I Congresso de Estatística e Investigação Operacional da Galiza e Norte de Portugal
VII Congreso Galego de Estatística e Investigación de Operacións
Guimarães 26, 27 e 28 de Outubro de 2005

SOLVING THE TRAVELING REPAIRMAN PROBLEM WITH DIFFERENTIATED WAITING TIMES THROUGH LAGRANGIAN RELAXATION

Ana Maria Rocha[1], João Soares[2] and Edite M.G.P. Fernandes[1]

[1]Departamento de Produção e Sistemas
Universidade do Minho
[2]Departamento de Matemática
Universidade de Coimbra

ABSTRACT

In this paper we present a formulation of the traveling repairman problem with differentiated waiting times that is derived from the extended disaggregated flow formulation for the asymmetric traveling salesman problem. We focus on the usage of the Lagrangian approach as a mechanism of speeding up the solution of the linear relaxation by a simplex method. We show some computational results.

1. INTRODUCTION

The traveling repairman problem with differentiated waiting times is a variation of the traveling repairman problem, which is also known in the literature as delivery man problem, minimum latency problem and traveling salesman problem with cumulative costs. The latter aims to find a tour, or an Hamiltonian cycle, on a given, directed or undirected, graph that minimizes the total traveling time plus the waiting time sum across every customer (relatively to some fixed node, say node 1, the repairman home). The waiting time, or latency, of customer k is the total time involved in the path 1 to k in the tour and can also be thought of as delay for service. The traveling time is sometimes viewed as latency of the repairman.

The model that we propose accounts for a relative measurement of waiting time associated with each arc. We assume there are n time costs $c_{ij}^k, k = 1, 2, 3, \dots, n$, associated with each arc (i, j) in the given undirected graph. The scalar c_{ij}^1 is the traveling time of arc (i, j) while c_{ij}^k $(k \geq 2)$ is the k-th customer perception of waiting time associated with arc (i, j). The proposed problem is to find a tour that minimizes the total traveling time plus the perceived waiting time sum across every customer.

The traveling repairman problem is a special case of the proposed problem – the one in which c_{ij}^k is independent of k. The minimum weighted latency problem is also a special case – the one in which $c_{ij}^k = w_k c_{ij}$, where w_k is a weight value associated with customer k. The traveling salesman problem is also a special case - the one in which $c_{ij}^k = 0$, for every $k \geq 2$.

Approximation algorithms for the traveling repairman problem are analyzed in, for example, Arora and Karakostas (2003). Exact approaches are analyzed in Fischetti, Laporte and Martello (1993) and Wu, Huang and Zhan (2004). Here we are concerned with the computational finding of near optimal solutions for a more general model. This work is part of an ongoing research project in which the authors are engaged to study the usage of fast approximation schemes to speeding up the optimal basis identification in simplex algorithms. Problems of particular interest are the linear programming relaxations of difficult combinatorial optimization problems, namely, those that (a) are large in the number of variables and/or constraints, and (b) have an embedding complicating characteristic that if removed makes the problem "easy". These characteristics are found in the proposed problem.

Our mathematical formulation of the traveling repairman problem with differentiated waiting times is derived from the extended disaggregated flow formulation for the asymmetric traveling salesman problem (ATSP). This formulation has $\mathcal{O}(n^3)$ variables and $\mathcal{O}(n^2)$ constraints, which difficults the solving of linear programming relaxations. For a simplex method, the basis can be so large that the method hardly reaches the optimal basis. For interior point methods, the difficulty lies in memory requirements. In this context, it seems natural to approach the solving, exactly or approximately, of linear programming relaxations through Lagrangian relaxation.

In this work, we empirically analyze the use of the volume algorithm (VA), a variation of the subgradient algorithm proposed by Barahona and Anbil (2000), on a Lagrangian relaxation of the proposed problem that stems from dualizing flow constraints. Lagrangian subproblems are assignment problems, which are efficiently reoptimized. Emphasis is put in the use of the solution output from VA as a means of deriving a good initial basis. Computational results were obtained for a selection of problems from TSPLib.

The article is structured as follows. Section 2 introduces the disaggregated flow formulation and presents the Lagrangian problem. Section 3 recalls the volume algorithm especially adapted to our study. Section 4 presents numerical experiments and Section 5 contains the conclusions.

2. THE DISAGGREGATED FLOW FORMULATION

Let $G = (V, E)$ be an undirected graph with vertex set $V = \{1, 2, \ldots, n\}$ and arc set $E \subseteq V \times V$ of cardinality m. The graph is assumed simple and Hamiltonian. There are known costs c^k_{ij}, for each $k \in V$, associated with each arc (i, j). Node 1 is the repairman home while the other nodes are customer locations. The traveling repairman problem with differentiated waiting times is to find a tour H in G defined by $H = \{1, (1 \equiv i_1, i_2), i_2, (i_2, i_3), i_3 \ldots, i_n, (i_n, i_{n+1} \equiv 1), 1\}$, that minimizes the sum of the tour traveling time with the total perceived waiting time sum across every customer. Mathematically, the total costs function is

$$c(H) \equiv \sum_{(i,j)\in E(H)} c^1_{ij} + \sum_{k=2}^{n} \left(\sum_{l=1}^{k-1} c^{i_k}_{i_l i_{l+1}} \right) = \sum_{k=1}^{n} \left(\sum_{l=1}^{k} c^{i_{k+1}}_{i_l i_{l+1}} \right). \tag{1}$$

The set of characteristic vectors of tours in G is a subset of the set of integer vectors x that satisfy

$$Ax = \mathbb{1}, x \geq 0, \tag{2}$$

where A denotes the node-edge incidence $(n \times m)$ matrix of the undirected bipartite graph $G' = (V \times V, E)$ and $\mathbb{1}$ is a column-vector of all-ones. The following lemma completes the algebraic characterization of tours in G.

Lemma 1 (Claus (1984)) *Let x be an integral vector satisfying (2). Then, x is the incidence vector of a tour in G if and only if there are vectors y^k, for every $k \in V_1 \equiv V \setminus \{1\}$, satisfying the following system of equalities and inequalities*

$$By^k = b^k \qquad (k \in V_1) \tag{3a}$$

$$0 \leq y^k \leq x \qquad (k \in V_1). \tag{3b}$$

where B denotes the node-arc incidence ($n \times m$) matrix of the undirected graph G, and, for each $k \in V_1$, b^k is a column-vector of all zeros except for $b_1^k = -1$ and $b_k^k = 1$. For each arc (i, j), each column of B has $+1$ at position i and -1 at position j.

Constraints (2) and (3) define the *extended disaggregated flow formulation* of the ATSP proposed by Claus (1984). The projection of the underlying polyhedron onto the m-dimensional space of the x variables defines the formulation of Dantzig, Fulkerson and Johnson (1954) that has an exponential number of constraints. The following theorem shows that (2) and (3) also define a formulation for the proposed problem.

Theorem 1 *Let (x, y) be an integral vector. Then, x is the characteristic vector of a tour H in G and each y^k, for $k \in V_1$, is the characteristic vector of the path from 1 to k in H if and only if (2) and (3) holds.*

Proof: Let $(\bar{x}, \bar{y})$ be such that $\bar{x}$ is the characteristic vector of a tour H in G and each $\bar{y}^k$, for $k \in V_1$, is the characteristic vector of the path from 1 to k in H. Clearly, (2) and "$y^k \leq x, k \in V_1$" are satisfied. Moreover, $\bar{y}^k$ is an extreme point of the polyhedron $P_k \equiv \left\{y^k \colon By^k = b^k, y^k \geq 0\right\}$. Thus, one of the inclusions is proved. Reciprocally, let $(\bar{x}, \bar{y})$ be integral and belonging to the polyhedron Q defined by (2) and (3). Thus, $(\bar{x}, \bar{y})$ is an extreme point of Q. In particular, $\bar{x}$ is an extreme point of $P_1 \equiv \{x \colon Ax = \mathbb{1}, x \geq 0\}$ and each $\bar{y}^k$ is an extreme point of the polyhedron $P'_k \equiv \left\{y^k \colon By^k = b^k, 0 \leq y^k \leq \bar{x}\right\}$. The first remark implies that $\bar{x}$ is the characteristic vector of a tour H in G (from Lemma 1, $\bar{x}$ cannot define subtours). The second remark implies that $\bar{y}^k$ is the characteristic vector of a trail from 1 to k in H. Since H is a tour and $k \neq 1$, the trail must be a path. □

Hence, the traveling repairman problem with differentiated waiting times can be formulated as the following integer program

$$z_I^* = \min \left\{ c^1 x + \sum_{k \in V_1} c^k y^k \colon (x, y) \text{ satisfies (2), (3) and integrality} \right\} \tag{4}$$

that has n^3 nonnegative variables and $2n + nm$ constraints. Note that moderated size graphs may originate very large models. Table 1 summarizes the behavior of CPLEX 7.0, with default settings, on solving the LP relaxations of a few of the small instances from TSPLIB understood as instances of the Traveling Repairman Problem (TRP, for short). Each problem is defined in a complete digraph and the arc costs are defined as follows $c_{ij}^k = \lfloor c_{ij} \xi / 100 \rfloor$, where for each pair $(k, (i, j))$, ξ is a pseudo-random number in the interval $[80, 120]$ using the zero as seed.

Our work is motivated by the difficulty of the dual-simplex method on identifying the optimal basis. We will try the volume algorithm within a Lagrangian relaxation framework to rapidly define a better initial basis.

Consider the Lagrangian problem that arises from dualizing the flow balance constraints (3a),

$$z_L^* = \max \left\{ z(\pi) : \pi = \left(\pi^k\right) \in \mathbb{R}^{(|V|-1) \times |V|} \right\} \tag{5}$$

where

$$z(\pi) \equiv \min \left\{ c^1 x + \sum_{k \in V_1} c^k y^k + \sum_{k \in V_1} \pi^k \left(b^k - By^k\right) : (x, y) \text{ satisfies (2) and (3b)} \right\} \tag{6}$$

TRP INSTANCE				CPLEX(dualopt)	
ID	$\|V\|$	$\|E\|$	Frac. Optimal	Simplex It.	Time (sec.)
ftv33	34	1122	6884.3	17982	107.1
ftv35	36	1260	7794.0	18388	123.8
ftv38	39	1482	8046.4	23451	205.0
ftv44	45	1980	9195.7	31171	409.7
ftv47	48	2256	10937.3	44127	870.3
ftv55	56	3080	10364.2	60673	1587.9
ftv64	65	4160	12021.0	120436	5243.3
ftv70	71	4970	13287.0	151253	7830.5
ry48p	48	2256	89554.4	75635	1962.6
ft53	53	2756	45250.6	94492	2848.9
ft70	70	4830	185080.0	73176	8139.8

Table 1: CPLEX 7.0 performance, with default settings.

and z_L^* is the optimal value of the LP relaxation of (4), because the integrality property holds. This Lagrangian subproblem can be solved rapidly and can be efficiently reoptimized. In fact, an optimal solution in (6) can be derived from solving a single assignment problem as formally proved below.

Theorem 2 *Let* $\bar{c} = \left[\bar{c}_{ij} \equiv c^1_{ij} + \sum_{k\in V_1}\left(c^k_{ij} - \pi^k_i + \pi^k_j\right)^-\right]$. *Then,* $(\bar{x}, \bar{y})$ *is optimal in (6) if and only if* $\bar{x}$ *is optimal for the assignment problem*

$$\min\{\bar{c}x \colon x \text{ satisfies (2)}\} \tag{7}$$

and $\bar{y}$ *is componentwise defined by*

$$\bar{y}^k_{ij} = \left\{\begin{array}{ll} \bar{x}_{ij} & \text{if } c^k_{ij} - \pi^k_i + \pi^k_j < 0, \\ \xi \in [0, \bar{x}_{ij}] & \text{if } c^k_{ij} - \pi^k_i + \pi^k_j = 0, \\ 0 & \text{if } c^k_{ij} - \pi^k_i + \pi^k_j > 0. \end{array}\right\} \quad ((i,j)\in E, k \in V_1)\,. \tag{8}$$

Proof: For any $\bar{x}$ fixed, but satisfying $Ax = 1\!\!1, x \geq 0$, the set of y optimal solutions in (6) is characterized by (8). Hence, the optimal value of (6) equals that of (7), which proves one implication. Reciprocally, if $(\bar{x}, \bar{y})$ is optimal for (6) then $\bar{y}$ must be of the form (8). For such a fixed $\bar{y}$, $\bar{x}$ must be optimal in (7). □

Problem (7) can be solved by the Hungarian method in $\mathcal{O}(n^3)$ operations but, due to small changes in objective function and reoptimization, this number will be greatly reduced. In the next section we will recall the volume algorithm and explain how it was particularly tuned to solving (5) so that the reader can, in principle, replicate our numerical findings.

3. THE VOLUME ALGORITHM

The dual problem (5) needs to be solved by nonsmooth optimization tools that use subgradients, cutting-planes or analytic centers. All these tools have advantages and drawbacks, and their efficiency depends on the problem characteristics. The volume algorithm of Barahona and Anbil (2000) incorporates in the subgradient method ideas from the early developments of the bundle method namely, in each iteration, the direction of movement is a convex combination of the new and previous subgradients. The idea aims at avoiding the typical zig-zagging behavior of the subgradient algorithm and, in this way, accelerating solution convergence.

Numerical testing with the volume algorithm has produced "optimal" primal solutions that are "close" to satisfy the dualized constraints within a Lagrangian framework. This claim is purely empirical. Numerical experiments on some linear programming problems show that the volume algorithm is quite reliable in producing "near optimal" solutions for linear programming relaxations. The volume algorithm is formally described in Figure 1.

Input: $\pi^0 \in \mathbb{R}^{(|V|-1)\times|V|}$.
Initialization: Let (x^0, y^0) be optimal in (6) for $\pi = \pi^0$.
Compute $v^0 = b - By^0 \in \partial z(\pi^0)$.
Set $\bar{\pi}^1 = \pi^0$, $(\bar{x}, \bar{y}) = (x^0, y^0)$, $\bar{w} = v^0$, $j = 1$ and $l = 1$.
Generic Iteration j:
Step 1: Choose a stepsize $s_j > 0$ and compute $\pi^j = \bar{\pi}^l + s_j\bar{w}$.
Step 2: Let (x^j, y^j) be optimal in (6) for $\pi = \pi^j$ and let $v^j = [v^{k,j} \equiv b^k - By^{k,j}] \in \partial z(\pi^j)$.
Step 3: Choose $\alpha_j \in [0, 1]$ and update

$$(\bar{x}, \bar{y}, \bar{w}) \quad \leftarrow \quad \alpha_j \left(x^j, y^j, v^j\right) + (1 - \alpha_j)\,(\bar{x}, \bar{y}, \bar{w})$$

Step 4: If $z(\pi^j) > z(\bar{\pi}^l)$, then update $\bar{\pi}^{l+1} = \pi^j$ and set $l \leftarrow l + 1$.
Step 5: Test stopping criterion. Let $j \leftarrow j + 1$ and go to *Step 1.*

Figure 1: The volume algorithm (VA)

We have used in our numerical experiments the implementation available at the COIN-OR webpage, `http://www.coin-or.org/`. Our choices for the scalars s_j and α_j are essentially empirical and similar to the ones used in Barahona and Anbil (2000).

The algorithm terminates when each constraint in $b - By$ is violated by at most 0.01 and the relative difference between the lower bound (L) and the value of the primal objective function (P) is less than 1%.

4. COMPUTATIONAL RESULTS

This section summarizes the numerical experiments. All the codes were implemented in C++ and all the computational tests were performed on a PC with a 3GHz Pentium IV microprocessor and 1Gb of memory running RedHat Linux 9. We use the solution output from the volume algorithm as input to the dual-simplex method. In each iteration of the volume algorithm, the assignment problem (7) is solved by the dual-simplex method of Cplex. After VA, Cplex is left the task of identifying an initial basis with the primal vector (x, y) output from VA.

Table 2 summarizes our results. Column 1 identifies the TRP instance. Columns 2-6 report the VA results, namely: the number of iterations, the lower bound (based on dual information), the primal objective function value at the last primal solution found; the l_∞ constraint violation value at this last solution, and the time spent. Columns 7-8 have to do with Cplex and are as follows: the number of simplex iterations required by the dual simplex method and the time in seconds. The last two columns are the total time spent running the two algorithms and the percentage of the gain in time with this combined strategy (VA+CPLEX) as compared to the execution of CPLEX (dualopt) reported in Table 1. The registered times of all the tables were rounded to one decimal place.

TRP ID	**VOLUME ALGORITHM**					**CPLEX**		**Total Time**	**Time Gain**
	AV It.	L (dual)	P (primal)	Max. violation	Time (sec.)	Simp. It.	Time (sec.)		
ftv33	807	6872.5	6928.9	0.0099987	6.0	1285	11.2	17.2	89.5%
ftv35	582	7780.7	7807.4	0.0098364	5.1	950	9.6	14.7	88.1%
ftv38	2859	8043.9	8071.0	0.0099988	29.5	1091	14.8	44.3	78.4%
ftv44	548	9175.1	9209.1	0.0098912	9.0	1971	40.6	49.7	87.9%
ftv47	5527	10936.1	10929.5	0.0099988	101.2	1662	42.8	144.0	83.5%
ftv55	485	10320.3	10390.5	0.0099999	15.3	3616	145.5	160.8	89.9%
ftv64	534	11989.2	12031.6	0.0099786	26.0	6179	390.9	416.9	92.0%
ftv70	527	13242.1	13338.7	0.0096567	34.9	11233	857.5	892.4	88.6%
ry48p	478	89065.9	89920.7	0.0096508	11.3	7004	179.3	190.5	90.3%
ft53	2134	45205.2	45657.0	0.0054329	57.1	4918	172.2	229.3	92.0%
ft70	5303	185055.0	186905.0	0.0068805	417.4	3933	737.1	1154.4	85.8%

Table 2: CPLEX performance after VA.

Some conclusions may be drawn. In all cases, an improved performance of at least 75% was obtained. We can see that for problems ftv38 and ftv47, VA executes a large number of iterations for reaching the stopping criterion and the time spent by VA dominates over the CPLEX time. Probably, a limitation on the number of VA iterations could give a gain in time even greater.

5. CONCLUSIONS

Our study showed that the volume algorithm is a viable mechanism of deriving a good initial basis for the solving of a polynomial formulation of the TRP like the extended disaggregated flow formulation by a simplex algorithm.

Of course our numerical experiments were limited and further testing is required especially on even larger models.

6. REFERENCES

Arora, S. and Karakostas, G. (2003). "Approximation schemes for minimum latency problems". *SIAM J. Comput.* 32, 1317-1337.

Barahona, F and Anbil, R. (2000). "The volume algorithm: Producing primal solutions with a subgradient method". *Mathematical Programming* 87, 385-399.

Claus, A. (1984). "A new formulation for the travelling salesman problem". *SIAM Journal of Algebraic and Discrete Methods* 5, 21-25.

Dantzig, G., Fulkerson, D. and Johnson, S. (1954). "Solution of a large-scale traveling salesman problem". *Operations Research* 2, 393-410.

Fischetti, M., Laporte, G. and Martello, S. (1993). "The delivery man problem with cumulative matroids". *Oper. Res.* 41, 1055-1064.

Wu, B. Y., Huang, Z.-N. and Zhan, F.-J. (2004). "Exact algorithms for the minimum latency problem". *Inform. Process. Lett.* 92, 303-309.