Mixed Integer Programming Model For A Variant Of Periodic Vehicle Routing Problem

Bambang Irawan, Herman Mawengkang

Abstract— **The paper develops a model for the optimal management of periodic deliveries of a given commodity called Periodic Vehicle Routing Problem (PVRP) incorporated time windows, fleet and driver scheduling, pick-up and delivery in the periodic planning.. The goal is to schedule the deliveries according to feasible combinations of delivery days and to determine the scheduling of fleet and driver and routing policies of the vehicles. The objective is to minimize the sum of the costs of all routes over the planning horizon. We model the problem as a linear mixed integer program and we propose a feasible neighbourhood search approach to solve the problem.**

Index Terms—**Vehicle routing problem, scheduling, integer programming, optimal management, neighbourhood search**

I. Introduction

Vehicle Routing Problem (VRP) is one of the important issues that exist in transportation system. This is a well known combinatorial optimization problem which requires the determination of an optimal set of routes used by a fleet of vehicles to serve a set of customers, taking into account various operational constraints.VRP was first introduced by [30]. Since then many researchers have been working in this area to discover new methodologies in selecting the best routes in order to find the better solutions. There are a number of survey can be found in literature for VRP, such as [18], [16], [2], [13], [3], [21].

The classical vehicle routing problem (*VRP*) is defined as follows: vehicles with a fixed capacity Q must deliver order quantities q_i ($i = 1,\ldots,n$) of goods to n customers from a single depot (i = 0). Knowing the distance d_{ij} between customers i and j ($i, j = 1,\ldots,n$), the objective of the problem is to minimize the total distance traveled by the vehicles in a way that only one vehicle handles the deliveries for a given customer and the total quantity of goods that a single vehicle delivers is not larger than Q [1].

In literature there are some variants of VRP which are grouped according to specific constraints. Some of the well known variants are: Capacitated VRP (CVRP), the vehicles are restricted to carry limited capacity; VRP with time windows (VRPTW), each customer is served within a defined time frame; multiple depots VRP (MDVRP), in this variant goods can be delivered to a customer from a set of depots; VRP with pick-up and delivery (VRPPD), goods not only need to brought from the depot to the customers, but also must be picked-up at a number of customers and brought back to the depot. Taking into account several days of planning for routing problems is another variant of the VRP, known as the priodic VRP (PVRP).

In PVRP, within a given time horizon, there is a set of customers needs to be visited once or several times. There would be a visiting schedules associated with each customer. A fleet of vehicles is available and each vehicle leaves the depot, serves a set of customers, when its work shift or capacity is over, returns to the depot. The problem is to minimize the total length of the routes travelled by the vehicles on the time horizon. This problem is very important in real world applications, such as, distribution for bakery companies [20] , blood product distribution [29], or pick-up of raw materials for a manufacture of automobile parts [15].

A survey on PVRP and its extensions can be found in [19]. Due to the complexity of the problem most of the works present heuristic approaches, nevertheless [22] proposed an exact method. [28] addressed a combined of heuristic and exact method for solving PVRP.

Early formulations of the PVRP were developed by [4] and by [24] who proposed heuristics applied to waste collection problems. [27] use the idea of the generalized assignment method proposed by [11] and assign a visiting schedule to each vertex. Eventually a heuristic for the VRP is applied to each day. [25] developed a heuristic organized in four phases. Solution methods in these papers have focused on two-stage (construction and improvement) heuristics. [8] present another algorithm: The solution algorithm is a TS heuristic which, differently from the above heuristics, may allow infeasible solutions during the search process. Similarly, good results were obtained in the more recent work of [10], [1], and [5] who provide specific practical applications of the PVRP. [23] used particle swarm optimization to tackle the problem.

[12] introduce the Period Vehicle Routing Problem with Service Choice (PVRP-SC) which allows service levels to be determined endogenously. They develop an integer programming formulation of the problem.

[9] presents modeling techniques for distribution problems with varying service requirements. [26] develop continuous approximation models for distribution network design with multiple service levels. These references show that continuous approximations can be powerful tools for strategic and tactical decisions when service choice exists. In continuous approximation models, aggregated data are used instead of more detailed inputs.

[20] present PVRP with time windows (PVRPTW). The problem requires the generation of a limited number of routes for each day of a given planning horizon. The objective is to minimize the total travel cost while satisfying several constraints.

Manuscript received May, 2014.
Bambang Irawan, *Graduate School of Mathematics, University of Sumatera Utara, Medan, Indonesia, Phone/ Mobile No 62618212593*
Herman Mawengkang, *Mathematics, University of Sumatera Utara, Medan, Indonesia, Phone/ Mobile No., 0811695879*

This paper concerns with a comprehensive model for the PVRP incorporated with time windows, fleet and driver scheduling, pick-up and delivery (PVRFDPDTWP) . The basic framework of the vehicle routing part can be viewed as a Heterogeneous Vehicle Routing Problem with Time Windows (HVRPTW) in which a limited number of heterogeneous vehicles, characterized by different capacities are available and the customers have a specified time windows for services. We propose a mixed integer programming formulation to model the problem. A feasible neighbourhood heuristic search is addressed to get the integer feasible solution after solving the continuous model of the problem.

Section 2 reviews the integer programming formulation of the PVRP with time windows from [20]. Section 3 describes the mathematics formulation of the (PVRFDPDTWP). Feasible neighbourhood heuristic search is given in Section 4. The algorithm is described in Section 5. Finally Section 6 describes the conclusions.

II. MODEL OF THE PVRPTW

The formulation of the model is adopted from Nguyen et al. (2011). The VRP is defined in graph $\Gamma = (\varsigma, E)$, where $\varsigma = \{0, 1, \dots, n\}$ is the vertex set and $E = \{(i,j): i,j \in \varsigma, i \neq j\}$ is the edge set. A travel cost c_{ij} is associated with every edge $(i,j) \in E$. The depot vertex is indexed by 0. $\varsigma_c = \varsigma \backslash \{0\}$ is the set of customer vertices. Each vertex $i \in \varsigma_c$ has a demand $q_i \geq 0$ on each day of the planning horizon of T days, a service time $s_i \geq 0$, a time window $[e_i, l_i]$, where e_i is the earliest time service may begin and l_i is the latest time, and requires a fixed number of visits f_i to be performed according to one of the allowable visit-day patterns in the list P_i. The time window specifying the interval vehicles leave and return to the depot is given by $[e_0, l_0]$. A fleet of m vehicles, each with capacity Q_k is based at the depot. Vehicles are grouped into set K. Vehicle routes are restricted to a maximum duration of D_k, $k = 1, \dots, m$.

Assume that the vehicle fleet is homogen with $Q_k = Q$ and a common duration restriction $D_k = D$, $\forall k = 1, \dots, m$. The PVRPTW can then be seen as the problem of generating (at most) m vehicle routes for each day of the planning horizon, to minimize the total cost over the entire planning horizon, such as 1) each vertex i is visited the required number of times, f_i, corresponding to a single pattern of visit days chosen from P_i, and is serviced within its time window, i.e., a vehicle may arrive before e_i and wait to begin service; 2) each route starts from the depot, visits the vertices selected for that day, with a total demand not exceeding Q, and returns to the depot after a duration (travel time) not exceeding D.

Let a_{rt} be 1 if day $t \in T$ belongs to pattern r, and 0 otherwise. Route-selection, pattern-selection, and continuous timing decision variables are used in the formulation:

$$x_{ijk}^t = \begin{cases} 1 \text{ if vehicle } k \in K \text{ traverses edge } (i,j) \in E \text{ on day } t \in T; \\ 0 \text{ otherwise;} \end{cases}$$

$$y_{ir} = \begin{cases} 1 \text{ if pattern } r \in R_i \text{ is assigned to customer } i \in V_c; \\ 0 \text{ otherwise.;} \end{cases}$$

w_{ik}^t indicates the service starting time for vehicle $k \in K$ at customer $i \in \varsigma_c$ on day $t \in T$

Let M be an arbitrary large constant. The PVRPTW can then be formulated as

$$\text{Minimize } \sum_{t \in T} \sum_{(i,j) \in E} \sum_{k \in K} c_{ij} x_{ijk}^t \quad (1)$$

$$\text{S.t.} \quad \sum_{r \in R_i} y_{ir} = 1, \quad \forall i \in V_c, \quad (2)$$

$$\sum_{j \in V} x_{ijk}^t = \sum_{j \in V} x_{jik}^t, \forall i \in V, k \in K, t \in T, \quad (3)$$

$$\sum_{k \in K} \sum_{j \in V} x_{ijk}^t = \sum_{r \in R_i} y_{ir} a_{rt}, \quad \forall i \in V_c, t \in T, \quad (4)$$

$$\sum_{k \in K} \sum_{j \in V} x_{0jk}^t \leq m, \ \forall t \in T, \quad (5)$$

$$\sum_{i,j \in S} x_{ijk}^t \leq |S| - 1, \ \forall S \subseteq V_c, k \in K, t \in T, \quad (6)$$

$$\sum_{j \in V_c} x_{0jk}^t \leq 1, \forall k \in K, t \in T, \quad (7)$$

$$\sum_{j \in V_c} q_i \sum_{j \in V} x_{ijk}^t \leq Q, \ \forall k \in K, t \in T, \quad (8)$$

$$w_{ik}^t + s_i + c_{ij} - M(1 - x_{ijk}^t) \leq w_{jk}^t, \quad \forall (i,j) \in E, k \in K, t \in T, \quad (9)$$

$$e_i \sum_{i,j \in S} x_{ijk}^t \leq w_{ik}^t \leq l_i \sum_{j \in V} x_{ijk}^t, \forall i \in V_c, k \in K \ t \in T, \quad (10)$$

$$w_{ik}^t + s_i + c_{i0} - M(1 - x_{i0k}^t) \leq D, \quad \forall i \in V_c, k \in K \ t \in T, \quad (11)$$

$$x_{ijk}^t \in \{0,1\}, \ \forall (i,j) \in E, k \in K, t \in T, \quad (12)$$

$$y_{ir} \in \{0,1\}, \ \forall i \in V_c, r \in R_i, \quad (13)$$

$$w_{ik}^t \geq 0, \forall i \in V_c, k \in K \ t \in T, \quad (14)$$

The objective function (1) minimizes the total travel cost. Constraints (2) ensure that a feasible pattern is assigned to each customer. Constraints (3) enforce flow conservation ensuring that a vehicle arriving at a customer on a given day, leaves that customer on the same day. Constraints (4) guarantee that each customer is visited on the days corresponding to the assigned pattern, while Constraints (5) make sure that the number of vehicles used on each day does not exceed m. Relations (6) are subtour elimination constraints. Constraints (7) ensure that each vehicle is used at most once a day, while (8) guarantee that the load charged on a vehicle does not exceed its capacity. Constraints (9) enforce time feasibility, i.e., vehicle k cannot start servicing j before completing service at the previous customer i and traveling from i to j, i.e., not before $w_{ik}^t + s_i + c_{ij}$. Constraints (10) ensure that customer time window restrictions are respected, while (11) constrains the route length. Constraints (12), (13), and (14) define the sets of decision variables.

III. MATHEMATICAL FORMULATION OF PVRFDPDTWP

We denote the planning horizon by T and the set of drivers by D. The set of workdays for driver $l \in D$ is denoted by $T_l \subseteq T$. The start working time and latest ending time for driver $l \in D$ on day $t \in T$ are given by g_l^t and h_l^t, respectively. Let D_I and D_E denote the set of the internal and external drivers ($D = D_I \cup D_E$). For each internal driver $l \in D_I$, let H denote the maximum weekly working duration. We denote the maximum elapsed driving time without break by F and the duration of a break by G (according to the EU driver legislation).

Let K denote the set of vehicles. For each vehicle $k \in K$, let Q_k and P_k denote the capacity in weight and in volume, respectively. We assume the number of vehicles equals to the number of drivers. Denote the set of n customers (/nodes) by $N = \{1, 2, \ldots, n\}$. Denote the depot by $\{0, n+1\}$. Each vehicle starts from $\{0\}$ and terminates at $\{n+1\}$. Each customer $i \in N$ specifies a set of days to be visited, denoted by $T_i \subseteq T$. On each day $t \in T_i$, customer $i \in N$ requests service with demand of q_i^t in weight and p_i^t in volume, service duration d_i^t and time window $[a_i, b_i]$. Note that, for the depot $i \in \{0, n+1\}$ on day t, we set $q_i^t = p_i^t = d_i^t = 0$. Denote the set of preferable vehicles for visiting customer i by K_i ($K_i \in K$) and the extra service time per pallet by e if a customer is not visited by a preferable vehicle. The travel time between customer i and j is given by c_{ij}. Denote the cost coefficients of the travel time of the internal drivers by A and the working duration of the external drivers by B.

We define binary variable x_{ijk}^t to be 1 if vehicle k travels from node i to j on day t, binary variable $wt\ i$ to be 1 if customer i is not visited by a preferred vehicle on day t. Variable v_{ik}^t is the time that vehicle k visits node i on day t. Binary variable z_{ik}^t indicates whether vehicle k takes a break after serving customer i on day t. Variable u_{ik}^t is the elapsed driving time for vehicle k at customer i after the previous break on day t. Binary variable y_{lk}^t is set to 1 if vehicle k is assigned to driver l on day t. Variables r_l^t and s_l^t are the total working duration and the total travel time for driver l on day t, respectively.

This notations used are given as follows :

Set:

T	The set of workdays in the planning horizon,
D_I	The set of internal drivers,
D_E	The set of external drivers,
D	The set of drivers $D = D_I \cup D_E$,
T_l	The set of workdays for driver $l \in D$,
K	The set of vehicles,
N	The set of customers,
N_0	The set of customers and depot $N_0 = \{0, n+1\} \cup N$,
K_i	The set of preferable vehicles for customer $i \in N$,
T_i	The set of days on which customer $i \in N$ orders,

Parameter:

Q_k	The weight capacity of vehicle $k \in K$,
P_k	The volume capacity of vehicle $k \in K$,
c_{ij}	The travel time from node $i \in N_0$ to node $j \in N_0$,
$[a_i, b_i]$	The earliest and the latest visit time at node $i \in N_0$,
d_i^t	The service time of node $i \in N_0$ on day $t \in T_i$,
q_i^t	The weight demand of node $i \in N_0$ on day $t \in T_i$,
p_i^t	The volume demand of node $i \in N_0$ on day $t \in T_i$,
e	The extra service time per pallet when a non-preferable vehicle is used,
$[g_l^t, h_l^t]$	The start time and the latest ending time of driver $l \in D$ on day $t \in T$,
α_i^t	Pick up quantity for customer i on day $t \in T_i$,
β_i^t	Delivery quantity for customer I on day $t \in T_i$,
H	The maximum working duration for each internal driver over the planning horizon,
F	The maximum elapsed driving time without break,
G	The duration of the break for drivers,
K_1	The cost factor on the total travel time of internal drivers,
K_2	The cost factor on the total working duration of the external drivers,

Variables:

x_{ilk}^t	Binary variable indicating whether vehicle $k \in K$ travels from node $i \in N_0$ to $j \in N_0$ on day $t \in T$,
w_i^t	Binary variable indicating whether customer $i \in N_0$ is visited by a non-preferable vehicle on day $t \in T$,
v_{ik}^t	The time at which vehicle $k \in K$ starts service at node $i \in N_0$ on day $t \in T$,

z_{ik}^t Binary variable indicating whether vehicle $k \in K$ takes break after serving node $i \in N_0$ on day $t \in T$,

u_{ik}^t The elapsed driving time of vehicle $k \in K$ at node $i \in N_0$ after the previous break on day $t \in T$,

y_{lk}^t Binary variable indicating whether vehicle $k \in K$ is assigned to driver $l \in D$ on day $t \in T$,

r_l^t The total working duration of driver $l \in D$ on day $t \in T$,

s_l^t The total travel distance of driver $l \in D$ on day $t \in T$,

θ_{jk}^t Number of pick up demand of customer j served by vehicle $k \in K$ on day $t \in T$

σ_{jk}^t Number of delivery demands of customer j served by vehicle $k \in K$ on day $t \in T$

The mathematical formulation for this problem is presented as follows:

$$\min \quad K_1 . \sum_{l \in D_I} \sum_{t \in T_i} s_l^t + K_2 . \sum_{l \in D_E} \sum_{t \in T_i} r_l^t + Z' \qquad (15)$$

Subject to:

$$\sum_{k \in K} \sum_{j \in N_0} x_{ijk}^t = 1, \forall i \in N, t \in T_i, \qquad (16)$$

$$\sum_{k \in K \setminus K_i} \sum_{j \in N_0} x_{ijk}^t = w_i^t, \forall i \in N, t \in T_i, \qquad (17)$$

$$\sum_{i \in N} \sum_{j \in N_0} q_i^t x_{ijk}^t \le Q_k, k \in K, t \in T, \qquad (18)$$

$$\sum_{i \in N} \sum_{j \in N_0} p_i^t x_{ijk}^t \le P_k, \quad k \in K, t \in T, \qquad (19)$$

$$u_{jk}^t \ge u_{ik}^t + c_{ij} - M(1 - x_{ijk}^t) - M z_{ik}^t \quad \forall i, j \in N_0, k \in K, t \in T, \qquad (20)$$

$$u_{jk}^t \ge c_{ij} - M(1 - x_{ijk}^t) \quad \forall i, j \in N, k \in K, t \in T \qquad (21)$$

$$u_{ik}^t + \sum_{j \in N_0} c_{ij} x_{ijk}^t - F \le M z_{ik}^t, \quad \forall i, j \in N_0, k \in K, t \in T, \qquad (22)$$

$$v_{jk}^t \ge v_{ik}^t + d_i^t + e \cdot p_i^t \cdot w_j^t + c_{ij} + G \cdot z_{ik}^t - M(1 - x_{ijk}^t) \quad \forall i, j \in N_0, \ k \in K, t \in T \qquad (23)$$

$$b_i \ge v_{ik}^t \ge a_i, \quad i \in N, \ k \in K, t \in T_i \qquad (24)$$

$$v_{0k}^t \ge \sum_{l \in D} (g_l^t \cdot y_{lk}^t) \ \forall k \in K, t \in T, \qquad (25)$$

$$v_{n+1,k}^t \le \sum_{l \in D} (h_l^t \cdot y_{ik}^t) \ \forall k \in K, t \in T, \qquad (26)$$

$$s_l^t \ge \sum_{i \in N_0} \sum_{j \in N_0} c_{ij} x_{ijk}^t - M(1 - y_{ik}^t), \quad \forall l \in D_I, k \in K, t \in T_i \qquad (27)$$

$$r_l^t \ge v_{n+1,k}^t - g_l^t - M(1 - y_{ik}^t), \quad \forall l \in D, k \in K, t \in T_i \qquad (28)$$

$$\sum_{t \in T_i} r_l^t \le H, \forall \ l \in D_I \qquad (29)$$

$$\sum_{k \in K} \theta_{jk}^t = \alpha_j^t, \forall j \in N, t \in T \qquad (30)$$

$$\sum_{k \in K} \sigma_{jk}^t = \beta_j^t, \ \forall j \in N, t \in T \qquad (31)$$

$$x_{ijk}^t, w_i^t, z_{ik}^t, y_{lk}^t \in \{0,1\} \quad , \forall i, j \in N_0, l \in D, \ k \in K, t \in T \qquad (32)$$

$$v_{ik}^t, u_{ik}^t, r_l^t, s_l^t \ge 0, \ \forall i, j \in N_0, l \in D, k \in K, t \in T \qquad (33)$$

$$\theta_{jk}^t, \sigma_{jk}^t \in \{0,1,2,\ldots\}, \forall j \in N, k \in K, t \in T, \qquad (34)$$

The objective function (15) minimizes weighted sum of the travel time of the internal drivers and the working duration of the external drivers over the planning horizon.

Constraints (16) state that each customer must be visited by one vehicle on each of its delivery days. Constraints (17) define whether each customer is visited by a preferable vehicle. Constraints (18-19) guarantee that the vehicle capacities are respected in both weight and volume.

Constraints (20-21) define the elapsed driving time. More specifically, for the vehicle (k) travelling from customer i to j on day t, the elapsed driving time at j equals the elapsed driving time at i plus the driving time from i to j (i.e., $u_{jk}^t \ge u_{ik}^t + c_{ij}$) if the vehicle does not take a break at customer i (i.e., $z_{ik}^t = 0$); Otherwise, if the vehicle takes a break at customer i (i.e., $z_{ik}^t = 1$), the elapsed driving time at j will be constrained by (15) which make sure it is greater than or equal to the travel time between i and j (i.e., $u_{jk}^t \ge c_{ij}$). Constraints (22) guarantee that the elapsed driving time never exceeds an upper limit F by imposing a break at customer i (i.e., $z_{ik}^t = 1$) if driving from customer i to its successor results in a elapsed driving time greater than F.

Constraints (23) determine the time to start the service at each customer. If j is visited immediately after i, the time v_{jk}^t to start the service at j should be greater than or equal to the service starting time v_{ik}^{t} at *i* plus its service duration d_i^t, the extra service time $e \cdot p_i^t$ if *i* is visited by an inappropriate vehicle (i.e., w_j^t = 1), the travel time between the two customers *cij* , and the break time *G* if the driver takes a break after serving *I* (i.e., z_{ik}^t = 1). Constraints (24) make sure the services start within the customers' time window.

Constraints (25-26) ensure that the starting time and ending time of each route must lie between the start working time and latest ending time of the assigned driver. Constraints (27) calculate the total travel time for each internal driver. Constraints (28) define the working duration for each driver on every workday, which equals the time the driver returns to the depot minus the time he/she starts work. Constraints (29) make sure that the internal drivers work for no more than a maximum weekly working duration, referred to as 37 week-hour constraints. Constraints (30 – 31) define the pick up and delivery for each customer. Constraints (32-34) define the binary and positive variables used in this formulation.

IV. NEIGHBOURHOOD SEARCH

It should be noted that, generally, in integer programming the reduced gradient vector, which is normally used to detect an optimality condition, is not available, even though the problems are convex. Thus we need to impose a certain condition for the local testing search procedure in order to assure that we have obtained the "best" suboptimal integer feasible solution.

Further Scarf [14] proposed a quantity test to replace the pricing test for optimality in the integer programming problem. The test is conducted by a search through the neighbours of a proposed feasible point to see whether a nearby point is also feasible and yields an improvement to the objective function.

Let $[\beta]_k$ be an integer point belongs to a finite set of neighbourhood $N([\beta]_k)$ We define a neighbourhood system associated with $[\beta]_k$, that is, if such an integer point satisfies the following two requirements

1. if $[\beta]_j \in N([\beta]_k)$ then $[\beta]_k \in [\beta]_j$, j $\neq$ *k*.
2. $N([\beta]_k) = [\beta]_k + N(0)$

With respect to the neighbourhood system mentioned above, the proposed integerizing strategy can be described as follows.

Given a non-integer component, x_k, of an optimal vector, x_B. The adjacent points of x_k, being considered are $[x_k]$ dan $[x_k] + 1$. If one of these points satisfies the constraints and yields a minimum deterioration of the optimal objective value we move to another component, if not we have integer-feasible solution.

Let $[x_k]$ be the integer feasible point which satisfies the above conditions. We could then say if $[x_k] + 1 \in N([x_k])$ implies that the point $[x_k] + 1$ is either infeasible or yields an inferior value to the objective function obtained with respect to $[x_k]$. In this case $[x_k]$ is said to be an "optimal" integer feasible solution to the integer programming problem. Obviously, in our case, a neigbourhood search is conducted through proposed feasible points such that the integer feasible solution would be at the least distance from the optimal continuous solution.

V. THE ALGORITHM

After solving the relaxed problem, the procedure for searching a suboptimal but integer-feasible solution from an optimal continuous solution can be described as follows.
Let

$$x = [x] + f, \quad 0 \le f \le 1$$

be the (continuous) solution of the relaxed problem, $[x]$ is the integer component of non-integer variable x and f is the fractional component.

Stage 1.

Step 1. Get row i^* the smallest integer infeasibility, such that $\delta_{i^*} = \min\{f_i, 1 - f_i\}$
(This choice is motivated by the desire for minimal deterioration in the objective function, and clearly corresponds to the integer basic with smallest integer infeasibility).

Step 2. Do a pricing operation
$v_{i^*}^T = e_{i^*}^T B^{-1}$

Step 3. Calculate $\sigma_{ij} = v_{i^*}^T \alpha_j$
With *j* corresponds to
$\min_j \left\{ \left| \frac{d_j}{\alpha_{ij}} \right| \right\}$
Calculate the maximum movement of nonbasic *j* at lower bound and upper bound.
Otherwise go to next non-integer nonbasic or superbasic *j* (if available). Eventually the column j^* is to be increased form LB or decreased from UB. If none go to next i^*.

Step 4.
Solve $B\alpha_{j^*} = \alpha_{j^*}$ for α_{j^*}

Step 5. Do ratio test for the basic variables in order to stay feasible due to the releasing of nonbasic j^* from its bounds.

Step 6. Exchange basis

Step 7. If row $i^* = \{\varnothing\}$ go to Stage 2, otherwise
Repeat from step 1.

Stage 2. Pass1 : adjust integer infeasible superbasics by fractional steps to reach complete integer feasibility.
Pass2 : adjust integer feasible superbasics. The objective of this phase is to conduct a highly lovalized neighbourhood search to verify local optimality.

VI. CONCLUSIONS

This paper was intended to develop efficient technique for solving one of the most economic importance problems in optimizing transportation and distribution systems. The aim of this paper was to develop a model of Periodic vehicle Routing with Time Windows, Fleet and Driver Scheduling, Pick-up and Delivery Problem This problem has additional constraint which is the limitation in the number of vehicles. The proposed algorithm employs nearest neighbor heuristic algorithm for solving the model. This algorithm offers appropriate solutions in a very small amount of time

REFERENCES

[1] Angelelli. E., & Speranza. M.G, “The periodic vehicle routing problem with intermediate facilities”. *European Journal of Operational Research* 137, 2, 233–247, 2002.

[2] B. L. Golden, A. A. Assad, and E. A. Wasil, “Routing Vehicles in the real world: Applications in the solid waste, beverage, food, dairy, and newspaper industries”. In P. Toth and D. Vigo, editors, *The Vehicle Routing Problem,* pages 245-286. SIAM, Philadelphia, PA, 2002.

[3] B. L. Golden, S. Raghavan, and E. A. Wasil, “The vehicle routing problem: latest advances and new challenges”. Springer, 2008.

[4] Beltrami, E., Bodin, L., “Networks and vehicle routing for municipal waste collection”. Networks 4(1), 65—94, 1974.

[5] Blakely, F., Bozkaya, B., Cao, B., Hall, W., Knolmajer, J., “Optimizing periodic maintenance operations for Schindler Elevator Corporation”. Interfaces 33(1), 67—79, 2003.

[6] Chao, I.M., Golden, B., Wasil, E., “An improved heuristic for the period vehicle routing problem”. Networks 26(1), 25—44, 1995.

[7] Christofides, N., Beasley, J., “The period routing problem”. Networks 14(2), 237—256, 1984.

[8] Cordeau, J.F., Gendreau, M., Laporte, G., “A Tabu Search heuristic for periodic and multi-depot vehicle routing problems”. Networks 30(2), 105—119, 1997.

[9] Daganzo, C. F., “Modeling distribution problems with time windows: Part II”. *Transportation Science* 21(3), 180—187, 1987.

[10] Dji constantinuo. Ha, & Baldacci. E. “A multi-depot period vehicle routing problem arising in the utilities sector”. *Journal of Operations Research Society*, 49, 12, 1239–1248, 1998.

[11] Fisher. M.L., & Jaikumar. R., “A generalized assignment heuristic for vehicle routing”. Networks,11, 109±124, 1981.

[12] Francis, P., Smilowitz, K., Tzur, M., “The period vehicle routing problem with service choice”.*Transportation Science*, 2004.

[13] G. Laporte and F. Sement. Classical Heuristics for the Vehicle Routing Problem. In Toth, P. and Vigo, D., editors, *The vehicle Routing Problem,* SIAM Monographs on Discrete Matheamtics and Applications, pages 109-128. SIAM, Philadelphia, PA, 2002.

[14] H. E Scarf., *Testing for optimality in the absence of convexity* in Walter P Heller, Ross M Starr and David A Starrett (Eds) (Cambridge University Press) pp 117-134, 1986.

[15] J. Alegre, M. Laguna, J. Pacheco, “Optimizing the periodic pick-up for raw materials for a manufacturer of auto parts”, *European Journal of Operational Research* 179, 736-746, 2007.

[16] J.-F. Cordeau, G. Laporte, M.W.F. Savelsbergh, and D. Vigo. Vehivle Routing. In Barnhart, C. and Laporte, G., edotirs, *Transportation,* handbooks in Operation Research and Management Science, Pages 367-428. North-Holland, Amsterdam, 2007.

[17] J. Pacheco, A. Alvarez, I. García, F. Angel-Bello, “Optimizing vehicle routes in a bakeri company allowing flexibility in delivery dates”, *Journal of Operationsl Research Society* http://dx.doi.org/10.1057/jors.2011.51, 2011.

[18] O. Bräysy and M. Gendreau. “Vehicle Routing Problem with Times Windows, Part II: Metaheuristics”. *Transportation Science,* 39(1): 104-118, 2005.

[19] P. Francis, K. Smilowitz, M. Tzur, “The period vehicle routing problem and its extensions”, in : The Vehicle Routing Problem: Latest Advances and New Challenges, Vol. 43, Springer, 2008 (Chapter).

[20] P. K. Nguyen, T. G. Crainic, M. Toulouse, “A Hybrid Genetic Algorithm for the Periodic Vehicle Routing Problem with Time Window”. *CIRRELT*-2011-25, 2011.

[21] P. Toth and D. Vigo. “The Vehicle Routing Problem Society for Industrial and Applied Matheamtics”, Philadelpiha, PA, 2002.

[22] R. Baldacci, E. BArtolini, A. Mingozzi, A. Valletta, “An Exact algorithm for the period routing problem”, *Operations research* 59 (1) 228-241, 2011.

[23] R. Tavakkoli Moghaddam, A. Mohammad Zohrevand, K. Rafiee, “Solving a New Mathematical Model for a Periodic Vehicle Routing Problem by Particle Swarm Optimization”, *Trasportation Research Journal,* Vol. 2, No. 1, 2012/77-87, 2012.

[24] Russell, R., Igo, W., “An assignment routing problem”. Networks 9(1), 1—17, 1979.

[25] Russell. R.A., & Gribbin. D., “A multiphase approach to the period routing problem”. Networks, 21, 747±765, 1991.

[26] Smilowitz, K., Daganzo, C. F., “Cost modeling and solution techniques for complex transportation systems”. *IEMS Working paper* 04-006, Northwestern University, 2004.

[27] Tan. C.C.R., & Beasley. J.E., “A heuristic algorithm for the period vehicle routing problem”. *OMEGA, International Journal of Management Scienc*e, 12, 5, 497±504, 1984.

[28] V. Cacchiani, V. C. Hemmelmayr, F. Tricoire, “A set-covering based heuristic algorithm for the periodic vehicle routing problem”, *Discrete Applied Matehametics* , doi:10.1016/j.dam.2012.08.032, 2012.

[29] V. Hemmelmayr, K. Doerner, R. Hartl, M. Savelsbergh, “Delivery strategis for blood products supplies”, *OR Spectrum* 31 (4) 707-725, 2009.

[30] G.B. Dantzig and J.H. Ramser, “The truck dispatching problem”, Management Science, Vol. 6, 1959, 80.

Bambang Irawan Currently, he is a PhD student from Graduate School of Mathematics, University of Sumatera Utara. He received his Masters from University Technology Mara, Malaysia. He is a senior staff in the department of Mathematics, University of Sumatera Utara, Medan, Indonesia.

Herman Mawengkang, He is a professor of Operations Research at the University of Sumatera Utara. He gained his PhD from the University of New South Wales, Sydney, Australia. He already produced many journal papers published internationally. Currently he is the chairman of Graduate School of Mathematics, University of Sumatera Utara.

.

