Noname manuscript No.
(will be inserted by the editor)

The Multi-Hour Bandwidth Packing Problem with Queuing Delays: Bounds and Exact Solution Approach

Navneet Vidyarthi · Arjun Bhardwaj · Niraj Sinha

Received: date / Accepted: date

Abstract The multi-hour bandwidth packing problem arises in telecommunication networks that spans several time horizon. The problem seeks to select and route a set of messages from a given list of messages with prespecified requirement on demand for bandwidth under time varying traffic conditions on an undirected communication network such that the total profit is maximized. The total profit is computed based on the total revenue and the flow cost as well as communication delay cost. Under Poisson call arrival rates and exponential service time distributions on the links, the problem is setup as a network of spatially distributed M/M/1 queues and formulated as a nonlinear integer programming model. Using simple transformation and piecewise linearization, we present a linear mixed integer programming formulation of the model with large number of constraints. We derive lower and upper bounds for the linearized model and present a cutting plane algorithm based exact solution approach that makes successive improvements to the lower and corresponding upper bound as the iteration progresses. The extension of the proposed modelling framework and solution approached to generalized case with Poisson call arrival rates and general service time distributions on the links (M/G/1 case) is also presented. Computational results indicate that the exact method provides optimal solution in reasonable computational times.

Keywords Bandwidth Packing · Telecommunication Networks · Time Varying Demand · Call Routing · Queueing Delays · Congestion · Linearization · Exact Approach

N.K. Vidyarthi
Department of Supply Chain and Business Technology Management
John Molson School of Business, Concordia University
1455 Blvd. De Maisonneuave West, Montreal, H3G 1M8, Canada
Tel.: +001-514-848-2424x2990
Fax: +001-514-848-2824
E-mail: navneet.vidyarthi@gmail.com

A. Bhardwaj
School of Computing and Electrical Engineering
Indian Institute of Technology, Mandi, Himachal Pradesh, India

N. Sinha
Department of Mechanical Engineering
Indian Institute of Technology, Kanpur, Uttar Pradesh, India

1 Introduction

Bandwidth packing problem (BPP) arises in telecommunication networks. Given a set of calls with their bandwidth requirements (demand) and revenue generated over an undirected telecommunication network with its fixed arc/link capacities and costs, the classical BPP seeks to assign calls from a request table to paths such that network capacities are not violated and profit/revenue is maximized. The single-hour BPP was introduced by Cox et al. (1991). Since then, classical BPP and its variants have been addressed by Anderson et al. (1993), Laguna and Glover (1993), Parker and Ryan (1995), Park et al. (1996), Amiri et al. (1999), Rolland et al. (1999), Amiri and Barkhi (2000), Amiri (2005), Villa and Hoffman (2006), Bose (2009), Amiri and Barkhi (2012) and Han et al. (2013) among others.

The classical BPP seeks to maximize profit, and hence its solution probably has a large number of edges whose utilizations closely approaches edge capacities. Higher edge utilization may cause some packets to wait in the queue for a relatively long time before they are processed through the edge, which eventually results in congestion. This makes a network unreliable and degrades the quality of service (Han et al., 2013). To this end, there are two ways to account for congestion in BPP. The first is to incorporate it in the objective function as queueing delay costs and the other is to include it in the constraints with an upper bound (i.e. queuing delay guarantees). Amiri et al. (1999) and Rolland et al. (1999) extended the classical BPP to address network congestion by incorporating queueing delays. While Amiri et al. (1999) deal with queueing delay costs in the objective function, Rolland et al. (1999) incorporate service constraints with an upper limit on the queueing delay in the network. They propose Lagrangean heuristics based solution approach to deal with the models. Amiri and Barkhi (2000) extended the problem further to address the issue of traffic variation during peak and off-peak hours by allowing the demand of a call to vary by the hours of the days. Later, Bose (2009) studied another version of the problem where the calls belong to two priority classes: The calls belonging to the higher priority class are shorter in length, generate more revenue, and are more demand intensive compared to the calls belonging to the lower priority class.

....The multi-hour bandwidth packing problem arises in large telecommunications networks that span several time zones. The amount of traffic for a pair of communicating nodes (a call) usually varies over the hours of the day and peaks during di!erent periods called busy-hours. Failure to take into consideration this fact may lead to signi”cant revenue loss, excessive costs and/or inferior quality of service to users. Indeed, a network operating using average traffic conditions may cost less, but may fail to accommodate demand during the peaks of the di!erent busy-hours resulting in higher response time to users and often may lead to loss of revenues due to the selection of a sub-optimal set of calls to be routed. On the other hand, a network operating using peak traffics will result in under utilization of the network resources and/or revenue loss.

Solution techniques include tabu search (Anderson et al., 1993; Laguna and Glover, 1993), genetic algorithm (Cox et al., 1991), column generation (Parker and Ryan, 1995), Lagrangean heuristic (Rolland et al., 1999; Amiri et al., 1999; Amiri and Barkhi, 2000; Amiri, 2003, 2005; Amiri and Barkhi, 2012), and branch-and-bound technique using column generation and cutting plane approaches (Park et al., 1996; Villa and Hoffman, 2006).

The contribution of this paper lies in the development of bounds and an exact solution approach for solving multi-hour bandwidth packing problem with queueing delay cost. First, we present a model of multi-hour BPP that accounts for congestion due to queueing delays under Piosson call arrival rates and exponential service time distributions on the links. Then we extend the model to deal with more generalized settings with general service time distributions on the links, where the problem is setup as a network of spatially distributed M/G/1 queues and formulated as a nonlinear integer programming (NLIP) model. Secondly, using simple transformation and piecewise linearization of the nonlinear queueing delay function, we present a linear mixed integer programming (MIP) reformulation of the model with large number of constraints. We derive upper and lower bounds for the formulation and propose a cutting plane algorithm based exact and efficient solution approach to solve the model to optimality.

The remainder of the paper is organized as follows. In Section 2, we describe the problem and present a nonlinear IP formulation. Section 3 describes the piecewise linearization scheme and linear MIP reformulation of the problem. The exact solution algorithm is presented in Section 4. Computational results are reported in Sections 5. Section 6 summarizes the contribution with some directions for future research.

2 Problem Formulation

The multi-hour bandwidth packing problem with queueing delay cost (MHBPP-QDC) can be stated as follows: Given a set of calls with bandwidth requirements (demand) and the topology of an undirected telecommunication network with its fixed arc/link capacities, the problem is to select calls and route them through the network, such that the *total profit* (total revenues - flow costs - queueing delay costs) is maximized. Besides capacity restrictions (i.e. steady state conditions) on the links, there are flow conservation and variable linking constraints which ensure that if a call is routed, it must be routed using the links in the given network topology.

To model the problem, we use the following notations:

i, j	:	Indices for nodes in the network
t	:	Index for busy hours
N	:	Set of nodes in the network
E	:	Set of undirected links (i, j) in the network, where $i \in N$, $j \in N$ and $i < j$
M	:	Set of calls, where each call is represented by a connecting node pair
T	:	Set of busy hours;
d^{mt}	:	Demand of call m, $m \in M$ during busy-hour $t \in T$
r^m	:	Revenue from call m, $m \in M$
Q_{ij}	:	Bandwidth capacity of link (i, j)
C^t_{ij}	:	Cost of unit flow on link (i, j) during the busy-hour $t \in T$
D	:	Unit queueing delay cost per unit time

The decision variables are defined as follows:

$$Y^m = \begin{cases} 1 & \text{if call } m \text{ is routed;} \\ 0 & \text{otherwise.} \end{cases}$$

$$X_{ij}^{mt} = \begin{cases} 1 & \text{if call } m \text{ is routed through a path that uses link } (i,j) \text{ (in the direction from } i \text{ to } j\text{) during busy hour } t \in T; \\ 0 & \text{otherwise.} \end{cases}$$

Let i and j be the indices for nodes of a network, denoted by set N, $i, j \in N$, and $i < j$. Let E denote the set of undirected links (i, j) in the network and M be the set of calls, where each call is represented by a communicating node pair. T denotes the set of busy hours;...indexed by t. We use d^{mt} and R^m to denote the bandwidth requirement (or demand) and revenue generated from call $m \in M$ respectively. Let $1/\mu$ is the average message length and Q_{ij} be the bandwidth capacity of the link (i, j).

We assume that the arrival process of messages entering the network is independent and follows a Poisson distribution. We further assume that the propagation delay in the links is negligible, and there is single class of service for each communicating node pair. Every link is assumed to have an infinite buffer to store messages waiting for transmission. If X_{ij}^{mt} is a decision variable that equals 1 if call m is routed through a path that uses a link (i, j) (in the direction from i to j) during busy hour t, then the aggregate demand arrival rate on link (i, j) is also a random variable that follows a Poisson process with mean $\lambda_{ij}^t = \mu \sum_{m \in M} d^m X_{ij}^{mt}$ (due to the superposition of Poisson processes). Assuming that the service times on the link (i, j) follows an exponential distribution, each link can be modelled as an M/M/1 queue with a mean service rate of μQ_{ij}. Under steady state conditions ($\lambda_{ij}^t < \mu Q_{ij}$) and first-come first-serve (FCFS) queuing discipline, the *mean sojourn time* (waiting time in queue + service time) of a message on link (i, j) in a M/M/1 queue, denoted by $E[w_{ij}^t]$, is given by $E[w_{ij}^t] = \frac{\lambda_{ij}^t}{\mu Q_{ij}(\mu Q_{ij} - \lambda_{ij}^t)} + \frac{1}{\mu Q_{ij}}$. The average end-to-end queueing delay in the network can be estimated as the weighted sum of the expected delays on the links in the network as follows:

$$\sum_{(i,j)\in E} \lambda_{ij}^t E[w_{ij}^t] = \sum_{(i,j)\in E} \left\{ \frac{(\lambda_{ij}^t)^2}{\mu Q_{ij}(\mu Q_{ij} - \lambda_{ij}^t)} + \frac{\lambda_{ij}^t}{\mu Q_{ij}} \right\} \tag{1}$$

If D denotes the unit queueing delay cost per unit time, then the *total queueing delay/congestion cost* is given by $D \sum_{(i,j)\in E} \sum_{t \in T} \lambda_{ij}^t E[w_{ij}^t]$ and can be expressed as:

$$DC(\mathbf{X}) = D \sum_{(i,j)\in E} \sum_{t\in T} \frac{\mu \sum_{m\in M} d^{mt}(X_{ij}^{mt} + X_{ji}^{mt})}{\mu Q_{ij} - \mu \sum_{m\in M} d^{mt}(X_{ij}^{mt} + X_{ji}^{mt})} \tag{2}$$

MBPP-QD is to select calls and route them through the network, such that the total profit (total revenues - total flow costs - total queueing delay costs) is maximized. Besides capacity restrictions (i.e. steady state conditions) on the links, there are flow conservation and variable linking constraints which ensure that if a call is routed, it must be routed using the links in the given network topology. The resulting nonlinear integer programming

(NLIP) formulation is:

$$[P]: \ \max \sum_{m\in M} r^m Y^m - \sum_{m\in M}\sum_{(i,j)\in E}\sum_{t\in T} C^t_{ij} d^{mt}(X^{mt}_{ij} + X^{mt}_{ji}) - D\sum_{(i,j)\in E}\sum_{t\in T}\left\{\frac{\sum_{m\in M} d^m(X^{mt}_{ij}+X^{mt}_{ji})}{Q_{ij}-\sum_{m\in M} d^m(X^{mt}_{ij}+X^{mt}_{ji})}\right\} \quad (3)$$

$$\text{s.t.} \sum_{j\in N} X^{mt}_{ij} - \sum_{j\in N} X^{mt}_{ji} = \begin{cases} Y^m & \text{if } i = O(m); \\ -Y^m & \text{if } i = D(m); \forall i \in N, m\in M, t\in T \\ 0 & \text{otherwise}\end{cases} \quad (4)$$

$$\sum_{m\in M} d^{mt}(X^{mt}_{ij}+X^{mt}_{ji}) \le Q_{ij} \qquad \forall (i,j)\in E, t\in T \quad (5)$$

$$X^{mt}_{ij} \in \{0,1\}; \quad Y^m \in \{0,1\} \qquad \forall (i,j)\in E, m\in M, t\in T \quad (6)$$

The objective function (3) represents the total profit of routed calls. The first term is the total revenue, the second term is the total flow costs, whereas the third term is the total queueing delay costs. Constraint set (4) contains the flow conservation equations which define a route (path) for each routed call represented by a communicating node pair during every busy-hour. Constraints set (5) represents the capacity constraints on the link during every busy-hour. Constraints set (6) enforce the integrality conditions on the decision variables.

The nonlinearity in the formulation $[P]$ arises due to the expression for the total end-to-end queueing delay cost in the system. Furthermore, the presence of binary variables X^{mt}_{ij} (and X^{mt}_{ji}) complicates the objective function and makes it difficult to solve the model to optimality. In the next section, we linearize the expression for total queueing delay and present a linear MIP reformulation that is amenable to a cutting plane algorithm based exact approach.

3 Model Linearization

We define non-negative auxiliary variables W^t_{ij}, such that:

$$W^t_{ij} = \frac{\lambda^t_{ij}}{\mu Q_{ij} - \lambda^t_{ij}} = \frac{\sum_{m\in M} d^{mt}(X^{mt}_{ij}+X^{mt}_{ji})}{Q_{ij}-\sum_{m\in M} d^m(X^{mt}_{ij}+X^{mt}_{ji})} \quad (7)$$

This implies,

$$\sum_{m\in M} d^{mt}(X^{mt}_{ij}+X^{mt}_{ji}) = \frac{W^t_{ij}}{1+W^t_{ij}} Q_{ij} \quad (8)$$

Substituting (7) in the expression for congestion cost (2) above gives:

$$QDC(\mathbf{X}) = D\sum_{(i,j)\in E}\sum_{t\in T}\frac{\mu\sum_{m\in M} d^m(X^m_{ij}+X^m_{ji})}{\mu Q_{ij} - \mu\sum_{m\in M} d^m(X^m_{ij}+X^m_{ji})} = D\sum_{(i,j)\in E}\sum_{t\in T} W^t_{ij}$$

Differentiating the function $f(W_{ij}^t) = \frac{W_{ij}^t}{1+W_{ij}^t}$ w.r.t. W_{ij}^t, we get the first derivative $\frac{\delta f}{\delta W_{ij}^t} = \frac{1}{(1+W_{ij}^t)^2} > 0$, and the second derivative $\frac{\delta^2 f}{\delta (W_{ij}^t)2} = \frac{-2}{(1+W_{ij}^t)^3} < 0$. This implies that the function $f(W_{ij}^t) = \frac{W_{ij}^t}{1+W_{ij}^t}$ is concave in $W_{ij}^t \in [0, \infty)$.

Let the domain H of the auxiliary variable W_{ij} is a set of indices of points $\{R_j^h\}_{h\in H}$, at which the function $\rho_j(W_{ij}) = W_{ij}/(1+W_{ij})$ can be approximated arbitrary closely by a set of piecewise linear functions that are tangent to ρ_j. This implies that $\rho_j(R_j) = R_j/(1+R_j)$ can be expressed as the finite minimum of linearizations of ρ_j at a given set of point $\{R_j^h\}_{h\in H}$ as follows:

$$\rho_j = \min_{h\in H}\left\{\frac{1}{(1+R_j^h)^2}R_j + \frac{(R_j^h)^2}{(1+R_j^h)^2}\right\} \tag{9}$$

Hence, it can be approximated by a large set of piecewise linear functions that are tangent to $f(W_{ij}^t)$ at points $\{W_{ij}^{ht}\}_{h\in H}$, such that:

$$\frac{W_{ij}^t}{1+W_{ij}^t} = \min_{h\in H}\left\{\frac{1}{(1+W_{ij}^{ht})^2}W_{ij}^t + \frac{(W_{ij}^{ht})^2}{(1+W_{ij}^{ht})^2}\right\}$$

This is equivalent to the following set of constraints:

$$\frac{W_{ij}^t}{1+W_{ij}^t} \le \frac{1}{(1+W_{ij}^{ht})^2}W_{ij}^t + \frac{(W_{ij}^{ht})^2}{(1+W_{ij}^{ht})^2}, \qquad \forall (i,j)\in E, h\in H$$

Using (8), the above set of constraints can be rewritten as:

$$\sum_{m\in M} d^m(X_{ij}^m + X_{ji}^m) \le \frac{Q_{ij}}{(1+W_{ij}^{ht})^2}W_{ij}^t + \frac{Q_{ij}(W_{ij}^{ht})^2}{(1+W_{ij}^{ht})^2} \qquad \forall (i,j)\in E, t\in T, h\in H \tag{10}$$

$$\text{or} \sum_{m\in M} d^{mt}(1+W_{ij}^{ht})^2(X_{ij}^{mt} + X_{ji}^{mt}) - Q_{ij}W_{ij}^t \le Q_{ij}(W_{ij}^{ht})^2 \qquad \forall (i,j)\in E, t\in T, h\in H \tag{11}$$

The resulting linear MIP formulation is:

$$[PL(H)]: \max_{X,Y,W} \sum_{m\in M} r^m Y^m - \sum_{m\in M}\sum_{(i,j)\in E}\sum_{t\in T} C_{ij}^t d^{mt}(X_{ij}^{mt} + X_{ji}^{mt}) - D\sum_{(i,j)\in E}\sum_{t\in T} W_{ij}^t \tag{12}$$

$$\text{s.t. } (4)-(5)$$

$$\sum_{m\in M} d^{mt}(1+W_{ij}^{ht})^2(X_{ij}^{mt} + X_{ji}^{mt}) - Q_{ij}W_{ij}^t \le Q_{ij}(W_{ij}^{ht})^2 \quad \forall j, t, h\in H \tag{13}$$

$$X_{ij}^{mt},\ Y^m \in \{0,1\} \qquad \forall (i,j)\in E, m\in M, t\in T \tag{14}$$

$$W_{ij}^t \ge 0 \qquad \forall (i,j)\in E, t\in T \tag{15}$$

4 Bounds and Solution Method

The proposed solution approach relies on obtaining good lower and upper bounds for the linear model $[PL(H)]$. The algorithm makes successive improvements to the lower bound and the corresponding upper bound as the iterations progress. Below, we present lower and upper bounds that is used in the proposed solution approach.

Proposition 1: *For every given subset of points $\{W_{ij}^{ht}\}_{h \in H^q \subseteq H}$, the optimal objective function value of the problem $[PL(H^q)]$ is an upper bound to $[PL(H)]$* or *P*.

Note that the linearized model $[PL(H)]$ contains all the original variables $(\mathbf{X}, \mathbf{Y})$ and constraints (4)-(5) of model $[P]$ in addition to variables $(\mathbf{W})$ and constraints (13) as a result of linearization. For an infinite set of points in H, model $[PL(H)]$ and nonlinear model $[P]$ are equivalent, because the feasible region of $[PL(H)]$ is same as that of the $[P]$ as constraints (13) would be equivalent to constraints (7). However, for a finite set of points in $H^q \subset H$, the feasible region of the linear model $PL(H^p)$ is larger than the feasible region of $[P(H)]$ or $[P]$. Hence, any feasible solution(s) of $PL(H^q)$ will also be feasible to $[P]$ as it satisfies constraints (4)-(5). Therefore, for a finite set of points in $H^q \subset H$, $[PL(H^q)]$ is a relaxation of the full problem $[PL(H)]$. Solution of a relaxation of a maximization problem provides an upper bound on the objective function value of the original problem and hence the optimal solution of $[PL(H^q)]$ will be a upper bound on the optimal objective function value of $[PL(H)]$ or P.

.... Suppose, at any iteration, we use a subset of tangent points $\{W_{ij}^{ht}\}_{h \in H^q \subseteq H}$, and solve the corresponding problem $[PL(H^q)]$, which yields the solution $(\mathbf{X}^q, \mathbf{Y}^q, \mathbf{W}^q)$ with the objective function value $v(PL(H^q))$. Since $[PL(H^q)]$ is a relaxation of the full problem $[PL(H)]$, $v(PL(H^q)) \geq v(PL(H))$, and hence $v(PL(H^q))$ provides an upper bound, given by:

$$UB = v(PL(H^q)) = \sum_{m \in M} r^m Y^{mq} - \sum_{m \in M} \sum_{(i,j) \in E} \sum_{t \in T} C_{ij}^t d^{mt} (X_{ij}^{mtq} + X_{ji}^{mtq}) - D \sum_{(i,j) \in E} \sum_{t \in T} W_{ij}^{tq} \tag{16}$$

Therefore, for every subset of points $(W_{ij}^{ht})_{H^q \subset H}$, $v(PL(H^q)) \geq v(PL(H))$ or $v(P)$, and hence is an upper bound to $[PL(H)]$ or $[P]$.

Proposition 2: *For any subset of points $\{W_{ij}^{h}\}_{h \in H^q \subseteq H}$, the objective function of P evaluated at the optimal solution of $PL(H^q)$ provides a lower bound to $PL(H)$ or $[P]$.*

As stated above, the optimal solution $(\mathbf{X}^q, \mathbf{Y}^q)$ of $[PL(H^q)]$ is always feasible to the nonlinear model $[P]$ as it satisfies all the constraints (4)-(5) of model $[P]$. And a feasible solution to a maximization problem provides an lower bound on its optimal objective function value. Hence, we can get an lower bound on the optimal objective function value of $[PL(H)]$ or $[P]$ by computing the objective function value of $[P]$ using the optimal solution $(\mathbf{X}^q, \mathbf{Y}^q)$ of $[PL(H^q)]$.

Consider iteration q, where we use a subset of tangent points $(W_{ij}^{h})_{H^q \subset H}$ and solve the corresponding relaxed problem $[L(H^q)]$. Because the optimal solution $(\mathbf{X}^q, \mathbf{Y}^q, \mathbf{W}^q)$ of $[L(H^q)]$ is a feasible solution to $[P]$, the objective function of $[P]$ evaluated at the optimal solution of $L(H^q)$ provides a lower bound to $L(H)$ or $[P]$, given by

$$LB = Z(\mathbf{X^q}, \mathbf{Y^q}) = \sum_{m\in M} r^m Y^{mq} - \sum_{m\in M}\sum_{(i,j)\in E}\sum_{t\in T} C^t_{ij} d^{mt}(X^{mtq}_{ij} + X^{mtq}_{ji}) - D\sum_{(i,j)\in E}\left\{\frac{(\sum_{m\in M} d^{mt}(X^{mtq}_{ij} + X^{mtq}_{ji}))^2}{(Q_{ij} - \sum_{m\in M} d^{mt}(X^{mtq}_{ij} + X^{mtq}_{ji})}\right\} \tag{17}$$

4.1 Algorithm

The algorithm presented below makes successive improvements to the upper and lower bounds computed using the equations (14) and (15) respectively. At every iteration, a relaxed version of the linear model $[L(H)]$ is solved to obtain an optimal solution, an upper bound and a lower bound. This solution is used to generate a set of "cuts / constraints" that eliminate the best solution found so far and improve the upper bound on the remaining solutions. The procedure terminates when the gap between the current upper bound and the best lower bound is within the tolerance limits.

The algorithm starts with an initial subset $H^q \subset H$. The resulting model $[L(H^q)]$ is solved and the upper bound (UB^q) and the lower bound (LB^q) are computed using propositions 1 and 2, respectively. If the upper bound (UB^q) equals the best known lower bound (LB^q) within accepted tolerance (ϵ) at any given iteration q, then $(\mathbf{X}^q, \mathbf{Y}^q, \mathbf{W}^q)$ is an optimal solution to $[P]$ and the algorithm is terminated. Otherwise, a new set of candidate points $U^{h_{new}}_{ij}$ is generated using the current solution $(\mathbf{X}^q)$ as follows:

$$W^{th_{new}}_{ij} = \frac{\sum_{m\in M} d^{mt}(X^{tmq}_{ij} + X^{tmq}_{ji})}{Q_{ij} - \sum_{m\in M} d^{mt}(X^{tmq}_{ij} + X^{tmq}_{ji})} \tag{18}$$

This new set of points is appended to $(W^h_{ij})_{H^q\subset H}$ and the procedure is repeated again, until the stopping criteria is reached. The algorithm is outlined below.

Algorithm 1 Solution Algorithm for $[PL(H)]$

$q \leftarrow 1; UB^{q-1} \leftarrow +\infty; LB^{q-1} \leftarrow -\infty;$
Choose an initial set of points $\{W^h_{ij}\}_{th\in H^q}$ to approximate $W^t_{ij}/(1 + W^t_{ij})$ $\forall (i,j) \in E$.
while $(UB^{q-1} - LB^{q-1})/UB^{q-1} > \epsilon$ **do**
  Solve $[PL(H^q)]$ to obtain $(\mathbf{X}^q, \mathbf{Y}^q, \mathbf{R}^q)$.
  Update the upper bound: $UB^q \leftarrow v(PL(H^q))$.
  Update the lower bound: $LB^q \leftarrow \max\{LB^{q-1}, Z(\mathbf{X}^q, \mathbf{Y}^q)\}$.
  Compute new points: $W^{th_{new}}_{ij} = \frac{\sum_{m\in M} d^{mt}(X^{tmq}_{ij}+X^{tmq}_{ji})}{Q_{ij}-\sum_{m\in M} d^{mt}(X^{tmq}_{ij}+X^{tmq}_{ji})}$ $\forall (i,j) \in E$
  $H^{q+1} \leftarrow H^q \cup \{h_{new}\}$
  $q \leftarrow q + 1$
end while

Proposition 3: *The proposed algorithm is finite.*

Since X_{ij}^m is binary and $U_{ij} = \frac{\sum_{m\in M} d^m(X_{ij}^m+X_{ji}^m)}{Q_{ij}-\sum_{m\in M} d^m(X_{ij}^m+X_{ji}^m)}$, the number of values that U_{ij} can take is finite. Therefore, in order to prove that the algorithm is finite, it is sufficient to prove that the generated values of U_{ij}^h are not repeated. For that, consider an iteration q, where the algorithm has not yet converged, that is, $UB^q > LB^q$. Further, suppose $(\mathbf{X}^q, \mathbf{Y}^q)$ is the solution to $[L(H^q)]$. Then, the new points $U_{ij}^{h_{new}}$ generated at iteration q are given by

$$U_{ij}^{h_{new}} = \frac{\sum_{m\in M} d^m(X_{ij}^{mq} + X_{ji}^{mq})}{Q_{ij} - \sum_{m\in M} d^m(X_{ij}^{mq} + X_{ji}^{mq})} \qquad \forall (i,j) \in E$$

Suppose the values of $U_{ij}^{h_{new}}$ were already generated in one of the earlier iterations $\forall (i,j) \in E$. Then

$$\begin{aligned}(8) \Leftrightarrow \;& \frac{U_{ij}^{h_{new}}}{1+U_{ij}^{h_{new}}} \leq \frac{1}{1+U_{ij}^{h_{new}}} U_{ij}^q + \frac{(U_{ij}^{h_{new}})^2}{(1+U_{ij}^{h_{new}})^2} \\ \Rightarrow \;& U_{ij}^{h_{new}} \leq U_{ij}^q\end{aligned}$$

We now have:

$$\begin{aligned} UB^q &= \sum_{m\in M} R^m Y^{mq} - C \sum_{(i,j)\in E} U_{ij}^q \\ &\leq \sum_{m\in M} R^m Y^{mq} - C \sum_{(i,j)\in E} U_{ij}^{h_{new}} \\ &= \sum_{m\in M} R^m Y^{mq} - C \sum_{(i,j)\in E} \frac{\sum_{m\in M} d^m(X_{ij}^{mq} + X_{ji}^{mq})}{Q_{ij} - \sum_{m\in M} d^m(X_{ij}^{mq} + X_{ji}^{mq})} \\ &\leq \max\Bigg[LB^q, \sum_{m\in M} R^m Y^{mq} - \\ &\qquad C \sum_{(i,j)\in E} \frac{\sum_{m\in M} d^m(X_{ij}^{mq} + X_{ji}^{mq})}{Q_{ij} - \sum_{m\in M} d^m(X_{ij}^{mq} + X_{ji}^{mq})}\Bigg] = LB^q \end{aligned}$$

This contradicts our initial assumption $UB^q > LB^q$. Therefore, at any given iteration, at least one of the values of U_{ij}^h generated is different from all the previously generated values. Since, the number of values that U_{ij}^h can take is finite, and hence the algorithm terminates in a finite number of iterations.

5 Extension to Systems with General Service Time Distributions

Assuming that the service times on the link (i,j) follows an general distribution, each link can be modelled as an M/G/1 queue with a mean service rate of μQ_{ij}. Under steady state conditions ($\lambda_{ij} < \mu Q_{ij}$) and first-come first-serve (FCFS) queuing discipline, the *mean sojourn time* (waiting time in queue + service time) of a message on link (i,j) in a M/G/1 queue, denoted by $E[w_{ij}]$, is given by $E[w_{ij}] = \left(\frac{1+cv^2}{2}\right) \frac{\lambda_{ij}}{\mu Q_{ij}(\mu Q_{ij}-\lambda_{ij})} + \frac{1}{\mu Q_{ij}}$. The average end-to-end queueing delay in the network can be estimated as the weighted sum of the expected delays on the links in the network as follows:

$$\sum_{(i,j)\in E} \lambda_{ij} E[w_{ij}] = \sum_{(i,j)\in E} \left\{ \left(\frac{1+cv^2}{2}\right) \frac{\lambda_{ij}^2}{\mu Q_{ij}(\mu Q_{ij} - \lambda_{ij})} + \frac{\lambda_{ij}}{\mu Q_{ij}} \right\} \qquad (19)$$

If D denotes the unit queueing delay cost per unit time, then the *total queueing delay/congestion cost* is given by $D\sum_{(i,j)\in E}\lambda_{ij}E[w_{ij}]$ and can be expressed as:

$$DC(\mathbf{X}) = D\sum_{(i,j)\in E}\sum_{t\in T}\left\{\left(\frac{1+cv^2}{2}\right)\frac{\sum_{m\in M}d^m(X_{ij}^{mt}+X_{ji}^{mt})}{Q_{ij}(Q_{ij}-\sum_{m\in M}d^m(X_{ij}^{mt}+X_{ji}^{mt}))}+\frac{\sum_{m\in M}d^m(X_{ij}^{mt}+X_{ji}^{mt})}{Q_{ij}}\right\} \tag{20}$$

The resulting nonlinear integer programming formulation is:

$$[P]:\ \max\sum_{m\in M}r^mY^m-\sum_{m\in M}\sum_{(i,j)\in E}\sum_{t\in T}C_{ij}^td^{mt}(X_{ij}^{mt}+X_{ji}^{mt})-$$
$$D\sum_{(i,j)\in E}\sum_{t\in T}\left\{\left(\frac{1+cv^2}{2}\right)\frac{\sum_{m\in M}d^m(X_{ij}^{mt}+X_{ji}^{mt})}{Q_{ij}(Q_{ij}-\sum_{m\in M}d^m(X_{ij}^{mt}+X_{ji}^{mt}))}+\frac{\sum_{m\in M}d^m(X_{ij}^{mt}+X_{ji}^{mt})}{Q_{ij}}\right\} \tag{21}$$

In order to linearize, we rearrange the terms in (**??**), $E[W]$ can be rewritten as:

$$\begin{aligned}E[W]=&\frac{1}{\Lambda}\sum_{(i,j)\in E}\frac{1}{2}\left\{\left(1+cv^2\right)\frac{\lambda_{ij}}{\mu Q_{ij}-\lambda_{ij}}+\left(1-cv^2\right)\frac{\lambda_{ij}}{\mu Q_{ij}}\right\}\\ =&\frac{1}{\Lambda}\sum_{(i,j)\in E}\frac{1}{2}\left\{\left(1+cv^2\right)\frac{\sum_{m\in M}d^m(X_{ij}^m+X_{ji}^m)}{Q_{ij}-\sum_{m\in M}d^m(X_{ij}^m+X_{ji}^m)}+\left(1-cv^2\right)\frac{\sum_{m\in M}d^m(X_{ij}^m+X_{ji}^m)}{Q_{ij}}\right\}\end{aligned}$$

Using the non-negative auxiliary variables W_{ij}^t defined above, we can write the expression for $E[W]$ as follows:

$$E[W]=\frac{1}{\Lambda}\sum_{(i,j)\in E}\frac{1}{2}\left\{\left(1+cv^2\right)W_{ij}+\left(1-cv^2\right)\frac{\sum_{m\in M}d^m(X_{ij}^m+X_{ji}^m)}{Q_{ij}}\right\}$$

Using (8), the above set of constraints can be rewritten as:

$$\sum_{m\in M}d^m(X_{ij}^m+X_{ji}^m)-\frac{Q_{ij}}{(1+R_{ij}^h)^2}W_{ij}\leq\frac{Q_{ij}(W_{ij}^h)^2}{(1+W_{ij}^h)^2}\qquad\forall(i,j)\in E,h\in H \tag{22}$$

provided $\exists h\in H$ such that (22) holds with equality.

The above substitutions result in the following linear MIP model:

$$[PL(H)]:$$
$$\max\sum_{m\in M}r^mY^m-\sum_{m\in M}\sum_{(i,j)\in E}\sum_{t\in T}C_{ij}^td^{mt}(X_{ij}^{mt}+X_{ji}^{mt})-$$
$$\frac{D}{2}\sum_{(i,j)\in E}\sum_{t\in T}\left\{\left(1+cv^2\right)W_{ij}^t+\left(1-cv^2\right)\frac{\sum_{m\in M}d^m(X_{ij}^{mt}+X_{ji}^{mt})}{Q_{ij}}\right\} \tag{23}$$

$$\text{s.t. } (\mathbf{??})-(\mathbf{??}),(22)$$
$$W_{ij}^t\geq 0\qquad\forall(i,j)\in E \tag{24}$$

6 Computational Experiments

We report on the performance of the proposed exact solution approach. The algorithm was coded in OPL studio and the model $[L(H^q)]$ was solved using IBM Ilog CPLEX 12.4. The experiments were conducted on a Dell machine (Precision T5600) with Intel Xeon CPU ES-2650, 2.00 GHz CPU; 16 GB RAM, Windows 7P 64 bits OS.

6.1 Test Instances

The proposed approach was tested on randomly generated networks; the generation scheme of the random networks was adopted from Amiri and Barkhi (2000) and is briefly described as follows: First, the generator locates the specified number of nodes on a 100×100 grid. Each node has a degree equal to 2, 3 or 4 with probability of 0.6, 0.3 and 0.1, respectively. We repeat the following procedure for each node i. Determine node i's closest neighbor (in terms of Euclidean distance) with unsatisfied degree requirement, call this node j. Add arc (i, j) and repeat this until first, node i's degree requirement is satisfied or second, all the nodes with unsatisfied degree requirements have been considered. In the latter case, connect node i to its closest neighbors to which it is not already connected until the degree requirement of node i is satisfied. At the end, check to see if the network is connected; if not, add links necessary to make it connected. The other parameters are generated as follows:

- *Demand:* The traffic requirements during every busy-hour or each call are generated randomly as $d^{mt} \sim TRM * U(20, 40)$, where TRM, called the traffic requirement multiplier, is a constant that varies between 0.50 and 1.50 with increment of 0.25 (this constant is equal to 1 for the base case).
- *Revenue*: The revenue for each call is generated randomly from a uniform distribution as $r^m \sim U(500, 1500)$.
- *Flow Cost*: The unit flow cost for each link (i, j) during busy-hour t is determined randomly as $C^t_{ij} = FCM * U(2, 6)$, where FCM, called the flow cost multiplier, is a constant that varies between 0.50 and 1.50 with increment of 0.25 (this constant is equal to 1 for the base case).
- *Delay Cost*: The unit queuing delay cost is set to $D = \{1, 25, 70, 130, 170\}$. Note that we conduct preliminary sensitivity analysis to select the values of unit delay cost to reflects different penalties associated with congestion.

Furthermore, in the implementation of the algorithm, we start with an initial set of cuts for the function $f(W) = W/(1+W)$. These cuts are generated based on the piecewise linear approximation $\widehat{f}(W)$ of the function $f(W)$ such that the approximation error (measured by $\widehat{f}(W) - f(W)$) is at most ϵ (Elhedhli, 2005). This is in part motivated by our initial computational results, which show that the option of starting with a carefully chosen initial set of cuts improves the performance of the solution approach. Hence, in all the test problems, we use 32 cuts which corresponds to an approximation error of $\epsilon = 0.001$.

We generated 10 problem instances for every combination of values of $|N| = \{10, 20, 30, 40\}$, $P = \{0.5, 0.6, 0.7, 0.8, 0.9\}$, $D = \{1, 25, 70, 130, 170\}$ and $cv = \{0.5, 1.0, 1.5\}$. This results in a total of $10 \times 4 \times 5 \times 5 \times 3 = 3000$ problem instances for the base case scenarios (Table 2 and Table 3). The results are reported in Tables 2 and 3.

In order to analyze the effect of traffic requirement multiplier (TRM), we generated 10 problem instances for every combination of values of $|N| = \{20, 30, 40\}$, $D = \{1, 25, 70,$

130, 170}, TRM = {0.5, 0.75, 1.25, 1.5, 2 } and cv = {0.5, 1.0, 1.5}. The other values are set to $P = 0.6$ and $FCM = 1$. The results are reported in Table 4.

To analyze the effect of flow cost multiplier, we generated 10 problem instances for every combination of values of $|N|$ = {20, 30, 40 }, D = {1, 25, 70, 130, 170}, CM = {0.5, 0.75, 1.25, 1.5, 2 } and cv = {0.5, 1.0, 1.5}. The other values are set to $P = 0.6$ and $TRM = 1$. The results are reported in Table 5 .

6.2 Analysis of Results

Table 2-4 display the number of nodes ($|N|$), average number of edges ($|E|$), percentage of calls routed (P), the unit delay cost (D), the total profit, the total revenue, the total flow cost, the total queueing delay cost, the average (Avg. Util.) and the maximum (Max. Util.) link utilizations, the number of iterations of the algorithm (ITR.), and CPU time in seconds (CPU).

The stopping criteria for the algorithm is $(UB - LB) \leq 10^{-4}$. Each row in Table 1 reports the average results of 10 problem instances. The effect of changes in the unit delay costs can be seen in Table 1. With an increase in the unit delay cost, the percentage of total delay costs become more significant with respect to the total revenue. As the unit delay cost increases, the total revenue generated from the routed calls decreases since fewer calls are selected for routing in order to minimize overall congestion in the network. This is also depicted by the decrease in the average as well as the maximum link utilizations. For example, for the 50 node networks with $P = 0.5$, average and the maximum link utilizations decrease from 100% to 70% and 60% to 30%.

The algorithm succeeds in finding optimal solutions to the problem instances within reasonable computation time. For the problem instances, the CPU times vary from 0.3 secs (for $N = 10$, $P = 0.6$, $C = 170$; $N = 10$, $P = 0.5$, $C = 130$ and $N = 10$, $P = 0.5$, $C = 170$) to 1815 secs (for $N = 50$, $P = 0.5$, $C = 1$), with an average of 165 secs. It is worthwhile noting that as the unit delay cost increases, the CPU time decreases. This trend is more prominent as the number of nodes increases. For example, refer to the problem with 50 nodes and 90% of the calls. As C increases from 1 to 25 to 70 to 130 to 170, the CPU time decreases from 1691 to 763 to 561 to 377 to 320 secs. This is because the problem becomes easier to solve as fewer calls are selected for routing in an attempt to minimize congestion in the network. It is also interesting to note that the average number of iterations of the algorithm is not affected by the change in the parameters of the problem. It ranges from 1.7 (for $N = 10$, $P = 0.6$, $C = 170$) to 2.4 (for $N = 40$, $P = 0.8$, $C = 25$ and $N = 40$, $P = 0.5$, $C = 25$). This indicates that only a small fraction of the large number of constraints in $L(H)$ is generated (and required) to find the optimal solution. Thus, while our model captures the tradeoffs between the total revenue and congestion due to queueing delays, the computational results demonstrate the efficiency of the exact solution approach in providing optimal solutions over a wide range of problem instances.

Table 1 Effect of a-priori cuts on the solution method

				$cv = 0.5$						$cv = 1.5$					
					w/o a-priori cuts		with a-priori cuts		% red.		w/o a-priori cuts		with a-priori cuts		
N	E	P	D	Profit	#itr	cpu	#itr	cpu	cpu	Profit	#itr	cpu	#itr	cpu	%red.
30	38	0.6	1	25,098	3.1	28	2	21	26	24,642	3.7	47	2	26	45
			25	23,668	4.8	67	2	29	57	24,261	5.0	75	2	19	74
			70	22,784	4.8	57	2	16	72	19,994	5.2	51	2	16	68
			130	19,865	5.0	45	2	17	62	18,429	5.2	51	2	27	47
			170	18,454	5.0	35	2	12	67	18,064	5.1	39	2	23	40
		0.7	1	32,074	3.1	46	2	31	34	31,124	3.1	58	2	30	49
			25	28,015	4.3	59	2	32	46	29,100	4.9	83	2	30	63
			70	23,851	4.9	81	2	23	72	23,239	5.9	83	2	27	67
			130	22,986	5.4	69	2	17	76	21,746	5.0	67	2	30	55
			170	21,471	4.8	52	2	25	51	20,768	5.0	63	2	22	65
		0.8	1	35,244	3.0	55	2	40	28	33,786	3.4	71	2	42	41
			25	30,031	4.9	109	2	43	61	28,319	5.4	120	2	33	72
			70	27,693	4.7	73	2	34	53	26,190	5.7	104	2	31	71
			130	25,861	5.1	85	2	21	75	24,495	5.6	111	2	43	62
			170	23,794	5.0	72	2	27	63	20,197	5.4	88	2	23	74
		0.9	1	38,921	3.7	114	2	51	55	35,996	3.4	125	2	51	59
			25	33,327	5.5	181	2	36	80	34,704	5.6	218	2	57	74
			70	31,751	5.1	139	2	29	79	32,952	5.8	140	2	45	68
			130	25,846	5.3	107	2	40	62	27,763	5.4	106	2	45	58
			170	27,814	5.4	75	2	51	32	25,610	5.9	102	2	35	65
			avg.		4.6	77	2	30	58		5.0	90	2	33	61
			max		5.5	181	2	51	80		5.9	218	2	57	74

6.2.1 Effect of Adding a-priori Cuts on the Performance of the Algorithm

6.2.2 Effect of Changes in the Percentage of Number of Available Calls

6.2.3 Effect of Changes in the Link Cost Multiplier

6.2.4 Effect of Changes in the Traffic Requirement Multiplier

				cv = 0.5								cv = 1								cv = 1.5							
N	E	P	D	Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU
10	13	0.5	1	5,979	14,012	57	0.3	0.21	0.71	2	0.3	6,338	13,926	54	0.2	0.25	0.73	2	0.1	6,303	13,587	53	0	0.21	0.75	2	0.5
			25	5,888	14,095	56	2.6	0.21	0.58	2	0.4	6,020	13,116	51	2.6	0.18	0.63	2	0.3	4,945	11,982	56	3	0.18	0.58	2	0.3
			70	5,147	12,079	52	5.8	0.16	0.55	2	0.3	6,348	13,322	47	5.6	0.16	0.51	2	0.6	5,796	12,475	48	6	0.15	0.53	2	0.5
			130	4,974	11,535	48	9.1	0.14	0.49	2	0.7	6,203	13,538	47	7.5	0.13	0.46	2	0.5	4,616	11,028	51	7	0.10	0.32	2	0.5
			170	4,482	11,091	48	11.7	0.14	0.49	2	0.3	3,435	8,510	49	10.4	0.10	0.40	2	0.3	4,115	10,795	49	13	0.13	0.40	2	0.5
		0.6	1	6,652	14,566	54	0.3	0.26	0.78	2	0.4	7,804	17,436	55	0.3	0.27	0.82	2	0.6	7,655	16,675	54	0	0.26	0.78	2	0.7
			25	7,576	15,960	49	3.1	0.26	0.70	2	0.8	8,513	18,292	50	3.2	0.26	0.68	2	1.0	6,446	14,733	53	3	0.19	0.60	2	0.5
			70	6,788	14,058	46	5.5	0.19	0.50	2	0.4	6,671	15,363	50	6.9	0.20	0.63	2	0.6	5,824	13,246	50	6	0.16	0.54	2	0.6
			130	6,908	15,288	46	8.4	0.17	0.52	2	0.6	5,609	14,023	51	9.2	0.16	0.48	2	1.0	5,324	12,678	50	8	0.13	0.41	2	0.5
			170	6,270	14,477	48	9.1	0.14	0.44	2	0.7	5,612	12,263	45	9.7	0.13	0.43	2	1.2	5,839	13,276	45	11	0.14	0.46	2	0.6
		0.7	1	8,867	19,064	53	0.2	0.25	0.72	2	0.5	9,181	19,611	53	0.2	0.31	0.84	2	0.7	8,368	18,786	55	1	0.29	0.90	2	0.6
			25	7,424	16,685	53	2.5	0.24	0.66	2	0.7	7,957	17,659	52	3.2	0.25	0.71	2	0.9	6,961	16,428	54	3	0.24	0.65	2	1.0
			70	7,567	17,604	51	5.8	0.22	0.61	2	1.1	7,492	16,336	48	6.1	0.20	0.58	2	0.8	6,992	16,936	52	7	0.22	0.56	2	0.7
			130	8,183	17,753	46	7.7	0.18	0.49	2	0.9	6,473	15,222	47	10.3	0.19	0.54	2	1.0	6,826	14,236	43	10	0.17	0.46	2	1.2
			170	5,372	13,435	50	9.6	0.14	0.45	2	0.9	5,281	13,083	49	10.4	0.14	0.41	2	0.5	6,152	15,135	46	13	0.17	0.50	2	1.2
20	26	0.5	1	14,741	33,687	56	0.3	0.29	0.89	2	3.5	13,960	33,366	58	0.4	0.27	0.90	2	4.6	13,703	32,459	57	0	0.28	0.87	2	3.2
			25	10,744	27,262	57	3.7	0.25	0.77	2	3.7	13,234	30,054	53	3.3	0.22	0.76	2	3.7	12,159	27,761	53	3	0.21	0.70	2	2.9
			70	11,776	29,148	53	6.5	0.21	0.68	2	3.3	12,841	30,247	51	6.6	0.20	0.66	2	3.3	10,743	25,978	52	7	0.18	0.60	2	2.8
			130	11,634	28,451	50	8.9	0.16	0.57	2	3.7	9,811	25,009	51	9.7	0.16	0.54	2	2.6	11,316	27,207	50	9	0.15	0.52	2	4.1
			170	8,534	20,295	48	9.7	0.11	0.49	2	2.1	11,724	26,615	46	9.5	0.13	0.46	2	2.4	9,423	22,701	48	11	0.12	0.40	2	3.1
		0.6	1	18,276	40,835	55	0.3	0.33	0.94	2	5.1	15,972	37,747	57	0.5	0.34	0.93	2	4.7	18,083	41,684	56	0	0.32	0.93	2	6.2
			25	15,460	35,793	54	2.9	0.26	0.77	2	10.7	17,072	39,816	55	2.4	0.24	0.72	2	4.0	15,566	36,892	55	3	0.25	0.69	2	5.7
			70	12,978	29,595	50	6.4	0.21	0.66	2	4.2	14,101	31,678	49	6.4	0.20	0.65	2	3.3	13,406	32,796	53	6	0.20	0.62	2	4.1
			130	13,081	32,464	51	8.7	0.18	0.61	2	3.3	11,232	27,394	50	9.0	0.16	0.54	2	3.2	13,238	32,000	50	8	0.16	0.51	2	5.1
			170	11,308	26,952	47	10.9	0.15	0.58	2	3.6	11,331	28,942	50	11.0	0.16	0.53	2	4.2	10,677	25,656	46	12	0.15	0.49	2	6.0
		0.7	1	19,838	45,586	56	0.2	0.33	0.92	2	5.8	17,636	41,650	57	0.3	0.33	0.90	2	4.8	19,408	45,598	57	0	0.35	0.91	2	5.5
			25	18,236	41,075	53	2.9	0.29	0.80	2	6.3	16,779	40,487	56	3.0	0.27	0.74	2	5.6	17,639	39,282	52	3	0.27	0.76	2	6.6
			70	15,763	37,239	52	6.0	0.25	0.67	2	4.9	15,696	34,743	48	6.9	0.23	0.67	2	6.4	13,846	33,458	51	7	0.21	0.64	2	5.2
			130	15,422	37,427	49	9.5	0.21	0.64	2	5.7	13,263	29,787	46	9.8	0.18	0.61	2	4.7	15,718	35,588	47	9	0.18	0.57	2	5.8
			170	12,027	30,371	49	11.1	0.17	0.58	2	4.9	12,949	31,670	49	10.6	0.17	0.50	2	4.9	12,627	28,344	45	11	0.15	0.46	2	5.3
30	38	0.5	1	22,196	55,083	59	0.3	0.30	0.94	2	19	21,169	50,827	58	0.3	0.26	0.90	2	14	22,693	54,213	58	0	0.29	0.90	2	15
			25	19,952	47,129	55	2.7	0.22	0.78	2	15	19,038	44,793	55	2.6	0.20	0.73	2	11	21,839	51,455	55	3	0.24	0.72	2	15
			70	18,807	45,080	52	6.0	0.20	0.69	2	13	17,408	40,388	50	6.5	0.19	0.63	2	17	18,512	43,212	50	7	0.19	0.65	2	14
			130	17,823	43,463	50	8.5	0.17	0.60	2	12	15,271	38,099	51	9.4	0.15	0.57	2	13	14,660	35,525	48	10	0.15	0.53	2	16
			170	16,003	39,533	49	10.9	0.16	0.61	2	15	16,363	40,749	50	9.4	0.13	0.49	2	15	15,126	36,476	48	10	0.13	0.46	2	10
		0.6	1	25,098	58,815	57	0.3	0.32	0.93	2	21	27,804	66,037	58	0.3	0.31	0.95	2	25	24,642	59,233	58	0	0.33	0.95	2	26
			25	23,668	55,252	54	3.2	0.27	0.82	2	29	22,470	53,456	55	3.0	0.25	0.76	2	22	24,261	55,651	53	3	0.25	0.78	2	19
			70	22,784	51,546	50	5.8	0.22	0.68	2	16	20,624	49,060	52	6.1	0.21	0.68	2	17	19,994	46,160	50	7	0.21	0.65	2	16
			130	19,865	49,991	51	9.3	0.20	0.65	2	17	19,899	47,890	49	9.4	0.18	0.60	2	22	18,429	44,362	49	10	0.18	0.56	2	27
			170	18,454	44,769	48	10.8	0.18	0.58	2	12	17,549	42,952	48	10.9	0.16	0.55	2	19	18,064	44,562	48	12	0.16	0.57	2	23
		0.7	1	32,074	74,904	57	0.3	0.38	0.95	2	31	30,196	69,192	56	0.3	0.34	0.94	2	23	31,124	74,241	58	0	0.37	0.95	2	30
			25	28,015	66,290	55	2.7	0.29	0.81	2	32	24,996	57,944	53	3.7	0.29	0.80	2	42	29,100	66,458	53	3	0.27	0.78	2	30
			70	23,851	57,326	52	6.2	0.25	0.72	2	23	27,386	62,199	50	6.2	0.25	0.69	2	30	23,239	55,471	51	7	0.23	0.67	2	27
			130	22,986	55,089	50	8.3	0.20	0.65	2	17	18,219	44,916	49	10.9	0.19	0.67	2	32	21,746	51,800	48	10	0.19	0.60	2	30
			170	21,471	53,754	49	10.8	0.20	0.62	2	25	19,029	46,755	48	10.9	0.17	0.55	2	20	20,768	51,348	49	11	0.17	0.53	2	22

Table 2 Contd...:

N	E	P	D	cv = 0.5 Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	cv = 1 Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	cv = 1.5 Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU
40	51	0.5	1	30,834	72,808	57	0.3	0.28	0.94	2	57	29,546	73,980	60	0.3	0.30	0.93	2	150	29,152	68,600	57	0	0.30	0.93	2	66
			25	28,261	65,254	54	2.6	0.23	0.77	2	54	29,736	69,713	54	2.9	0.23	0.79	2	90	26,681	61,000	53	3	0.22	0.74	2	52
			70	22,775	55,507	53	6.0	0.20	0.68	2	32	23,026	55,044	51	6.7	0.19	0.67	2	74	23,586	58,183	53	7	0.20	0.65	2	52
			130	20,204	50,311	51	9.2	0.16	0.59	2	43	19,002	47,682	51	9.3	0.15	0.55	2	51	20,538	49,192	48	10	0.15	0.54	2	59
			170	19,097	48,495	49	11.4	0.15	0.60	2	189	19,614	48,106	48	11.4	0.14	0.55	2	46	18,827	47,938	49	12	0.14	0.52	2	50
		0.6	1	33,059	78,944	58	0.2	0.29	0.94	2	75	33,786	80,340	58	0.3	0.31	0.95	2	97	35,546	82,441	56	0	0.32	0.95	2	84
			25	35,361	84,226	55	3.0	0.30	0.84	2	80	30,892	72,484	54	3.2	0.27	0.79	2	74	32,331	72,817	52	4	0.28	0.78	2	59
			70	28,332	68,835	53	5.6	0.22	0.72	2	71	29,883	68,697	51	5.9	0.21	0.69	2	52	25,388	62,284	53	7	0.20	0.65	2	67
			130	23,113	56,521	49	9.7	0.18	0.65	2	59	24,005	57,809	49	9.6	0.17	0.62	2	46	22,658	53,407	47	11	0.17	0.58	2	53
			170	24,684	57,790	47	10.7	0.17	0.62	2	51	23,453	57,318	47	11.8	0.17	0.60	2	69	25,219	61,276	48	11	0.16	0.52	2	61
		0.7	1	38,805	91,688	57	0.3	0.35	0.95	2	92	38,237	89,979	57	0.4	0.36	0.96	2	148	44,912	105,788	57	0	0.37	0.95	2	114
			25	38,089	86,882	53	2.9	0.30	0.83	2	114	36,358	85,439	54	3.2	0.30	0.80	2	104	35,496	82,168	54	3	0.27	0.76	3	137
			70	32,045	74,510	50	6.6	0.26	0.73	2	67	30,802	72,752	51	7.1	0.25	0.71	2	72	30,456	72,173	51	7	0.23	0.69	2	66
			130	28,358	68,602	49	9.8	0.21	0.69	2	121	31,383	75,524	49	9.0	0.20	0.65	2	87	28,634	69,720	49	10	0.20	0.62	2	112
			170	27,674	67,064	48	11.1	0.19	0.63	2	152	26,118	62,847	47	11.0	0.17	0.57	2	60	26,963	63,885	47	11	0.17	0.54	2	79

N	E	P	D	cv = 0.5 Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	cv = 1 Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	cv = 1.5 Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU
10	13	0.8	1	9,622	20,870	54	0.2	0.31	0.87	2	0.6	10,757	22,455	52	0.3	0.34	0.87	2	0.6	11,536	24,495	53	0	0.32	0.86	2	1.0
			25	10,361	21,582	49	2.7	0.28	0.73	2	0.8	8,780	18,121	48	3.7	0.28	0.72	2	0.9	8,940	20,451	53	3	0.27	0.70	2	0.7
			70	9,240	20,777	49	6.1	0.26	0.63	2	0.9	7,810	17,545	50	5.6	0.21	0.60	2	0.6	8,048	18,141	49	7	0.22	0.58	2	0.8
			130	6,276	15,396	51	8.3	0.18	0.55	2	0.5	8,371	18,784	47	8.8	0.20	0.53	2	0.9	8,717	19,518	47	8	0.19	0.46	2	1.0
			170	6,623	15,703	46	12.2	0.19	0.55	2	0.8	6,485	15,747	49	9.4	0.15	0.44	2	0.6	7,376	16,781	46	10	0.16	0.44	2	0.7
		0.9	1	10,879	24,475	55	0.4	0.37	0.92	2	0.8	10,467	22,803	54	0.4	0.36	0.90	2	0.8	11,689	25,266	53	0	0.33	0.88	2	1.1
			25	9,301	20,463	52	2.7	0.27	0.72	2	0.9	11,666	23,938	48	3.0	0.30	0.73	2	0.8	8,799	19,955	53	3	0.24	0.67	2	0.7
			70	9,232	19,522	48	4.6	0.21	0.58	2	0.6	8,798	18,970	47	6.4	0.23	0.63	2	0.6	8,864	20,641	50	7	0.25	0.63	2	1.2
			130	8,229	20,253	50	9.8	0.24	0.57	2	1.4	7,043	16,137	47	9.8	0.18	0.55	2	0.9	7,545	19,259	50	11	0.22	0.54	2	1.0
			170	8,148	18,826	46	10.3	0.20	0.49	2	0.7	6,941	18,103	51	11.1	0.20	0.48	2	0.9	9,048	19,484	44	9	0.16	0.45	2	0.9
20	26	0.8	1	22,415	48,765	54	0.3	0.34	0.94	2	7.8	22,339	49,984	55	0.3	0.34	0.94	2	9.3	23,693	52,984	55	0	0.39	0.93	2	7.9
			25	20,061	46,551	54	3.2	0.32	0.81	2	8.7	18,962	43,207	53	3.3	0.30	0.79	2	9.6	19,403	41,589	50	3	0.27	0.73	2	8.3
			70	16,990	39,854	51	6.5	0.27	0.70	2	5.4	18,868	43,079	49	7.3	0.29	0.67	2	7.0	16,623	38,971	51	6	0.23	0.65	2	6.0
			130	16,767	39,821	48	9.9	0.24	0.66	2	7.3	14,933	33,369	46	9.4	0.20	0.56	2	5.8	15,166	34,797	47	10	0.20	0.53	2	5.6
			170	15,767	36,850	48	9.6	0.19	0.56	2	4.4	15,943	37,160	46	10.8	0.20	0.55	2	6.1	14,750	34,904	48	10	0.17	0.52	2	5.9
		0.9	1	21,862	48,194	54	0.3	0.34	0.94	2	6.9	24,252	55,219	56	0.4	0.40	0.94	2	11.9	20,817	47,128	55	0	0.38	0.92	2	8.8
			25	21,635	48,713	53	3.1	0.32	0.82	2	10.8	20,575	47,517	53	3.4	0.33	0.79	2	8.6	20,928	45,671	51	3	0.28	0.74	2	8.6
			70	22,327	49,583	49	5.8	0.29	0.71	2	7.3	20,161	47,665	51	6.4	0.28	0.67	2	6.8	21,419	47,571	49	6	0.26	0.65	2	9.9
			130	19,289	45,506	48	9.2	0.25	0.65	2	7.4	18,458	42,915	47	9.6	0.23	0.64	2	11.7	16,666	37,498	46	9	0.20	0.59	2	8.2
			170	16,117	37,852	48	9.3	0.19	0.56	2	8.7	16,078	40,260	49	11.1	0.21	0.58	2	8.6	16,202	37,967	46	11	0.19	0.54	2	8.9
30	38	0.8	1	35,244	78,155	55	0.3	0.39	0.96	2	40	33,534	74,354	54	0.4	0.38	0.96	2	41	33,786	75,565	55	1	0.38	0.96	2	42
			25	30,031	69,405	54	3.0	0.31	0.82	2	43	29,758	69,390	54	3.4	0.32	0.82	2	35	28,319	65,474	53	4	0.30	0.79	2	33
			70	27,693	65,846	52	5.8	0.27	0.73	2	34	24,266	57,924	51	6.7	0.26	0.71	2	32	26,190	61,476	51	7	0.25	0.68	2	31
			130	25,861	59,384	48	8.9	0.23	0.65	2	21	24,745	58,043	48	9.6	0.22	0.64	2	37	24,495	57,446	48	10	0.21	0.60	2	43
			170	23,794	57,783	48	11.3	0.22	0.61	2	27	23,031	54,815	47	10.6	0.19	0.59	2	31	20,197	48,154	48	10	0.16	0.49	2	23
		0.9	1	38,921	85,521	54	0.4	0.42	0.97	2	51	38,106	83,645	54	0.4	0.41	0.96	2	44	35,996	80,641	55	0	0.42	0.95	2	51
			25	33,327	72,494	51	3.1	0.33	0.84	2	36	30,913	72,515	54	3.5	0.33	0.83	2	42	34,704	76,487	51	4	0.34	0.81	2	57
			70	31,751	73,568	51	5.7	0.29	0.74	2	29	30,624	72,868	52	6.0	0.28	0.71	2	33	32,952	76,072	50	7	0.28	0.69	2	45
			130	25,846	62,779	49	9.8	0.25	0.69	2	40	26,120	60,756	47	9.7	0.23	0.64	2	37	27,763	64,558	48	9	0.22	0.61	2	45
			170	27,814	66,653	49	9.7	0.22	0.65	2	51	25,979	61,376	47	10.9	0.21	0.59	2	40	25,610	59,123	46	11	0.20	0.59	2	35
40	51	0.8	1	41,054	97,727	58	0.3	0.37	0.96	2	134	47,170	112,886	58	0.4	0.39	0.96	2	154	47,035	110,039	57	0	0.39	0.96	2	131
			25	42,585	97,190	53	3.0	0.32	0.85	2	131	42,513	93,582	51	3.1	0.30	0.82	2	122	41,268	91,930	52	3	0.30	0.80	2	85
			70	39,512	88,697	50	5.4	0.26	0.73	3	139	35,313	85,001	52	6.4	0.26	0.71	2	87	36,381	81,215	49	7	0.25	0.69	2	96
			130	32,333	78,410	50	9.3	0.23	0.66	2	86	30,568	73,328	49	9.5	0.21	0.63	2	132	32,634	77,991	48	10	0.22	0.63	2	122
			170	32,665	78,791	48	10.4	0.21	0.63	2	84	29,880	72,644	48	10.8	0.20	0.58	2	97	29,719	72,827	49	11	0.18	0.59	2	139
		0.9	1	48,024	112,473	57	0.4	0.41	0.97	2	205	49,095	111,030	55	0.3	0.39	0.96	2	160	48,734	109,375	55	0	0.41	0.96	2	226
			25	45,519	102,595	52	3.1	0.36	0.85	2	172	44,816	101,106	52	3.4	0.34	0.83	2	171	43,098	97,775	52	4	0.33	0.81	2	119
			70	39,097	93,874	52	5.9	0.29	0.74	2	151	41,801	94,739	50	6.4	0.28	0.72	2	136	41,085	94,080	49	7	0.28	0.71	2	136
			130	35,286	82,600	48	8.8	0.23	0.67	2	94	38,735	90,361	48	9.2	0.24	0.68	2	201	35,978	85,335	48	10	0.23	0.64	2	180
			170	36,843	86,964	47	10.2	0.22	0.67	2	93	32,386	75,886	46	11.3	0.21	0.65	2	163	32,003	76,805	48	10	0.19	0.55	2	164

N	E	D	TRM	$cv = 0.5$								$cv = 1$								$cv = 1.5$							
				Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU
20	26	1	0.5	35,919	63,217	43	0.4	0.41	0.96	2	23	42,090	72,438	41	0.4	0.42	0.96	2	30	38,163	69,658	45	1	0.42	0.96	2	16
			0.75	25,497	52,649	51	0.3	0.36	0.93	2	9	27,296	56,704	52	0.4	0.39	0.95	2	8	27,209	55,988	51	0	0.39	0.95	2	10
			1.25	10,870	30,100	64	0.2	0.25	0.83	2	3	11,139	30,953	64	0.3	0.25	0.84	2	4	11,366	28,810	60	0	0.25	0.87	2	3
			1.5	9,561	24,713	61	0.2	0.19	0.79	2	5	8,053	21,943	63	0.4	0.19	0.88	2	3	8,198	22,729	64	0	0.19	0.81	2	3
			2	4,388	14,998	71	0.1	0.13	0.67	2	1	4,670	13,328	65	0.2	0.15	0.77	2	2	4,605	14,432	68	0	0.14	0.69	2	2
		25	0.5	36,684	67,151	42	3.3	0.39	0.87	3	16	38,642	72,095	43	3.1	0.37	0.84	3	14	32,284	57,867	40	4	0.34	0.83	3	14
			0.75	23,098	51,096	52	3.2	0.33	0.84	2	14	22,076	48,228	51	2.8	0.28	0.78	3	10	22,506	45,407	47	3	0.30	0.76	2	6
			1.25	10,650	28,056	59	3.3	0.22	0.79	2	3	11,241	27,284	56	3.1	0.20	0.76	2	3	10,965	27,155	56	3	0.21	0.66	2	3
			1.5	6,087	17,492	63	2.3	0.13	0.57	2	3	6,576	18,379	61	2.7	0.15	0.62	2	2	8,291	20,869	57	3	0.16	0.62	2	6
			2	3,910	11,756	63	3.8	0.12	0.66	2	1	4,640	14,172	64	3.3	0.11	0.63	2	2	5,424	14,048	57	4	0.14	0.65	2	1
		70	0.5	34,219	65,313	42	5.6	0.32	0.77	2	9	30,943	56,141	38	6.6	0.30	0.75	2	9	36,172	66,975	39	7	0.32	0.74	3	23
			0.75	21,536	46,647	48	5.4	0.25	0.69	2	5	18,707	40,595	48	6.1	0.24	0.68	2	6	21,452	44,615	46	6	0.23	0.63	2	7
			1.25	9,808	25,294	55	6.3	0.17	0.64	2	4	9,151	23,983	55	6.4	0.17	0.55	2	3	8,463	21,786	54	7	0.16	0.57	2	3
			1.5	6,481	19,323	60	6.6	0.15	0.56	2	12	7,553	22,009	59	6.4	0.16	0.50	2	2	5,067	15,414	61	6	0.10	0.51	2	3
			2	5,734	17,439	60	7.0	0.13	0.65	2	2	3,527	12,027	65	6.1	0.08	0.53	2	2	3,162	11,197	64	8	0.08	0.56	2	2
		130	0.5	31,372	57,078	37	7.8	0.26	0.69	3	7	29,709	57,515	39	9.8	0.29	0.68	3	33	30,420	56,339	36	10	0.26	0.65	3	9
			0.75	18,481	39,829	45	8.8	0.22	0.64	2	7	20,317	44,273	45	8.8	0.22	0.61	2	5	18,765	40,558	43	11	0.23	0.58	2	6
			1.25	9,663	25,782	54	8.2	0.15	0.53	2	6	8,093	21,345	53	9.4	0.13	0.49	2	3	8,236	21,806	52	10	0.13	0.49	2	4
			1.5	6,706	17,708	52	10.6	0.12	0.56	2	3	6,248	18,107	55	10.7	0.12	0.51	2	2	5,336	16,762	58	10	0.10	0.49	2	4
			2	2,940	11,974	67	8.5	0.07	0.44	2	3	3,058	11,303	63	10.1	0.07	0.49	2	2	2,781	10,216	66	7	0.04	0.41	2	2
		170	0.5	29,758	57,697	38	10.2	0.27	0.67	2	21	28,193	53,716	37	10.5	0.24	0.63	3	11	27,394	53,308	38	11	0.24	0.60	3	10
			0.75	18,050	37,760	43	9.1	0.18	0.57	2	9	16,119	36,765	45	10.8	0.20	0.54	2	6	16,687	38,737	45	12	0.21	0.54	2	5
			1.25	6,352	18,759	55	11.6	0.12	0.47	2	3	8,190	20,434	50	10.0	0.11	0.42	2	3	7,680	20,961	52	12	0.12	0.45	2	4
			1.5	6,247	17,238	54	10.1	0.10	0.45	2	2	6,208	18,222	56	10.2	0.09	0.44	2	3	4,382	14,138	58	11	0.08	0.44	2	2
			2	2,316	7,430	59	9.4	0.04	0.37	2	2	2,145	7,498	61	10.2	0.04	0.40	2	2	2,334	9,626	65	11	0.05	0.37	2	3
30	38	1	0.5	60,707	111,717	45	0.4	0.45	0.98	2	513	66,675	121,563	45	0.4	0.43	0.97	2	145	63,116	114,251	44	1	0.46	0.97	2	253
			0.75	42,004	86,588	51	0.3	0.41	0.96	2	34	37,245	80,850	54	0.4	0.37	0.95	2	31	37,054	80,335	53	1	0.40	0.96	2	47
			1.25	18,845	48,983	61	0.2	0.27	0.91	2	15	19,051	47,515	60	0.3	0.25	0.86	2	17	19,626	48,031	59	0	0.27	0.90	2	18
			1.5	13,411	34,409	61	0.3	0.22	0.91	2	11	14,552	38,494	62	0.4	0.22	0.90	2	15	14,200	39,118	63	0	0.23	0.91	2	16
			2	8,363	24,029	65	0.1	0.14	0.74	2	7	8,568	25,880	67	0.2	0.16	0.78	2	10	7,276	22,579	68	0	0.13	0.75	2	9
		25	0.5	60,168	115,666	45	3.3	0.42	0.88	3	60	55,772	101,014	41	3.7	0.37	0.87	2	54	50,231	97,308	45	4	0.36	0.84	3	49
			0.75	37,406	78,378	49	3.1	0.33	0.86	2	40	34,778	71,035	48	3.5	0.31	0.83	2	37	33,493	74,491	51	4	0.32	0.80	2	37
			1.25	16,177	42,505	59	3.4	0.24	0.81	2	13	16,987	42,979	57	3.9	0.24	0.80	2	15	16,073	40,993	57	4	0.21	0.77	2	15
			1.5	13,199	35,302	60	2.8	0.20	0.71	2	13	12,161	32,068	59	2.9	0.16	0.67	2	13	10,007	29,263	62	4	0.17	0.70	2	15
			2	6,914	21,267	64	3.6	0.14	0.72	2	9	6,929	23,084	66	4.0	0.14	0.72	2	6	6,570	21,875	65	5	0.14	0.73	2	6
		70	0.5	55,879	106,016	41	6.4	0.37	0.80	3	115	54,781	107,323	42	6.5	0.36	0.76	2	68	51,841	98,530	40	7	0.33	0.76	3	76
			0.75	34,058	73,333	47	6.1	0.29	0.75	2	28	30,745	69,535	50	6.1	0.27	0.72	2	31	32,901	71,408	48	6	0.26	0.67	2	27
			1.25	16,850	42,823	54	6.5	0.19	0.72	2	21	15,507	41,090	56	6.1	0.18	0.62	2	15	14,771	37,641	55	6	0.15	0.58	2	14
			1.5	10,052	28,314	58	6.8	0.16	0.56	2	16	11,817	32,363	57	6.0	0.14	0.60	2	14	10,724	29,453	57	7	0.14	0.57	2	9
			2	5,445	17,926	62	7.5	0.09	0.68	2	6	4,510	16,441	65	8.0	0.08	0.63	2	7	5,169	17,499	60	10	0.10	0.67	2	9
		130	0.5	47,539	91,158	39	8.9	0.30	0.73	3	39	46,427	90,600	40	9.1	0.29	0.69	3	48	42,276	81,896	39	9	0.27	0.64	2	27
			0.75	29,712	64,970	46	8.4	0.24	0.66	3	30	29,727	67,039	47	9.1	0.24	0.63	2	25	28,587	62,339	44	10	0.22	0.59	2	35
			1.25	13,351	34,694	52	9.0	0.14	0.59	2	13	13,608	35,841	51	10.6	0.16	0.55	2	14	12,601	34,863	53	11	0.16	0.52	2	16
			1.5	9,446	27,229	56	9.4	0.12	0.56	2	10	9,782	27,520	55	9.8	0.12	0.57	2	13	9,187	25,987	54	11	0.11	0.53	2	16
			2	5,385	18,433	60	10.9	0.08	0.65	2	9	3,374	12,295	63	9.9	0.06	0.44	2	6	5,049	17,218	61	9	0.06	0.50	2	11
		170	0.5	45,910	92,258	39	10.8	0.29	0.69	3	53	44,732	85,989	38	9.9	0.25	0.65	3	40	43,400	83,915	37	11	0.24	0.62	3	46
			0.75	27,458	61,094	45	10.0	0.21	0.60	2	32	26,860	62,353	46	10.7	0.21	0.57	2	36	28,272	62,981	44	11	0.20	0.56	2	25
			1.25	11,263	30,264	52	11.1	0.13	0.49	2	11	12,923	34,935	52	10.6	0.13	0.49	2	12	12,021	33,801	52	12	0.13	0.47	2	12
			1.5	9,519	26,447	54	9.8	0.10	0.50	2	8	6,814	22,597	56	13.5	0.10	0.52	2	15	8,842	25,432	53	12	0.10	0.50	2	20
			2	4,106	14,979	60	12.2	0.06	0.52	2	8	5,432	17,883	60	9.6	0.06	0.43	2	8	4,351	14,658	62	8	0.04	0.34	2	8

Table 4 Contd..: Effect of Changes in the Traffic Requirement Multiplier

N	E	D	TRM	$cv = 0.5$								$cv = 1$								$cv = 1.5$							
				Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU
40	51	1	0.5	90,685	167,706	46	0.4	0.48	0.98	2	444	81,975	153,021	46	0.4	0.44	0.98	3	853	85,217	158,185	45	1	0.49	0.98	2	419
			0.75	54,276	117,616	54	0.3	0.39	0.96	2	210	51,200	112,690	54	0.4	0.40	0.97	2	151	52,063	112,053	53	1	0.42	0.96	3	244
			1.25	25,127	63,248	60	0.2	0.26	0.92	2	59	23,808	57,227	58	0.2	0.24	0.88	2	36	23,357	59,587	61	0	0.24	0.89	2	48
			1.5	18,982	52,398	64	0.3	0.24	0.90	2	65	18,556	50,716	63	0.3	0.22	0.92	2	45	17,355	47,109	63	0	0.24	0.92	2	42
			2	10,127	31,547	68	0.1	0.15	0.75	2	22	9,316	29,450	68	0.2	0.13	0.73	2	19	9,713	30,536	68	0	0.14	0.74	2	28
		25	0.5	81,697	155,206	44	3.0	0.41	0.89	3	273	79,678	150,108	43	3.6	0.40	0.89	3	452	79,009	144,050	41	4	0.40	0.86	3	287
			0.75	48,598	106,033	51	3.2	0.34	0.87	2	109	45,949	99,647	50	3.4	0.33	0.84	2	107	41,832	92,682	51	4	0.32	0.81	2	96
			1.25	24,102	60,356	57	3.4	0.23	0.84	2	69	22,199	56,447	57	3.8	0.23	0.81	2	78	24,390	63,106	58	3	0.22	0.78	2	61
			1.5	14,661	41,116	61	3.3	0.17	0.78	2	42	14,144	37,389	59	2.8	0.15	0.67	2	27	17,758	47,996	60	3	0.18	0.69	2	38
			2	7,983	28,892	68	4.1	0.15	0.75	2	25	9,290	28,909	64	3.9	0.13	0.73	2	26	9,253	29,615	64	4	0.13	0.73	2	18
		70	0.5	76,658	144,406	40	6.7	0.39	0.81	3	240	70,754	134,338	41	6.8	0.35	0.79	3	211	69,342	129,536	40	7	0.33	0.77	3	186
			0.75	44,967	97,840	48	5.9	0.29	0.77	2	115	44,046	93,399	47	6.2	0.26	0.75	2	96	40,974	87,179	47	6	0.25	0.72	3	141
			1.25	20,382	52,351	55	6.1	0.18	0.70	2	53	16,350	44,256	57	6.0	0.15	0.61	2	32	20,440	54,599	56	7	0.18	0.68	2	78
			1.5	14,078	40,049	58	6.5	0.15	0.64	2	37	14,030	39,077	56	7.7	0.16	0.62	2	31	13,562	37,991	57	7	0.13	0.60	2	32
			2	7,943	25,510	62	6.9	0.09	0.69	2	18	7,650	24,756	62	7.0	0.09	0.65	2	20	6,309	20,308	62	7	0.06	0.61	2	21
		130	0.5	63,834	127,820	41	9.5	0.33	0.73	3	213	59,288	115,508	39	9.7	0.28	0.73	3	190	63,950	123,977	39	9	0.28	0.69	3	209
			0.75	36,494	80,883	46	9.1	0.24	0.68	2	73	41,600	91,192	45	9.5	0.25	0.66	2	136	38,199	83,523	45	10	0.22	0.61	2	109
			1.25	17,207	45,189	53	8.8	0.14	0.53	2	37	16,961	44,774	53	9.5	0.14	0.56	2	38	18,888	47,944	51	10	0.14	0.53	2	51
			1.5	11,396	33,389	57	8.7	0.10	0.55	2	17	13,002	38,381	56	9.9	0.12	0.55	2	29	11,784	36,897	57	11	0.12	0.56	2	50
			2	5,844	20,360	61	10.3	0.07	0.60	2	21	7,193	22,668	60	8.2	0.06	0.53	2	18	4,482	16,398	63	10	0.05	0.48	2	27
		170	0.5	58,893	118,991	41	9.6	0.26	0.58	3	182	65,704	129,326	38	11.2	0.30	0.68	3	216	56,025	113,329	39	12	0.26	0.63	3	205
			0.75	37,235	84,308	45	10.7	0.23	0.65	2	107	35,659	76,884	43	10.9	0.20	0.59	2	113	31,623	71,048	44	12	0.19	0.56	2	90
			1.25	16,989	47,144	52	11.9	0.15	0.56	2	41	15,557	41,839	52	11.2	0.12	0.50	2	26	16,609	44,351	51	12	0.13	0.51	2	51
			1.5	11,733	33,377	54	11.2	0.11	0.53	2	23	9,264	28,612	54	13.5	0.10	0.53	2	29	9,013	28,519	55	13	0.09	0.53	2	39
			2	6,141	19,495	59	9.7	0.05	0.50	2	21	5,280	16,287	57	10.2	0.04	0.48	2	26	5,933	19,106	59	10	0.05	0.41	2	23

N	E	D	FCM	$cv = 0.5$ Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	$cv = 1$ Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU	$cv = 1.5$ Profit	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU
20	26	1	0.5	32,799	55,369	40	1	0.53	0.97	2	30	33,268	56,173	40	0.6	0.50	0.96	2	19	31,201	56,267	44	0.8	0.50	0.96	2	22
			0.75	24,259	50,251	51	0	0.42	0.95	2	11	24,894	51,990	52	0.5	0.44	0.95	2	10	23,925	50,974	48	0.5	0.41	0.94	2	9
			1.25	12,768	32,771	61	0	0.26	0.86	2	13	11,803	29,895	60	0.1	0.17	0.78	2	3	12,467	31,662	34	0.2	0.23	0.86	2	3
			1.5	8,241	23,652	65	0	0.16	0.75	2	2	9,848	28,061	65	0.2	0.17	0.76	2	2	9,147	25,630	29	0.1	0.15	0.77	2	2
			2	4,904	15,755	69	0	0.08	0.48	2	1	4,935	16,873	71	0.1	0.08	0.49	2	1	6,011	18,312	22	0.0	0.09	0.58	2	1
		25	0.5	32,110	55,734	38	4	0.43	0.86	2	15	29,154	50,735	37	5.2	0.43	0.86	2	18	26,510	45,549	29	4.5	0.41	0.84	2	15
			0.75	20,678	41,659	46	4	0.33	0.82	2	9	20,594	43,021	49	3.6	0.34	0.77	2	8	20,293	42,320	36	3.1	0.32	0.78	2	10
			1.25	10,077	27,680	62	2	0.18	0.67	2	2	12,131	31,294	58	3.0	0.21	0.73	2	4	10,333	25,660	26	1.2	0.16	0.62	2	4
			1.5	6,727	20,419	65	2	0.13	0.68	2	2	10,009	25,019	58	2.2	0.15	0.69	2	4	9,080	24,075	26	1.0	0.15	0.62	2	2
			2	4,673	13,741	65	1	0.06	0.42	2	1	4,401	14,142	67	1.4	0.07	0.43	2	1	4,097	13,258	16	0.3	0.06	0.35	2	1
		70	0.5	27,886	48,492	34	8	0.36	0.77	2	13	25,009	44,764	35	8.7	0.34	0.72	2	8	24,445	44,715	29	7.2	0.32	0.69	3	12
			0.75	19,046	39,495	45	6	0.27	0.68	2	7	16,988	35,844	45	7.3	0.25	0.68	2	6	21,044	42,833	32	6.2	0.29	0.68	2	7
			1.25	11,951	28,450	52	6	0.18	0.66	2	9	9,654	25,867	57	5.2	0.15	0.59	2	34	10,792	29,245	31	2.2	0.14	0.50	2	3
			1.5	8,399	23,895	60	5	0.14	0.58	2	2	7,499	20,858	59	4.9	0.12	0.54	2	2	7,742	20,528	21	1.9	0.12	0.50	2	3
			2	4,116	12,885	65	3	0.05	0.36	2	1	4,920	15,160	63	4.1	0.07	0.46	2	1	4,167	13,678	16	0.8	0.06	0.34	2	1
		130	0.5	26,217	49,829	36	12	0.32	0.72	2	14	24,609	46,383	34	13.2	0.31	0.66	2	14	22,448	41,952	25	9.8	0.28	0.64	2	10
			0.75	16,565	35,224	42	11	0.25	0.66	2	6	16,248	35,535	45	9.4	0.21	0.58	2	5	17,396	37,146	28	7.2	0.22	0.61	2	8
			1.25	9,111	24,256	55	8	0.13	0.50	2	2	9,789	25,930	54	8.0	0.13	0.53	2	3	8,348	21,623	20	3.4	0.12	0.49	2	3
			1.5	7,886	21,822	57	7	0.11	0.50	2	2	8,225	21,637	55	6.5	0.10	0.41	2	2	6,625	19,438	20	2.5	0.09	0.43	2	2
			2	3,271	10,968	65	5	0.04	0.33	2	1	4,670	14,562	63	4.7	0.05	0.35	2	1	3,916	12,332	13	1.6	0.05	0.42	2	1
		170	0.5	22,348	42,276	33	14	0.28	0.66	2	7	21,407	39,353	31	14.7	0.24	0.63	2	11	22,116	41,198	24	10.3	0.24	0.57	2	10
			0.75	15,497	34,885	44	11	0.20	0.58	2	4	13,276	30,990	45	12.3	0.19	0.53	2	4	14,091	31,973	25	6.8	0.18	0.55	2	7
			1.25	8,784	23,456	54	8	0.10	0.47	2	5	7,633	21,236	55	8.9	0.10	0.39	2	3	9,996	27,267	27	4.1	0.12	0.42	2	3
			1.5	7,292	19,961	55	9	0.10	0.46	2	5	7,282	20,987	58	7.7	0.09	0.40	2	2	7,007	19,319	19	2.9	0.09	0.40	2	30
			2	4,379	14,672	64	6	0.05	0.28	2	1	3,861	13,588	65	6.7	0.05	0.32	2	1	5,018	15,924	17	2.3	0.07	0.40	2	10
																						0	0.0				
30	38	1	0.5	45,730	78,067	41	1	0.50	0.97	2	119	49,860	83,997	40	0.7	0.52	0.98	2	117	52,581	92,197	69	1.1	0.51	0.97	3	173
			0.75	37,326	76,777	51	0	0.44	0.96	2	61	38,137	76,887	50	0.5	0.41	0.96	2	47	38,769	79,915	72	0.8	0.44	0.96	2	45
			1.25	19,945	48,771	59	0	0.24	0.88	2	18	17,522	45,831	62	0.2	0.22	0.87	2	14	18,639	48,544	53	0.2	0.22	0.90	2	19
			1.5	12,425	35,049	64	0	0.16	0.76	2	9	14,638	39,141	62	0.1	0.16	0.73	2	9	12,646	36,523	42	0.1	0.16	0.82	2	14
			2	7,740	24,543	68	0	0.10	0.61	2	6	8,331	24,352	66	0.1	0.09	0.60	2	5	8,877	26,797	32	0.0	0.09	0.58	2	5
		25	0.5	43,104	75,710	38	5	0.44	0.88	2	91	45,489	78,666	37	5.3	0.43	0.86	3	125	40,926	70,428	45	7.3	0.41	0.86	2	77
			0.75	31,141	64,622	48	4	0.35	0.85	2	34	34,340	72,131	49	3.6	0.35	0.82	2	46	33,250	66,760	55	4.8	0.33	0.81	2	54
			1.25	18,616	46,076	58	2	0.19	0.73	2	12	19,480	48,766	58	2.5	0.20	0.74	2	16	17,300	44,685	47	1.8	0.17	0.66	2	11
			1.5	13,497	35,545	60	2	0.15	0.67	2	10	13,479	38,044	62	2.3	0.15	0.70	2	10	12,866	35,008	38	1.6	0.15	0.66	2	12
			2	8,285	24,134	64	1	0.08	0.54	2	5	7,395	24,476	68	1.5	0.08	0.54	2	5	7,784	24,744	29	0.7	0.08	0.50	2	6
		70	0.5	43,835	79,980	36	9	0.40	0.80	2	65	39,995	72,046	34	10.2	0.38	0.76	2	60	38,068	68,053	42	11.6	0.34	0.73	2	44
			0.75	29,019	60,486	45	7	0.28	0.73	2	24	29,616	62,713	45	7.4	0.28	0.71	3	43	26,729	55,823	44	7.7	0.26	0.70	2	24
			1.25	15,720	40,898	56	5	0.17	0.67	2	11	15,043	40,508	57	5.7	0.17	0.63	2	16	14,739	38,598	39	3.8	0.16	0.61	2	11
			1.5	13,595	35,138	57	4	0.12	0.53	2	10	10,719	29,909	59	5.3	0.12	0.58	2	10	13,196	35,462	37	2.8	0.11	0.59	2	10
			2	5,987	20,021	67	3	0.05	0.44	2	4	7,825	25,386	65	3.9	0.08	0.52	2	6	7,447	23,688	27	1.4	0.07	0.43	2	6
		130	0.5	35,418	66,103	33	13	0.31	0.71	2	53	42,579	81,000	35	12.2	0.33	0.70	3	61	34,349	65,443	38	17.4	0.31	0.68	2	43
			0.75	28,995	63,291	44	10	0.27	0.67	2	30	25,721	55,723	43	10.8	0.24	0.64	2	32	24,072	50,871	37	10.5	0.22	0.61	2	38
			1.25	14,678	40,538	56	8	0.14	0.64	2	16	13,165	35,869	55	8.4	0.13	0.56	2	11	14,665	39,120	38	5.7	0.14	0.52	2	15
			1.5	10,104	28,586	58	7	0.09	0.46	2	10	11,493	31,451	56	7.1	0.10	0.50	2	8	11,728	32,151	32	3.9	0.10	0.49	2	8
			2	7,473	22,964	61	6	0.07	0.43	2	5	4,961	17,797	67	5.3	0.05	0.37	2	4	6,454	22,240	25	2.8	0.07	0.43	2	5
		170	0.5	35,843	70,326	35	14	0.31	0.67	3	56	32,442	62,937	33	15.6	0.28	0.66	2	39	31,908	60,663	35	16.5	0.25	0.61	2	47
			0.75	25,603	55,981	41	13	0.24	0.64	2	39	23,533	52,253	43	12.2	0.21	0.59	2	30	24,815	54,955	41	13.0	0.21	0.60	2	51
			1.25	12,525	32,412	51	10	0.12	0.54	2	13	12,719	34,086	52	10.5	0.12	0.54	2	13	11,413	31,748	30	5.9	0.12	0.42	2	9
			1.5	11,016	31,352	57	8	0.10	0.46	2	9	8,686	23,934	55	9.1	0.08	0.40	2	7	9,315	26,067	26	3.6	0.07	0.40	2	10
			2	6,216	17,970	59	6	0.05	0.32	2	4	6,096	19,127	60	7.8	0.06	0.40	2	6	5,333	17,126	19	2.5	0.05	0.36	2	6

N	E	D	FCM	Profit ($cv = 0.5$)	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU
40	51	1	0.5	73,588	124,061	40	1	0.56	0.98	2	5053
			0.75	47,076	97,372	51	0	0.42	0.96	2	133
			1.25	23,608	64,112	63	0	0.23	0.92	2	48
			1.5	16,953	49,971	66	0	0.16	0.76	2	31
			2	11,550	35,899	68	0	0.09	0.65	2	20
		25	0.5	58,922	102,522	37	5	0.45	0.90	2	1073
			0.75	44,813	91,912	48	3	0.34	0.85	2	119
			1.25	23,105	61,682	60	2	0.20	0.77	2	38
			1.5	19,687	53,324	61	2	0.16	0.73	2	34
			2	9,931	33,239	69	1	0.08	0.53	2	21
		70	0.5	55,496	101,547	36	9	0.40	0.81	3	298
			0.75	39,650	83,904	45	7	0.31	0.75	2	96
			1.25	23,396	59,999	56	5	0.18	0.68	2	34
			1.5	16,648	45,664	59	4	0.12	0.60	2	22
			2	8,062	26,989	66	4	0.07	0.60	2	15
		130	0.5	50,536	95,783	35	13	0.34	0.74	2	146
			0.75	37,738	83,611	44	11	0.27	0.70	2	110
			1.25	18,432	50,698	55	9	0.15	0.60	2	28
			1.5	16,604	44,962	56	7	0.11	0.53	2	20
			2	7,702	23,961	62	5	0.05	0.41	2	15
		170	0.5	44,616	86,884	34	15	0.29	0.70	2	132
			0.75	32,058	73,811	44	12	0.23	0.65	2	84
			1.25	16,570	44,961	53	10	0.12	0.60	2	27
			1.5	11,607	34,592	57	10	0.09	0.55	2	17
			2	8,247	28,681	64	7	0.06	0.46	2	19

N	E	D	FCM	Profit ($cv = 1$)	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU
40	51	1	0.5	69,063	116,857	40	0.6	0.53	0.97	2	872
			0.75	48,773	97,570	50	0.5	0.42	0.97	2	181
			1.25	24,376	62,304	61	0.3	0.24	0.92	2	57
			1.5	19,021	52,174	63	0.2	0.16	0.84	2	30
			2	10,945	33,175	67	0.1	0.08	0.68	2	19
		25	0.5	64,633	114,555	38	5.1	0.45	0.89	3	539
			0.75	45,739	93,729	47	3.8	0.34	0.83	2	148
			1.25	22,056	56,899	59	2.5	0.19	0.73	2	43
			1.5	17,260	47,814	62	1.9	0.14	0.60	2	46
			2	10,614	31,667	65	1.4	0.08	0.53	2	20
		70	0.5	57,381	104,154	36	9.1	0.37	0.78	3	250
			0.75	36,972	82,262	47	8.0	0.30	0.75	2	147
			1.25	22,222	55,167	54	5.3	0.16	0.66	2	37
			1.5	15,950	47,872	62	4.8	0.13	0.64	2	28
			2	8,917	28,558	65	3.4	0.06	0.47	2	15
		130	0.5	45,622	87,140	34	13.7	0.31	0.68	2	175
			0.75	33,885	73,204	43	11.0	0.23	0.66	2	95
			1.25	20,295	56,738	56	8.1	0.15	0.60	2	33
			1.5	14,060	40,720	58	7.7	0.10	0.57	2	28
			2	8,807	28,130	63	6.1	0.06	0.45	2	17
		170	0.5	44,167	85,555	33	15.2	0.28	0.67	2	132
			0.75	31,725	69,798	42	13.0	0.21	0.63	2	73
			1.25	17,551	48,881	55	8.9	0.12	0.49	2	33
			1.5	12,629	35,215	56	8.6	0.08	0.46	2	25
			2	8,488	28,684	63	7.1	0.06	0.38	2	17

N	E	D	FCM	Profit ($cv = 1.5$)	Revenue	Flow Cost (%)	Delay Cost (%)	Avg. Util.	Max. Util.	#Itr	CPU
40	51	1	0.5	70,390	115,060	78	1.6	0.50	0.98	2	558
			0.75	51,194	102,110	89	1.1	0.42	0.96	2	123
			1.25	25,241	66,637	73	0.4	0.25	0.92	2	54
			1.5	19,348	56,337	66	0.2	0.18	0.86	2	60
			2	11,543	35,150	42	0.1	0.09	0.66	2	20
		25	0.5	59,103	101,892	65	10.6	0.43	0.86	2	364
			0.75	39,583	82,505	70	6.1	0.32	0.81	2	292
			1.25	24,005	64,486	68	3.6	0.22	0.75	2	43
			1.5	17,203	50,109	56	2.2	0.15	0.68	2	30
			2	8,870	28,232	34	0.8	0.07	0.50	2	16
		70	0.5	48,739	90,570	58	16.2	0.35	0.74	2	144
			0.75	35,202	73,647	57	11.3	0.28	0.72	2	73
			1.25	19,280	51,547	52	4.9	0.15	0.62	2	34
			1.5	16,110	43,621	45	3.7	0.12	0.59	2	20
			2	9,754	31,201	36	2.3	0.08	0.50	2	20
		130	0.5	47,040	88,065	51	21.9	0.30	0.67	3	158
			0.75	33,093	73,686	57	14.9	0.23	0.63	2	126
			1.25	17,845	46,962	44	7.3	0.13	0.51	2	38
			1.5	13,435	38,885	40	5.2	0.10	0.48	2	21
			2	7,397	26,010	30	3.2	0.06	0.46	2	20
		170	0.5	42,151	78,639	42	23.0	0.26	0.64	2	128
			0.75	30,422	68,944	52	16.5	0.21	0.58	2	105
			1.25	15,923	42,178	40	7.1	0.10	0.47	2	30
			1.5	13,778	38,435	38	6.2	0.09	0.44	2	20
			2	8,874	27,228	29	3.7	0.06	0.39	2	18

Acknowledgment

The research was supported by the Discovery grant from National Science and Engineering Research Council of Canada. Their support is highly acknowledged.

References

Amiri, A. 2003. The multi-hour bandwidth packing problem with response time guarantees. *Information Technology and Management* **4** 113–127.

Amiri, A. 2005. The selection and scheduling of telecommunication calls with time windows. *European Journal of Operational Research* **167** 243–256.

Amiri, A., R. Barkhi. 2000. The multi-hour bandwidth packing problem. *Computer and Operations Research* **27** 1–14.

Amiri, A., R. Barkhi. 2012. The combinatorial bandwidth packing problem. *European Journal of Operations Research* **208** 37–45.

Amiri, A., E. Rolland, R. Barkhi. 1999. Bandwidth packing with queuing delay costs: Bounding and heuristic procedures. *European Journal of Operational Research* **112** 635–645.

Anderson, C.A., K. Fraughnaugh, M. Parkner, J. Ryan. 1993. Path assignment for call routing: An application of tabu search. *Annals of Operations Research* **41** 301–312.

Bose, I. 2009. Bandwidth packing with priority classes. *European Journal of Operational Research* **192** 313–325.

Cox, L., L. Davis, Y. Qui. 1991. Dynamic anticipatory routing in cuircuit-switched telecommunications networks. L. Davis, ed., *Handbook of Genetic Algorithms*, vol. 11. Van Norstrand/Reinhold, New York, 229–340.

Dantzer, J.-F., M. Haddani, B. Robert. 2000. On the stability of a bandwidth packing algorithm. *Probability in the Engineering and Information Sciences* **14** 57–79.

Elhedhli, S. 2005. Exact solution of a class of nonlinear knapsack problems. *Operations Research Letters* **33**(6) 615–624.

Han, J., K. Lee, C. Lee, S. Park. 2013. Exact algorithms for a bandwidth packing problem with queueing delay guarantees. *INFORMS Journal on Computing* **25**(3) 585–596.

Laguna, M., F. Glover. 1993. Bandwidth packing: A tabu search approach. *Management Science* **39** 492–500.

Park, K., S. Kang, S. Park. 1996. An integer programming approach to the bandwidth packing problem. *Management Science* **42** 1277–1291.

Parker, M., J. Ryan. 1995. A column generation algorithm for bandwidth packing. *Telecommunication Systems* **2** 185–196.

Rolland, E., A. Amiri, R. Barkhi. 1999. Queueing delay guarantees in bandwidth packing. *Computers and Operations Research* **26** 921–935.

Vidyarthi, N.K., S. Jayaswal, V.B.T. Chetty. 2013. Exact solution to bandwidth packing problem with queueing delays. *Working paper* .

Villa, C., K. Hoffman. 2006. A column-generation and branch-and-cut approach to the bandwidth-packing problem. *Journal of Research of the National Institute of Standards and Technology* **111** 161–185.