bipartite graphs. In the literature, the problem is studied for some restricted graph classes. It is addressed from mathematical programming point of view by Vassilaras and Christou (2011). We proposed a new vertex-based integer programming formulation for MIM, which has fewer number of binary variables and constraints than the formulation in the literature. Then, we described vertex-weighted and edge-weighted versions of the problem, and named them Maximum Vertex-Weighted Induced Matching problem (MVWIM) and Maximum Edge-Weighted Induced Matching problem (MEWIM), respectively. We reformulated the models in the literature and our model to solve weighted instances.

In the Maximum Weight Induced Matching problem (MWIM), we considered graphs with both edge and wertex weights. We formulated MWIM as a quadratic programming problem and gave its linearized version. As the formulation for MWIM contains many decision variables and constraints, we applied Benders decomposition to our proposed formulation (MWIM). Our decomposition algorithm seeks a feasible induced matching using a master problem and reaches optimality using cuts generated by a subproblem, which we showed to be solvable by inspection. We proposed upper bounds on variables and valid inequalities in the master problem to improve the efficiency of our algorithm.

We tested the efficacy of our approach on randomly generated graph instances. Our computational results show that our decomposition approach performs better than solving the underlying integer programming formulation. Also, it outperforms the formulations found in the literature for medium and high densities.

As a future research, one can focus on solving MIM, MVWIM, MEWIM and MWIM problems in some specific graph classes and consider developing additional valid inequalities, preprocessing rules and heuristics using their structural properties.

References

H. Balakrishnan, C. L. Barrett, V. A. Kumar, M. V. Marathe, and S. Thite. The distance-2 matching problem and its relationship to the MAC-layer capacity of ad hoc wireless networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1069–1079, 2004.

M. Bodur, T. Ekim, and Z. C. Taşkın. Decomposition algorithms for solving the minimum weight maximal matching problem. *Networks*, 62(4):273–287, 2013.

K. Cameron. Induced matchings. *Discrete Applied Mathematics*, 24(1):97–102, 1989.

K. Cameron, R. Sritharan, and Y. Tang. Finding a maximum induced matching in weakly chordal graphs. *Discrete Mathematics*, 266(1):133–142, 2003.

M. S. Chang, L. H. Chen, and L. J. Hung. Moderately exponential time algorithms for the maximum induced matching problem. *Optimization Letters*, 9(5):981–998, 2015.

I. T. Christou and S. Vassilaras. A parallel hybrid greedy branch and bound scheme for the maximum distance-2 matching problem. *Computers & Operations Research*, 40(10):2387–2397, 2013.

K. K. Dabrowski, M. Demange, and V. V. Lozin. New results on maximum induced matchings in bipartite graphs and beyond. *Theoretical Computer Science*, 478:33–40, 2013.

B. Dezső, A. Jüttner, and P. Kovács. LEMON–an open source C++ graph template library. *Electronic Notes in Theoretical Computer Science*, 264(5):23–45, 2011.

W. Duckworth, N. Wormald, and M. Zito. Maximum induced matchings of random cubic graphs. *Journal of Computational and Applied Mathematics*, 142(1):39–50, 2002.

W. Duckworth, D. F. Manlove, and M. Zito. On the approximability of the maximum induced matching problem. *Journal of Discrete Algorithms*, 3(1):79–91, 2005.

J. Edmonds. Paths, trees, and flowers. *Canadian Journal of mathematics*, 17(3):449–467, 1965.

G. Fricke and R. Laskar. Strong matchings on trees. *Congressus Numerantium*, 89:239–243, 1992.

M. C. Golumbic and R. C. Laskar. Irredundancy in circular arc graphs. *Discrete Applied Mathematics*, 44 (1):79–89, 1993.

H. Moser and S. Sikdar. The parameterized complexity of the induced matching problem. *Discrete Applied Mathematics*, 157(4):715–727, 2009.

Y. Orlovich, G. Finke, V. Gordon, and I. Zverovich. Approximability results for the maximum and minimum maximal induced matching problems. *Discrete Optimization*, 5(3):584–593, 2008.

L. Stockmeyer and V. Vazirani. NP-completeness of some generalizations of the maximum matching problem. *Information Processing Letters*, 15(1):14–19, 1982.

Z. C. Taşkın and M. Cevik. Combinatorial Benders cuts for decomposing imrt fluence maps using rectangular apertures. *Computers & Operations Research*, 40(9):2178–2186, 2013.

S. Vassilaras and I. T. Christou. On the optimal MAC layer capacity of delay tolerant mobile ad hoc networks with a finite number of nodes. In *2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 772–776. IEEE, 2011.

Appendix A: Detailed Results of Table 5

In Table 7 and Table 8, we gave the detailed results (lower bounds, upper bounds, optimality gaps and solutions times) of Table 5 for each instance. All instances are available online at `www.ie.boun.edu.tr/~taskin/data/mwim_graphs.zip`

Table 7 Detailed results of Table 5

			(VC2011 MWIM)				(MWIM)				Single t + (20)				Multiple t + (21)			
Graph Name	D	$\|V\|$	LB	UB	Gap	Time	LB	UB	Gap	Time	LB	UB	Gap	Time	LB	UB	Gap	Time
Graph0.050000_25_0	0.05	25	102	102.0	0%	0.1	102	102.0	0%	0.3	102	102.0	0%	0.5	102	102.0	0%	0.6
Graph0.050000_25_1	0.05	25	102	102.0	0%	0.1	102	102.0	0%	0.3	102	102.0	0%	0.3	102	102.0	0%	0.2
Graph0.050000_25_2	0.05	25	110	110.0	0%	0.1	110	110.0	0%	0.3	110	110.0	0%	0.4	110	110.0	0%	0.7
Graph0.050000_25_3	0.05	25	108	108.0	0%	0.2	108	108.0	0%	0.2	108	108.0	0%	0.3	108	108.0	0%	0.3
Graph0.050000_25_4	0.05	25	107	107.0	0%	0.2	107	107.0	0%	0.2	107	107.0	0%	0.2	107	107.0	0%	0.1
Graph0.050000_50_0	0.05	50	209	209.0	0%	1.4	209	209.0	0%	1.5	209	209.0	0%	5.5	209	209.0	0%	1.3
Graph0.050000_50_1	0.05	50	205	205.0	0%	1.1	205	205.0	0%	1.3	205	205.0	0%	2.4	205	205.0	0%	1.2
Graph0.050000_50_2	0.05	50	232	232.0	0%	1.7	232	232.0	0%	1.0	232	232.0	0%	3.2	232	232.0	0%	1.5
Graph0.050000_50_3	0.05	50	207	207.0	0%	1.2	207	207.0	0%	1.3	207	207.0	0%	2.1	207	207.0	0%	1.5
Graph0.050000_50_4	0.05	50	247	247.0	0%	1.4	247	247.0	0%	1.2	247	247.0	0%	6.5	247	247.0	0%	2.3
Graph0.050000_75_0	0.05	75	317	317.0	0%	1.8	317	317.0	0%	7.9	317	317.0	0%	268.2	317	317.0	0%	3.9
Graph0.050000_75_1	0.05	75	306	306.0	0%	2.4	306	306.0	0%	9.8	306	306.0	0%	297.4	306	306.0	0%	7.1
Graph0.050000_75_2	0.05	75	322	322.0	0%	1.9	322	322.0	0%	7.4	322	322.0	0%	183.9	322	322.0	0%	3.3
Graph0.050000_75_3	0.05	75	315	315.0	0%	2.5	315	315.0	0%	12.0	315	315.0	0%	236.6	315	315.0	0%	4.5
Graph0.050000_75_4	0.05	75	301	301.0	0%	0.9	301	301.0	0%	7.4	301	301.0	0%	165.7	301	301.0	0%	3.2
Graph0.050000_100_0	0.05	100	382	382.0	0%	71.4	382	382.0	0%	882.9	379	427.2	13%	1800.0	382	382.0	0%	163.2
Graph0.050000_100_1	0.05	100	374	374.0	0%	117.6	374	374.0	0%	872.2	371	444.8	20%	1800.0	374	374.0	0%	129.9
Graph0.050000_100_2	0.05	100	406	406.0	0%	55.7	406	406.0	0%	818.2	401	446.2	11%	1800.0	406	406.0	0%	92.6
Graph0.050000_100_3	0.05	100	396	396.0	0%	81.4	396	396.0	0%	941.0	391	430.2	10%	1800.0	396	396.0	0%	72.1
Graph0.050000_100_4	0.05	100	388	388.0	0%	51.5	388	388.0	0%	878.6	388	422.7	9%	1800.0	388	388.0	0%	133.8
Graph0.050000_125_0	0.05	125	501	501.0	0%	163.2	501	651.9	30%	1800.0	-	-	-	-	501	510.1	2%	1800.0
Graph0.050000_125_1	0.05	125	482	482.0	0%	210.3	468	693.9	48%	1800.0	-	-	-	-	476	557.8	17%	1800.0
Graph0.050000_125_2	0.05	125	512	512.0	0%	171.4	508	647.9	28%	1800.0	-	-	-	-	512	529.6	3%	1800.0
Graph0.050000_125_3	0.05	125	468	468.0	0%	201.0	457	673.0	47%	1800.0	-	-	-	-	457	527.1	15%	1800.0
Graph0.050000_125_4	0.05	125	492	492.0	0%	192.5	477	791.9	66%	1800.0	-	-	-	-	471	571.1	21%	1800.0
Graph0.050000_150_0	0.05	150	569	569.0	0%	395.4	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.050000_150_1	0.05	150	590	590.0	0%	539.8	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.050000_150_2	0.05	150	545	545.0	0%	456.3	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.050000_150_3	0.05	150	596	596.0	0%	352.9	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.050000_150_4	0.05	150	551	551.0	0%	351.5	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.050000_175_0	0.05	175	595	636.7	7%	1800.0	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.050000_175_1	0.05	175	614	661.9	8%	1800.0	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.050000_175_2	0.05	175	531	598.9	13%	1800.0	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.050000_175_3	0.05	175	588	648.9	10%	1800.0	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.050000_175_4	0.05	175	589	629.6	7%	1800.0	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.200000_25_0	0.2	25	91	91.0	0%	0.1	91	91.0	0%	0.1	91	91.0	0%	0.4	91	91.0	0%	0.2
Graph0.200000_25_1	0.2	25	98	98.0	0%	0.4	98	98.0	0%	0.4	98	98.0	0%	0.4	98	98.0	0%	0.2
Graph0.200000_25_2	0.2	25	126	126.0	0%	0.3	126	126.0	0%	0.6	126	126.0	0%	0.4	126	126.0	0%	0.4
Graph0.200000_25_3	0.2	25	113	113.0	0%	0.2	113	113.0	0%	0.3	113	113.0	0%	0.4	113	113.0	0%	0.3
Graph0.200000_25_4	0.2	25	105	105.0	0%	0.2	105	105.0	0%	0.3	105	105.0	0%	0.4	105	105.0	0%	0.3
Graph0.200000_50_0	0.2	50	152	152.0	0%	0.4	152	152.0	0%	5.0	152	152.0	0%	9.4	152	152.0	0%	2.5
Graph0.200000_50_1	0.2	50	172	172.0	0%	0.4	172	172.0	0%	7.0	172	172.0	0%	9.3	172	172.0	0%	3.8
Graph0.200000_50_2	0.2	50	148	148.0	0%	0.4	148	148.0	0%	7.0	148	148.0	0%	7.2	148	148.0	0%	2.0
Graph0.200000_50_3	0.2	50	155	155.0	0%	0.3	155	155.0	0%	6.5	155	155.0	0%	6.4	155	155.0	0%	1.3
Graph0.200000_50_4	0.2	50	165	165.0	0%	0.4	165	165.0	0%	8.2	165	165.0	0%	5.6	165	165.0	0%	2.9
Graph0.200000_75_0	0.2	75	201	201.0	0%	10.0	201	201.0	0%	943.6	201	211.4	5%	1800.0	201	201.0	0%	390.8
Graph0.200000_75_1	0.2	75	200	200.0	0%	9.8	200	200.0	0%	1330.6	200	200.0	0%	1039.1	200	200.0	0%	532.9
Graph0.200000_75_2	0.2	75	188	188.0	0%	12.0	188	188.0	0%	1316.9	188	188.0	0%	1325.0	188	188.0	0%	307.3
Graph0.200000_75_3	0.2	75	193	193.0	0%	8.4	193	193.0	0%	1328.7	193	209.6	9%	1800.0	193	193.0	0%	467.9
Graph0.200000_75_4	0.2	75	198	198.0	0%	9.2	198	198.0	0%	1236.3	193	230.7	20%	1800.0	198	198.0	0%	422.7
Graph0.200000_100_0	0.2	100	223	223.0	0%	251.5	220	466.3	112%	1800.0	220	298.8	36%	1800.0	223	239.8	8%	1800.0
Graph0.200000_100_1	0.2	100	238	238.0	0%	216.6	210	731.1	248%	1800.0	238	291.8	23%	1800.0	238	268.2	13%	1800.0
Graph0.200000_100_2	0.2	100	255	255.0	0%	312.3	228	578.0	154%	1800.0	255	284.5	12%	1800.0	255	290.2	14%	1800.0
Graph0.200000_100_3	0.2	100	222	222.0	0%	201.9	222	414.7	87%	1800.0	221	301.6	36%	1800.0	222	252.1	14%	1800.0
Graph0.200000_100_4	0.2	100	221	221.0	0%	189.3	206	658.2	219%	1800.0	221	309.0	40%	1800.0	221	251.8	14%	1800.0
Graph0.200000_125_0	0.2	125	230	314.9	37%	1800.0	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.200000_125_1	0.2	125	248	309.1	25%	1800.0	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.200000_125_2	0.2	125	222	283.0	27%	1800.0	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.200000_125_3	0.2	125	239	328.2	37%	1800.0	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.200000_125_4	0.2	125	232	327.4	41%	1800.0	-	-	-	-	-	-	-	-	-	-	-	-
Graph0.500000_25_0	0.5	25	55	55.0	0%	0.3	55	55.0	0%	0.6	55	55.0	0%	0.4	55	55.0	0%	0.9
Graph0.500000_25_1	0.5	25	68	68.0	0%	0.2	68	68.0	0%	0.4	68	68.0	0%	0.4	68	68.0	0%	0.9
Graph0.500000_25_2	0.5	25	70	70.0	0%	0.1	70	70.0	0%	0.5	70	70.0	0%	0.4	70	70.0	0%	0.4
Graph0.500000_25_3	0.5	25	63	63.0	0%	0.2	63	63.0	0%	0.5	63	63.0	0%	0.2	63	63.0	0%	0.4
Graph0.500000_25_4	0.5	25	65	65.0	0%	0.3	65	65.0	0%	0.6	65	65.0	0%	0.2	65	65.0	0%	0.5

Table 8 Detailed results of Table 5

			(VC2011 MWIM)				(MWIM)				Single t + (20)				Multiple t + (21)			
Graph Name	D	$\|V\|$	LB	UB	Gap	Time	LB	UB	Gap	Time	LB	UB	Gap	Time	LB	UB	Gap	Time
Graph0.500000_50_0	0.5	50	94	94.0	0%	5.3	94	94.0	0%	11.1	94	94.0	0%	8.7	94	94.0	0%	3.5
Graph0.500000_50_1	0.5	50	88	88.0	0%	4.6	88	88.0	0%	11.1	88	88.0	0%	7.1	88	88.0	0%	2.7
Graph0.500000_50_2	0.5	50	95	95.0	0%	5.3	95	95.0	0%	11.6	95	95.0	0%	7.0	95	95.0	0%	3.8
Graph0.500000_50_3	0.5	50	86	86.0	0%	4.7	86	86.0	0%	11.8	86	86.0	0%	10.3	86	86.0	0%	5.1
Graph0.500000_50_4	0.5	50	93	93.0	0%	5.1	93	93.0	0%	14.4	93	93.0	0%	7.9	93	93.0	0%	2.4
Graph0.500000_75_0	0.5	75	107	107.0	0%	81.3	107	107.0	0%	74.4	107	107.0	0%	25.2	107	107.0	0%	20.8
Graph0.500000_75_1	0.5	75	95	95.0	0%	69.7	95	95.0	0%	68.4	95	95.0	0%	61.8	95	95.0	0%	26.9
Graph0.500000_75_2	0.5	75	103	103.0	0%	67.0	103	103.0	0%	125.3	103	103.0	0%	26.4	103	103.0	0%	19.9
Graph0.500000_75_3	0.5	75	101	101.0	0%	62.1	101	101.0	0%	90.0	101	101.0	0%	25.0	101	101.0	0%	23.5
Graph0.500000_75_4	0.5	75	95	95.0	0%	74.0	95	95.0	0%	106.8	95	95.0	0%	57.7	95	95.0	0%	25.4
Graph0.500000_100_0	0.5	100	117	117.0	0%	1319.3	117	117.0	0%	1747.6	117	117.0	0%	138.7	117	117.0	0%	63.5
Graph0.500000_100_1	0.5	100	125	125.0	0%	1054.7	125	125.0	0%	1497.2	125	125.0	0%	136.4	125	125.0	0%	44.1
Graph0.500000_100_2	0.5	100	96	96.0	0%	1587.7	96	96.0	0%	1457.6	96	96.0	0%	156.0	96	96.0	0%	78.8
Graph0.500000_100_3	0.5	100	115	115.0	0%	1464.8	115	115.0	0%	1432.3	115	115.0	0%	189.3	115	115.0	0%	52.8
Graph0.500000_100_4	0.5	100	108	108.0	0%	1042.6	108	108.0	0%	1182.8	108	108.0	0%	145.4	108	108.0	0%	58.1
Graph0.500000_125_0	0.5	125	116	295.9	155%	1800.0	116	1,031.9	790%	1800.0	119	119.0	0%	699.5	119	119.0	0%	601.9
Graph0.500000_125_1	0.5	125	117	267.9	129%	1800.0	117	1,002.9	757%	1800.0	117	117.0	0%	569.3	117	117.0	0%	621.7
Graph0.500000_125_2	0.5	125	120	290.2	142%	1800.0	120	930.2	675%	1800.0	122	122.0	0%	473.3	122	122.0	0%	366.7
Graph0.500000_125_3	0.5	125	119	301.3	153%	1800.0	119	1,048.3	781%	1800.0	119	119.0	0%	664.7	119	119.0	0%	458.3
Graph0.500000_125_4	0.5	125	112	241.2	115%	1800.0	112	921.2	723%	1800.0	115	115.0	0%	666.7	115	115.0	0%	519.8
Graph0.500000_150_0	0.5	150	-	-	-	-	-	-	-	-	123	142.4	16%	1800.0	123	136.7	11%	1800.0
Graph0.500000_150_1	0.5	150	-	-	-	-	-	-	-	-	121	188.5	56%	1800.0	121	142.6	18%	1800.0
Graph0.500000_150_2	0.5	150	-	-	-	-	-	-	-	-	118	155.6	32%	1800.0	119	134.3	13%	1800.0
Graph0.500000_150_3	0.5	150	-	-	-	-	-	-	-	-	120	150.8	26%	1800.0	120	140.2	17%	1800.0
Graph0.500000_150_4	0.5	150	-	-	-	-	-	-	-	-	121	147.5	22%	1800.0	121	136.3	13%	1800.0
Graph0.800000_25_0	0.8	25	51	51.0	0%	0.3	51	51.0	0%	0.6	51	51.0	0%	0.5	51	51.0	0%	0.2
Graph0.800000_25_1	0.8	25	40	40.0	0%	0.3	40	40.0	0%	0.8	40	40.0	0%	0.3	40	40.0	0%	0.4
Graph0.800000_25_2	0.8	25	39	39.0	0%	0.2	39	39.0	0%	0.8	39	39.0	0%	0.7	39	39.0	0%	0.7
Graph0.800000_25_3	0.8	25	51	51.0	0%	0.3	51	51.0	0%	0.7	51	51.0	0%	0.4	51	51.0	0%	0.3
Graph0.800000_25_4	0.8	25	49	49.0	0%	0.4	49	49.0	0%	0.7	49	49.0	0%	0.5	49	49.0	0%	0.4
Graph0.800000_50_0	0.8	50	52	52.0	0%	3.2	52	52.0	0%	10.5	52	52.0	0%	3.4	52	52.0	0%	2.0
Graph0.800000_50_1	0.8	50	53	53.0	0%	3.6	53	53.0	0%	10.7	53	53.0	0%	2.6	53	53.0	0%	4.3
Graph0.800000_50_2	0.8	50	50	50.0	0%	2.7	50	50.0	0%	17.0	50	50.0	0%	7.7	50	50.0	0%	6.6
Graph0.800000_50_3	0.8	50	58	58.0	0%	4.0	58	58.0	0%	10.7	58	58.0	0%	3.6	58	58.0	0%	1.9
Graph0.800000_50_4	0.8	50	52	52.0	0%	4.2	52	52.0	0%	11.0	52	52.0	0%	5.1	52	52.0	0%	2.2
Graph0.800000_75_0	0.8	75	55	55.0	0%	45.8	55	55.0	0%	67.5	55	55.0	0%	8.9	55	55.0	0%	5.5
Graph0.800000_75_1	0.8	75	57	57.0	0%	48.4	57	57.0	0%	47.8	57	57.0	0%	15.9	57	57.0	0%	8.2
Graph0.800000_75_2	0.8	75	65	65.0	0%	50.6	65	65.0	0%	45.0	65	65.0	0%	6.9	65	65.0	0%	5.2
Graph0.800000_75_3	0.8	75	57	57.0	0%	41.7	57	57.0	0%	46.9	57	57.0	0%	15.8	57	57.0	0%	7.4
Graph0.800000_75_4	0.8	75	55	55.0	0%	51.5	55	55.0	0%	50.8	55	55.0	0%	16.4	55	55.0	0%	11.7
Graph0.800000_100_0	0.8	100	64	64.0	0%	208.6	64	64.0	0%	213.4	64	64.0	0%	38.3	64	64.0	0%	8.2
Graph0.800000_100_1	0.8	100	67	67.0	0%	193.2	67	67.0	0%	275.3	67	67.0	0%	29.1	67	67.0	0%	11.3
Graph0.800000_100_2	0.8	100	69	69.0	0%	224.1	69	69.0	0%	240.9	69	69.0	0%	18.8	69	69.0	0%	23.3
Graph0.800000_100_3	0.8	100	67	67.0	0%	160.2	67	67.0	0%	260.4	67	67.0	0%	22.2	67	67.0	0%	13.1
Graph0.800000_100_4	0.8	100	67	67.0	0%	213.7	67	67.0	0%	328.5	67	67.0	0%	20.1	67	67.0	0%	8.6
Graph0.800000_125_0	0.8	125	72	72.0	0%	932.3	72	72.0	0%	1015.6	72	72.0	0%	56.6	72	72.0	0%	42.1
Graph0.800000_125_1	0.8	125	79	79.0	0%	791.3	79	79.0	0%	1114.3	79	79.0	0%	81.2	79	79.0	0%	45.2
Graph0.800000_125_2	0.8	125	68	68.0	0%	767.3	68	68.0	0%	736.8	68	68.0	0%	93.7	68	68.0	0%	34.9
Graph0.800000_125_3	0.8	125	70	70.0	0%	1234.2	70	70.0	0%	912.0	70	70.0	0%	111.7	70	70.0	0%	26.1
Graph0.800000_125_4	0.8	125	71	71.0	0%	822.5	71	71.0	0%	895.5	71	71.0	0%	76.3	71	71.0	0%	49.1
Graph0.800000_150_0	0.8	150	75	134.1	79%	1800.0	75	4,623.5	6065%	1800.0	75	75.0	0%	265.6	75	75.0	0%	65.4
Graph0.800000_150_1	0.8	150	73	130.8	79%	1800.0	69	4,487.6	6404%	1800.0	73	73.0	0%	358.5	73	73.0	0%	50.7
Graph0.800000_150_2	0.8	150	70	123.9	77%	1800.0	69	4,208.7	6000%	1800.0	70	70.0	0%	210.5	70	70.0	0%	81.7
Graph0.800000_150_3	0.8	150	74	136.2	84%	1800.0	74	4,669.0	6210%	1800.0	74	74.0	0%	208.6	74	74.0	0%	82.2
Graph0.800000_150_4	0.8	150	74	140.0	89%	1800.0	71	4,304.8	5963%	1800.0	74	74.0	0%	245.3	74	74.0	0%	77.2
Graph0.800000_175_0	0.8	175	-	-	-	-	-	-	-	-	75	75.0	0%	550.5	75	75.0	0%	94.8
Graph0.800000_175_1	0.8	175	-	-	-	-	-	-	-	-	84	84.0	0%	294.6	84	84.0	0%	68.4
Graph0.800000_175_2	0.8	175	-	-	-	-	-	-	-	-	75	75.0	0%	365.5	75	75.0	0%	115.0
Graph0.800000_175_3	0.8	175	-	-	-	-	-	-	-	-	74	74.0	0%	380.2	74	74.0	0%	114.6
Graph0.800000_175_4	0.8	175	-	-	-	-	-	-	-	-	75	75.0	0%	480.7	75	75.0	0%	89.1
Graph0.800000_200_0	0.8	200	-	-	-	-	-	-	-	-	75	75.0	0%	965.4	75	75.0	0%	270.7
Graph0.800000_200_1	0.8	200	-	-	-	-	-	-	-	-	74	74.0	0%	891.7	74	74.0	0%	226.2
Graph0.800000_200_2	0.8	200	-	-	-	-	-	-	-	-	75	75.0	0%	1439.7	75	75.0	0%	242.7
Graph0.800000_200_3	0.8	200	-	-	-	-	-	-	-	-	76	76.0	0%	1407.9	76	76.0	0%	247.0
Graph0.800000_200_4	0.8	200	-	-	-	-	-	-	-	-	77	77.0	0%	1054.3	77	77.0	0%	158.3
Graph0.800000_225_0	0.8	225	-	-	-	-	-	-	-	-	75	131.9	76%	1800.0	75	75.0	0%	446.5
Graph0.800000_225_1	0.8	225	-	-	-	-	-	-	-	-	73	134.3	84%	1800.0	79	79.0	0%	409.8
Graph0.800000_225_2	0.8	225	-	-	-	-	-	-	-	-	73	96.5	32%	1800.0	77	77.0	0%	578.3
Graph0.800000_225_3	0.8	225	-	-	-	-	-	-	-	-	75	126.0	68%	1800.0	78	78.0	0%	371.8
Graph0.800000_225_4	0.8	225	-	-	-	-	-	-	-	-	74	139.9	89%	1800.0	80	80.0	0%	485.1
Graph0.800000_250_0	0.8	250	-	-	-	-	-	-	-	-	-	-	-	-	79	79.0	0%	609.9
Graph0.800000_250_1	0.8	250	-	-	-	-	-	-	-	-	-	-	-	-	81	81.0	0%	650.8
Graph0.800000_250_2	0.8	250	-	-	-	-	-	-	-	-	-	-	-	-	81	81.0	0%	739.0
Graph0.800000_250_3	0.8	250	-	-	-	-	-	-	-	-	-	-	-	-	83	83.0	0%	516.4
Graph0.800000_250_4	0.8	250	-	-	-	-	-	-	-	-	-	-	-	-	82	82.0	0%	486.8
Graph0.800000_300_0	0.8	300	-	-	-	-	-	-	-	-	-	-	-	-	83	91.9	11%	1800.0
Graph0.800000_300_1	0.8	300	-	-	-	-	-	-	-	-	-	-	-	-	84	84.0	0%	1390.0
Graph0.800000_300_2	0.8	300	-	-	-	-	-	-	-	-	-	-	-	-	76	76.0	0%	1428.5
Graph0.800000_300_3	0.8	300	-	-	-	-	-	-	-	-	-	-	-	-	76	96.0	26%	1800.0
Graph0.800000_300_4	0.8	300	-	-	-	-	-	-	-	-	-	-	-	-	76	89.9	18%	1800.0

Appendix B: Automatic Benders decomposition feature of CPLEX 12.7

CPLEX 12.7, which was released while our paper was under second round of peer review, introduced automatic Benders decomposition feature. We conducted an experiment to test the effect of this new feature on problem instances used in Table 5. In this experiment we compared direct solution of (MWIM), our proposed algorithm and automatic Benders decomposition feature of CPLEX's new version applied to (MWIM). The first two sets of columns in Table 9 are taken from Table 5 and show the results for CPLEX 12.6.3 whereas the last set of columns show the results of automatic Benders decomposition algorithm of CPLEX 12.7. We observe that automatic Benders algorithm yields worse results than direct solution of (MWIM). This is consistent with our implementation of Benders decomposition with a single t variable and without any additional valid inequalities (see "Single t" column in Table 3). Our proposed algorithm that contains multiple t-variables in the master problem along with valid inequalities (21), and generates Benders cuts by utilizing Algorithm 1 instead of solving a linear programming problem (set of columns titled "Multiple t + (21)") is faster than other methods.

Table 9 Effect of automatic Benders decomposition feature of CPLEX 12.7

		CPLEX 12.6.3								CPLEX 12.7			
		(MWIM)				Multiple t + (21)				(MWIM) + Automated Benders			
D	$\|V\|$	Solved	Gap	Time	BB Nodes	Solved	Gap	Time	BB Nodes	Solved	Gap	Time	BB Nodes
0.05	25	5	0%	0.3	0	5	0%	0.4	0	5	0%	0.4	0
	50	5	0%	1.3	0	5	0%	1.6	110	5	0%	1.7	191
	75	5	0%	8.9	959	5	0%	4.4	2981	5	0%	12.2	16264
	100	5	0%	878.6	8596	5	0%	118.3	224166	5	0%	1266.3	1568173
	125	0	44%	1800.0	16214	0	12%	1800.0	1055447	0	49%	1800.0	1350254
	150	-	-	-	-	-	-	-	-	-	-	-	-
	175	-	-	-	-	-	-	-	-	-	-	-	-
0.2	25	5	0%	0.4	0	5	0%	0.3	120	5	0%	1.5	245
	50	5	0%	6.7	13917	5	0%	2.5	9963	5	0%	28.1	35283
	75	5	0%	1231.2	55539	5	0%	424.3	371791	0	34%	1800.0	748769
	100	0	164%	1800.0	90722	0	12%	1800.0	5649446	-	-	-	-
	125	-	-	-	-	-	-	-	-	-	-	-	-
0.5	25	5	0%	0.5	0	5	0%	0.6	383	5	0%	2.2	722
	50	5	0%	12	3418	5	0%	3.5	4450	5	0%	25.9	21599
	75	5	0%	93	49988	5	0%	23.3	26959	5	0%	778.5	137426
	100	5	0%	1463.5	83631	5	0%	59.4	82384	0	309%	1800.0	169870
	125	0	745%	1800	153238	5	0%	513.7	113043	-	-	-	-
	150	-	-	-	-	0	14%	1800	107242	-	-	-	-
0.8	25	5	0%	0.7	0	5	0%	0.4	161	5	0%	7.6	576
	50	5	0%	12	420	5	0%	3.4	1400	5	0%	32.6	4382
	75	5	0%	51.6	4255	5	0%	7.6	4580	5	0%	341.3	26340
	100	5	0%	263.7	8783	5	0%	12.9	9434	5	0%	877.3	46588
	125	5	0%	934.8	16193	5	0%	39.5	16973	5	0%	1466.0	93868
	150	0	6128%	1800	17180	5	0%	71.4	32648	0	706%	1800.0	68105
	175	-	-	-	-	5	0%	96.4	57453	-	-	-	-
	200	-	-	-	-	5	0%	229	89962	-	-	-	-
	225	-	-	-	-	5	0%	458.3	103481	-	-	-	-
	250	-	-	-	-	5	0%	600.6	146259	-	-	-	-
	300	-	-	-	-	2	11%	1643.7	154641	-	-	-	-