REPORTS
IN
INFORMATICS

ISSN 0333-3590

Designing delivery districts for the vehicle routing problem with stochastic demands

Dag Haugland, Sin C. Ho and Gilbert Laporte

REPORT NO 260 **December 2003**

Department of Informatics

UNIVERSITY OF BERGEN

Bergen, Norway

This report has URL http://www.ii.uib.no/publikasjoner/texrap/ps/2003-260.ps

Reports in Informatics from Department of Informatics, University of Bergen, Norway, is available at http://www.ii.uib.no/publikasjoner/texrap/.

Requests for paper copies of this report can be sent to:

Department of Informatics, University of Bergen, Høyteknologisenteret, P.O. Box 7800, N-5020 Bergen, Norway

Designing delivery districts for the vehicle routing problem with stochastic demands

Dag Haugland* Sin C. Ho* Gilbert Laporte†

Abstract

This paper considers the problem of designing districts for vehicle routing problems with stochastic demands. In particular, demands are assumed to be uncertain when the districts are being made, and these are revealed only after the districting decisions are determined. Tabu search and descent heuristics are developed and compared. Computational results are reported.

Keywords: districting, vehicle routing, tabu search, stochastic programming

*Department of Informatics, University of Bergen, N-5020 Bergen, Norway. {dag,sin}@ii.uib.no

†Canada Research Chair in Distribution Management, HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7. gilbert@crt.umontreal.ca

1 Introduction

Districting involves partitioning customer locations into districts according to some criteria. In general, these criteria include contiguity and balance of the districts. Each district is independently responsible for the operations performed within its border. One of the operations is the determination of vehicle routes. From a planning perspective, there is a fundamental difference between districting and routing. While districting decisions are made at a strategic or tactical level of management, routing decisions are operational, i.e. these decisions are made quite often. We believe that it is possible to achieve long-term savings if one is careful with the design of the districts because it does have an impact on routing. Some typical applications of districting include the design of political districts (Bozkaya et al. 2003), sales territories (Fleischmann and Paraschis 1988, Skiera and Albers 1998, Drexl and Haase 1999), emergency and health care districts (Pezzella et al. 1981, Blais et al. 2003), school districts (Ferland and Guénette 1990), police districts (D'Amico et al. 2002), etc. In this paper we focus on the design of delivery districts for the vehicle routing problem with stochastic demands.

The *Vehicle Routing Problem* (VRP) is defined on a complete graph $\mathcal{G} = (\mathcal{V}', \mathcal{A})$, where $\mathcal{V}' = \{0, 1, \ldots, n\}$ is the set of vertices representing customers, with 0 denoting the depot, and $\mathcal{A} = \{(i, j) : i, j \in \mathcal{V}', i \neq j\}$ is the arc set. With every vertex $i > 0$ is associated a nonnegative demand D_i. Each arc (i, j) is associated with a traveling cost c_{ij}. A fleet of m homogenous vehicles of capacity q is stationed at the depot. The VRP consists of designing m delivery routes such that (i) every route starts and ends at the depot; (ii) every customer is visited exactly once; and (iii) the total demand of any vehicle route does not exceed q. An overview can be found at Laporte and Semet (2002) and Gendreau et al. (2002).

In the *Stochastic Vehicle Routing Problem* (SVRP), demands D_i are random variables. Vehicle routes are planned in the first stage, and demands are only revealed later. As a result, some vehicle routes may violate the capacity constraints, i.e. route failures may occur. There exist different recourse policies to handle these situations. One possible policy is to follow the planned route until the vehicle is emptied, return to the depot to replenish, and then continue with the deliveries at the customer on the planned route where the failure occurred. Another policy is to plan "preventive breaks" along the planned route. A survey on the SVRP can be found in Gendreau et al. (1996).

In this paper we consider the problem of partitioning a set of customers into contiguous districts such that all customers within the same district are serviced by exactly one vehicle. The allocation of customers to districts is constant over time, even if an occurring demand pattern indicates that a regrouping of the customers is cost effective. The actual demand is observed only after the districts have been defined, and a travel plan must be produced within each district. This approach makes sense from a managerial point of view since districts boundaries tend to remain the same over time while customer demand varies from day to day.

Since we assume that routing decisions are not fixed until the demand is observed, the VRP to be solved in the route planning phase is deterministic. Demands are revealed before the vehicles leave the depot, and our recourse policy is to plan breaks so as to minimize the routing cost. In the first stage, however, demand must be considered as a stochastic variable, and the goal is to define the districts in such a way that the expected total travel cost is minimized. Solutions fulfilling this goal typically consist of districts leaving a fair amount of flexibility in the routing decisions, that is an ability to adjust to the actual demand. The most flexible partition of the customers is obviously obtained by defining one all-inclusive district and leave the others empty. This would however lead to extreme expected travel cost for one district, whereas there are no cost associated with the others. In order to balance the districts more evenly, we impose the constraint that the districts have to be constructed in such a way that the expected travel cost for each district does not exceed a given upper bound.

The remainder of the paper is organized as follows. Section 2 provides a formulation of the model. The approach of approximating the expected cost is described in Section 3. The tabu search heuristic and descent heuristic are presented in Sections 4 and 5, respectively. This is followed by computational results in Section 6, and by the conclusions in Section 7.

2 Model formulation

The problem just described can be modeled as a *two-stage stochastic program with recourse*. In the first stage, the districting decisions are made, while in the second stage one VRP is solved for each district. The constraints that apply to the composition of a district are that the expected travel cost is not larger than a given number, and that the total expected demand does not exceed the capacity.

Let $V = \{1, \ldots, n\}$ be the set of customers, let q denote the vehicle capacity, for any $u, v \in V$ let c_{uv} denote the cost of traveling directly from u to v, and assume $c_{uw} \leq c_{uv} + c_{vw}$ for all $u, v, w \in V$. Let D_v $(v = 1, \ldots, n)$ be the demand from customer v, assume that $D_1, \ldots, D_n$ are independent stochastic variables, and that for all $v = 1, \ldots, n$, $D_v - \underline{d}_v$ has a binomial distribution with parameters $\overline{d}_v - \underline{d}_v$ and p. Denoting by $P(\mathcal{E})$ the probability of event $\mathcal{E}$, we thus have $P(D_v = d) = p^{d-\underline{d}_v}(1-p)^{\overline{d}_v - d} \binom{\overline{d}_v - \underline{d}_v}{d - \underline{d}_v}$, and the sample space of the demand vector $D = (D_1, \ldots, D_n)$ becomes the hyperrectangle $\Omega = [\underline{d}_1, \overline{d}_1] \times \cdots \times [\underline{d}_n, \overline{d}_n]$. Here $p \in [0, 1]$, and $\underline{d}_v \leq \overline{d}_v$ $(v = 1, \ldots, n)$ are given integers. For $S \subseteq V$, let $f(S, d)$ denote the optimal travel cost in the VRP defined on the customer set S given the demand $d \in \Omega$, and let $VRP(S, d)$ denote the problem instance itself. Let h be the fixed cost associated with a district, which can be viewed as the cost of using a vehicle, and define $\chi(S)$ to be 1 if $S \neq \emptyset$ and 0 otherwise. Imposing the bound t on travel cost in each district, we arrive at the model

$$
\begin{aligned}
\min_{S_1, \ldots, S_m} \quad & \textstyle\sum_{j=1}^m (E_D f(S_j, D) + h\chi(S_j)) && (1) \\
\text{s.t.} \quad & \{S_1, \ldots, S_m\} \text{ is a partition of } V && (2) \\
& f(S_j, \overline{d}) \leq t \qquad\qquad j = 1, \ldots, m. && (3)
\end{aligned}
$$

Note that the removal of (3) makes $\{S_1, \ldots, S_m\} = \{V, \emptyset, \ldots, \emptyset\}$ an optimal solution (see the discussion in Section 1). It can however be argued that (3) could be relaxed to either the *probabilistic* constraint $P(f(S_j, D) \leq t) \geq 1 - \alpha$ or to the bound $E_D f(S_j, D) \leq t$. However, complying with the conventional format of two-stage recourse problems, we introduce neither probabilistic constraints nor constraints involving expectations.

A two-stage stochastic program is said to have *complete recourse* if the second stage problem is feasible regardless of the first stage decisions and the outcome of the random variables (Kall and Wallace 1994). If this is true under the condition that the first stage decisions are feasible, we have *relatively complete recourse*. Even in the weaker of these cases, feasibility of the recourse can be disregarded when solving the first stage problem. For our model, we introduce recourse completeness in the following sense:

Definition 1. *Problem (1)-(3) is said to have* (k, ℓ)-complete recourse *when* $\forall S \subseteq V$ *the condition*

$$\sum_{v \in S} ED_v \leq kq \tag{4}$$

implies that there exists a permutation $\pi_S = (v_1, \dots, v_{|S|})$ *of* S*, an integer* $r \in \{1, \dots, \ell\}$*, and integers* $0 = i_0 < i_1 < \cdots < i_r = |S|$ *such that the following inequalities hold:*

$$\sum_{j=i_{p-1}+1}^{i_p} \overline{d}_{v_j} \leq q \qquad \forall p = 1, \dots, r \tag{5}$$

$$g(\pi_S, i_1, \dots, i_{r-1}) = \sum_{p=1}^{r} \left(c_{0 v_{i_{p-1}+1}} + c_{0 v_{i_p}} + \sum_{j=i_{p-1}+1}^{i_p - 1} c_{v_j v_{j+1}} \right) \leq t. \tag{6}$$

Note that $g(\pi_S, i_1, \dots, i_{r-1})$ is the total travel cost of the r routes $(v_0, v_1, \dots, v_{i_1}, v_0)$, $(v_0, v_{i_1+1}, \dots, v_{i_2}, v_0)$, $\dots$, $(v_0, v_{i_{r-1}+1}, \dots, v_{i_r}, v_0)$, where $v_0 = 0$ signifies the depot.

In other words, (k, ℓ)-completeness implies that if the customers are partitioned such that the total expected demand in each district is no greater than the capacity of k vehicles, feasibility is guaranteed. Furthermore, regardless of the outcome of the demand, inequalities (5)-(6) imply that the routing problem in each district can be solved by no more than ℓ non-empty routes. Given a partition $\{S_1, \dots, S_m\}$ of V, where all of $S_1, \dots, S_m$ satisfy (4), the completeness property means that only travel plans with this few routes need to be considered when computing the expected recourse cost.

3 Approximation of the expected cost

Since computing the expected cost of a given district S requires the solution of a number of routing problems, this is in its own right a resource demanding operation. By relying on an approximate solution to the subproblems, provided by a 2-opt procedure, the heuristic can be guided by upper bounds rather than the expected cost itself. In Section 3.1 we show how to compute such bounds. The 2-opt procedure (Lin 1965) is a local search improvement method which replaces two arcs $(i, i+1)$ and $(j, j+1)$ by two other arcs (i, j) and $(i+1, j+1)$, where $i+1$ and $j+1$ are the successors of i and j respectively in a given orientation of the route. In order to maintain the feasible orientation of the route, the path $(i+1, \dots, j)$ needs to be reversed. This procedure is applied to the route until it is not possible to obtain a better permutation.

3.1 Approximating the expected recourse cost

Consider a district $S \subseteq V$ satisfying (4), and let $\pi_S = (v_1, \dots, v_{|S|})$ be a permutation of its customers given by a 2-opt procedure. For a given $d \in \Omega$, we define

$$\begin{aligned}
g^*(\pi_S, d) &= \min_{r, i_0, \dots, i_r} g(\pi_S, i_1, \dots, i_{r-1}) && (7)\\
\text{s.t.} \quad & 0 = i_0 < i_1 < \cdots < i_r = |S| && (8)\\
& r \in \{1, \dots, \ell\} && (9)\\
& \sum_{j=i_{p-1}+1}^{i_p} d_{v_j} \leq q \qquad \forall p = 1, \dots, r && (10)
\end{aligned}$$

which is computable in $\mathcal{O}\left(|S|^{\ell+1}\right)$ time.

Since the set of routes $\left\{\left(0, v_{i_{p-1}+1}, \dots, v_{i_p}, 0\right)\right\}_{p=1}^{r}$ constitutes a feasible solution to $VRP(S, d)$, it follows that $f(S, d) \leq g^*(\pi_S, d)$. By (k, ℓ)-completeness, we also have $f(S,d) \leq t$, and hence the upper bound

$$E_D f(S,D) \leq B(\pi_S) = \sum_{d\in\Omega} P(D=d) \min\{t, g^*(\pi_S, d)\}.$$

Similarly, for $u \in S$ and $v \in V \setminus S$, we define $\pi_S - u$ and $\pi_S + v$ to be the permutations of $S\setminus\{u\}$ and $S\cup\{v\}$, respectively, where the order in π_S is preserved, and where $\pi_S + v$ is formed by inserting v in a position in π_S such that $B(\pi_S + v)$ is minimized.

3.2 The case of (1,2)-complete recourse

Assuming that the recourse is (1,2)-complete, the upper bound on $E_D f(S_j, D)$ can be written as $B(\pi_S) = P(D_S \leq q) g(\pi_S) + \sum_{i=1}^{|S|-1} P(\mathcal{E}_i) \min\{t, g(\pi_S, i)\}$, where $\mathcal{E}_i$ denotes the event that v_i is the *feasible return point* on π_S minimizing the saving $\delta_i = c_{0v_i} + c_{0v_{i+1}} - c_{v_i v_{i+1}}$. For simplicity, we assume distinct savings on route π_S (a consistent ordering can be introduced to break ties). A customer $v_i \in S$ is a feasible return point for the demand d if $\sum_{j=1}^{i} d_{v_j} \leq q$ and $\sum_{j=i+1}^{|S|} d_{v_j} \leq q$.

In order to compute $B(\pi_S)$, we derive explicit formulae for $P(\mathcal{E}_i)$ for all $i = 1, \ldots, |S|-1$ (note that $\mathcal{E}_0 = \mathcal{E}_{|S|}$, and that this event is identical to $D_S \leq q$). For $i_0 \in \{1, \ldots, |S|\}$, define the integers $\alpha, \beta \in \{1, \ldots, |S|\}$ and $0 = i_{-\alpha} < i_{-\alpha+1} < \cdots < i_{-1} < i_0 < i_1 < \cdots < i_\beta = |S|$ such that $\{i_{-\alpha}, \ldots, i_{-1}, i_1, \ldots, i_\beta\} = \{i = 0, \ldots, |S| : \delta_i < \delta_{i_0}\}$. Let $\mathcal{E}_i^-$ and $\mathcal{E}_i^+$ denote the events $\sum_{j=1}^{i} D_{v_j} \leq q$ and $\sum_{j=i+1}^{|S|} D_{v_j} \leq q$, respectively.

Proposition 1. $P(\mathcal{E}_{i_0}) = P(\mathcal{E}_{i_0}^-)P(\mathcal{E}_{i_0}^+) - P(\mathcal{E}_{i_{-1}}^+ \wedge \mathcal{E}_{i_0}^-) - P(\mathcal{E}_{i_1}^- \wedge \mathcal{E}_{i_0}^+) + P(\mathcal{E}_{i_{-1}}^+ \wedge \mathcal{E}_{i_1}^-)$

Proof. Since $\mathcal{E}_{i_j}^+ \to \mathcal{E}_{i_{j+1}}^+$ and $\mathcal{E}_{i_{j+1}}^- \to \mathcal{E}_{i_j}^-$ $(j = -\alpha, \ldots, \beta-1)$, we have

$$P(\mathcal{E}_{i_0}) = P\left(\mathcal{E}_{i_0}^- \wedge \mathcal{E}_{i_0}^+ \wedge \bigwedge_{\substack{j=-\alpha \\ j\neq 0}}^{\beta} \left(\neg\mathcal{E}_{i_j}^- \vee \neg\mathcal{E}_{i_j}^+\right)\right)$$

$$= P\left(\mathcal{E}_{i_0}^- \wedge \mathcal{E}_{i_0}^+ \wedge \bigwedge_{j=1}^{\alpha} \neg\mathcal{E}_{i_{-j}}^+ \wedge \bigwedge_{j=1}^{\beta} \neg\mathcal{E}_{i_j}^-\right)$$

$$= P\left(\mathcal{E}_{i_0}^- \wedge \mathcal{E}_{i_0}^+ \wedge \neg\mathcal{E}_{i_{-1}}^+ \wedge \neg\mathcal{E}_{i_1}^-\right)$$

$$= P\left(\mathcal{E}_{i_0}^- \wedge \mathcal{E}_{i_0}^+ \wedge \neg\mathcal{E}_{i_1}^-\right) - P\left(\mathcal{E}_{i_0}^- \wedge \mathcal{E}_{i_{-1}}^+ \wedge \neg\mathcal{E}_{i_1}^-\right)$$

$$= P\left(\mathcal{E}_{i_0}^-\right) P\left(\mathcal{E}_{i_0}^+\right) - P\left(\mathcal{E}_{i_1}^- \wedge \mathcal{E}_{i_0}^+\right) - P\left(\mathcal{E}_{i_0}^- \wedge \mathcal{E}_{i_{-1}}^+\right) + P\left(\mathcal{E}_{i_{-1}}^+ \wedge \mathcal{E}_{i_1}^-\right)$$

□

Note that if $\delta_i = \min\{\delta_j : j = 1, \ldots, |S|-1\}$, that is v_i is the best return point in π_S apart from $v_{|S|}$, then Proposition 1 states that $P(\mathcal{E}_i) = P(\mathcal{E}_i^-) P(\mathcal{E}_i^+) - P(D_S \leq q)$.

4 Tabu search heuristic

Our heuristic is a three-phase process. An initial solution is created, this solution is then improved by tabu search, and finally the best solution found is further improved by removing small size districts. Tabu search is a memory-based search strategy, originally proposed by Glover (1986), that guides the local search process beyond a local optimum. One way of achieving this is to keep track of recent moves or solutions made in the past. A tabu list records recently made moves or visited solutions. Whenever the algorithm attempts to move a solution forbidden by the tabu list, the move is banned. This rule forces other solutions to be explored. However, this feature is not strict, as it can be overridden when some aspiration criterion is satisfied. A common criterion is that the target function value be the best ever seen. If this is the case, it is obvious that this solution has never been encountered before.

4.1 Notation of the heuristic

An important feature of our heuristic is the possibility to explore infeasible regions of the search space. This idea was first proposed by Gendreau et al. (1994) in the context of the VRP. Each solution x consists of m districts such that every customer belongs to exactly one district but may violate the capacity (4) or the cost (3) constraints. For a solution x, let $\phi(x)$ denote the sum of the approximate expected cost of the districts, and let $q(x)$ and $\rho(x)$ denote the total violation of capacity and cost constraints. The function $\phi(x)$ is defined as $\sum_{j=1}^{m}(B(\pi_{S_j}) + h\chi(S_j))$. The total violation of capacity constraints is computed on a district basis with respect to the expected demand and q, i.e. $q(x) = \sum_{j=1}^{m} q(S_j)$, where $q(S_j) = \max\{kq - \sum_{v \in S_j} ED_v, 0\}$. For the violation of (3) we have $\max\{t - f(S_j, \overline{d}_{S_j}), 0\} \leq \max\{t - B(\pi_{S_j}), 0\} = \rho(S_j)$. We define the approximate total cost constraint violation as $\rho(x) = \sum_{j=1}^{m} \rho(S_j)$. Each solution is evaluated by a cost function defined as $z(x) = \phi(x) + \mu q(x) + \nu\rho(x)$, where μ and ν are self-adjusting positive parameters.

4.2 Initial solution

We have devised two procedures for the generation of an initial solution. In the first procedure, called I1, the first district is initiated by the farthest customer from the depot, and the district is expanded by including the closest neighbor to the district until the total expected demand reaches the capacity level, the approximated expected cost exceeds a predetermined bound or the candidate neighbor is invalid. Closeness between a customer and a district is defined as the distance from the customer to its closest neighbor in the district. A neighbor is invalid if there exists other neighbors that are closer to the district in question and already belonging to some other districts. If this is the case, then it is most likely that by including this particular neighbor we will create a district with high expected travel cost. The construction of the remaining districts is identical to the procedure described above but with the exception that the seed is chosen to be the closest neighbor of the current district. This method is likely to generate several districts containing one or two customers, but these will normally be eliminated throughout the improvement phase.

The second procedure, called I2, is based on the sweep technique, applying to instances defined in the Euclidean plane. The first district is initiated by the farthest customer from the depot. The remaining customers are then sorted in increasing order of the angle between the straight lines intersecting at the depot, and paasing through the customer and the seed, respectively. The district is expanded by including the customer with the smallest angle until any of the termination criteria of I1 is satisfied. The construction of the remaining districts is identical to the procedure described above, but with the exception that the seed customer is chosen to be the closest neighbor of the current district.

4.3 Neighborhoods

The neighborhood of solution x is defined as all solutions that can be reached by expanding district K through the transfer of the closest neighbor i of district L. Such a move is called a relocation move and is denoted by (K, i, L). The neighborhood of the solution x is denoted $N(x)$.

4.4 Recency-based memory and tabu tenure

To avoid cycling whenever a move (K, i, L) is performed, any move that transfers i back into district L is declared tabu for θ iterations, where θ is randomly chosen from an interval $[\theta_{\min}, \theta_{\max}]$. A forbidden move will still be chosen if it satisfies the aspiration criterion.

4.5 Frequency-based memory

To diversify the search and to guide it to explore a wider part of the solution space, frequently made moves are penalized. The frequency of move (K, i, L) is the number of times neighbor i has been moved to district K. For any solution $\overline{x} \in N(x)$, whenever $z(\overline{x}) \geq z(x)$, a penalty is added to $z(\overline{x})$. Only moves that lead to non-improved solutions are penalized since it makes no sense to penalize improving moves. The penalty $\psi(\overline{x})$ is defined as $\lambda\phi(\overline{x})\sqrt{m}\vartheta_{iK}$. The factor ϑ_{iK} denotes the number of times neighbor i has been moved to district K during the search so far. The factor m denotes the number of non-empty districts of solution $\overline{x}$, and λ is an user controlled parameter. This idea was first proposed by Taillard (1993) in the context of the VRP.

4.6 Search process

The tabu search heuristic starts from the initial solution described in Section 4.2. This solution always remains feasible with respect to the constraints. At each iteration we select the least cost non-tabu solution $\overline{x}$ from $N(x)$, and the values of the parameters μ and ν are adjusted as follows: if there is no violation of the load constraints, the value of μ is divided by $1 + \tau$, otherwise it is multiplied by $1 + \tau$, where τ is a positive parameter. A similar rule also applies to ν. This way of controlling the parameters was proposed by Cordeau et al. (1997) and enables a thorough exploration of the search space by generating a good mix of feasible and infeasible solutions.

Besides penalizing frequently performed moves to obtain diversity, we introduce two procedures that will also guide the search to that direction. The first consists of initiating a new district containing only one customer. The selection process starts with the most overloaded district and determines the customer to be relocated as the one having the smallest number of closest neighbors in its current district. When no district violates the capacity constraint, a district is chosen randomly. This procedure is applied every ω iterations. The second procedure eliminates districts of small size, where size is defined by the fraction of the total expected demand over the capacity. Any district for which this fraction is less than or equal to a given parameter φ is considered a small district. Customers of these districts are relocated to their nearest neighbor districts without taking feasibility constraints into account. This procedure is applied every σ iterations.

A parameter τ controls the restarting of the heuristic. Whenever several consecutive infeasible solutions have been created and the parameters μ and ν become very large, the heuristic replaces the current solution $\overline{x}$ with the best known solution x^*, and the parameters μ and ν are set to their initial values. However, the list of tabu moves remains unchanged. This was implemented to prevent $z(\overline{x})$ from becoming too large, in other words, whenever $z(\overline{x})$ exceeds a given parameter Z, a restart of the heuristic is performed.

This process is repeated until γ consecutive iterations are performed without any improvement to the best known solution x^*. The best solution x^* is then post-optimized by eliminating small size districts.

4.7 Post-optimization

The post-optimization phase is similar to the one used to eliminate small size districts described in Section 4.6. The difference is that we do not allow the final solution to be infeasible. We again start by identifying the smallest district, and one customer at a time is relocated to its nearest neighbor district. If including this customer yields an infeasible district, then one of its customers is relocated to its closest district. The selection of the customer to be relocated is based on the approximate expected cost of the district to which it would be transferred. The process is repeated until a feasible district is obtained. In some cases this is not possible, if after ξ consecutive relocations of the same customer and it is not possible to reach a feasible district, the post-optimization phase fails.

4.8 Summary of the tabu search heuristic

Set $\mu = 1$ and $\nu = 1$. If x is feasible, set $x^* = x$ and $\phi(x^*) = \phi(x)$; otherwise, set $\phi(x^*) = \infty$.
repeat
  Select a solution $\overline{x} \in N(x)$ that minimizes $z(\overline{x}) + \psi(\overline{x})$ and is non-tabu or satisfies the aspiration criterion.
  If $\overline{x}$ is feasible and $\phi(\overline{x}) < \phi(x^*)$, set $x^* = \overline{x}$ and $\phi(x^*) = \phi(\overline{x})$.
  Set the reverse move tabu for θ iterations, where $\theta \in [\theta_{\min}, \theta_{\max}]$.
  Compute $q(\overline{x})$ and $\rho(\overline{x})$ and update μ and ν.
  Apply the diversification procedures.
  Set $x = \overline{x}$.
until γ consecutive iterations are performed without any improvement to the best known solution
Apply a post-optimization procedure to x^*.

5 Descent heuristic

For comparison purposes we have also designed a descent search heuristic that does not allow a worsening of the objective function or infeasible solutions. Like the tabu search heuristic the descent heuristic also contains three phases. An initial solution is created, it is then improved by local search, and finally the best solution found is post-optimized.

6 Computational experiments

Because no benchmark instances exist for this problem, we have used the R1 instances designed by Solomon (1987) designed for the *Vehicle Routing Problem with Time Windows*. However, only the depot and customer locations are used. A lower

Table 1: Parameter settings

$\theta_{\min} = 2$	$\varphi = 0.4$
$\theta_{\max} = 5$	$\sigma = 9$
$\lambda = 0.5$	$\gamma = 100$
$\tau = 1$	$\xi = 2$
$\omega = 3$	$Z = 10000000$

bound a, and an upper bound b, are first given as fractions of the vehicle capacity for the demands. For each customer a lower bound $\underline{d}_v$, is chosen from the interval $[aq, bq]$, and an upper bound $\overline{d}_v$, is chosen in the interval $[\underline{d}_v, bq]$. The results of the following subsections were obtained with the parameter settings presented in Table 1. Note that these settings were not tuned to particular problem instances. The heuristic was coded in C++ and all experiments were performed on a Sun Ultra 10.

The main purpose of this section is to compare the two methods described in Sections 4 and 5. Because the problem is new, no known optimal solutions are available for comparison purposes. The heuristics start with procedure I1 or I2 for the generation of feasible initial solutions, which have an impact on the final solutions. We also denote the tabu search procedure by TS and the descent search procedure by DS. These are therefore four different combinations of the methods: (i) H1: I1+TS; (ii) H2: I2+TS; (iii) H3: I1+DS, and (iv) H4: I2+DS. In Section 6.1, the heuristics are tested for various problem sizes. They are further tested in Section 6.2 for different demand variations.

6.1 Test of heuristics over problem sizes and expected cost limits

The four heuristics were tested for instances ranging in size from 25 to 100 customers for R1 instances and cost limit t ranging from 100 to 300. Table 2 summarizes the results (CPU-time and in how many cases the algorithm was the best one) of 10 runs of every combination of n and t for each of the algorithms. For all problem instances, the vehicle capacity q is set equal to 100, a is set equal to 0.05, and b is set equal to 0.1. From the table we observe that H1 and H2 produce the best solutions, the reason being that the tabu search method conducts a broader search, while the descent heuristic is far more restricted. Heuristic H2 produces the best overall results among the four methods. The difference between H1 and H2 is the procedure used for generating an initial solution: I1 generates solutions with many districts, some of them rather small, while I2 does not have this drawback. For this reason, H1 requires a much longer computing time. Because DS terminates when no improvement can be made, its running time is shorter.

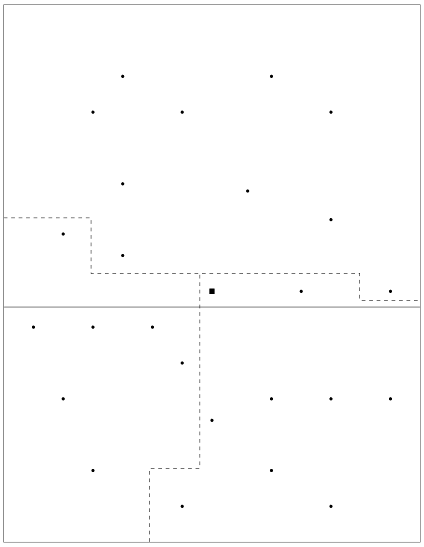
Figure 1 depicts the districts generated by H2 for $n = 25, 50, 75, 100$, $t = 200$, $a = 0.05$ and $b = 0.1$. These figures show the boundaries made by I2 and TS. The configuration of delivery districts in our solutions is consistent with the pattern observed in Daganzo and Newell (1986).

6.2 Comparison of heuristics over demand variations

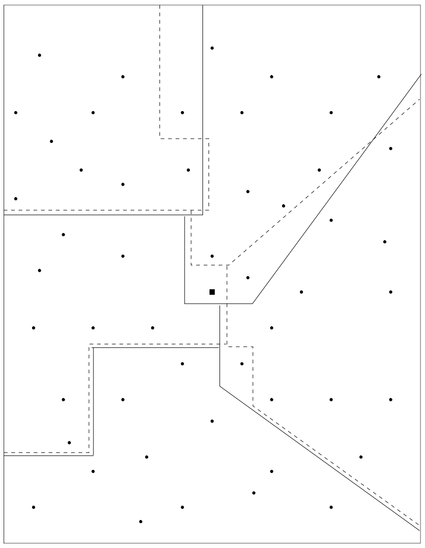
The heuristics were tested for different demand variations as shown in Table 3. For all problem instances we used $n = 100$, $t = 200$, and $q = 100$. Each heuristic was run 10 times for each pair of a and b. We observe that H2 again produced the best results, and H1 required much computing time. As the expected demands increase, the computation time decreases. This is due to the restriction of the number of

Table 2: Comparison of the heuristics for different problem sizes

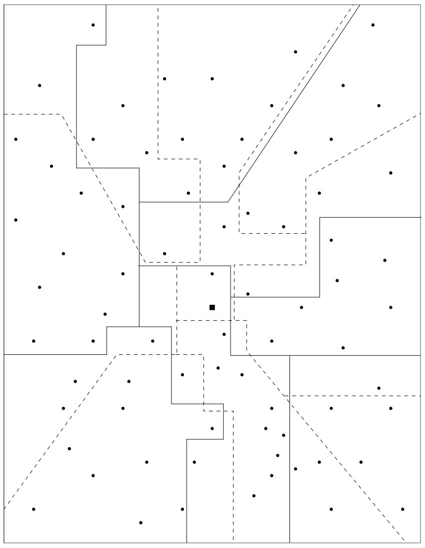
		H1		H2		H3		H4	
n	t	Number of best	Average CPU seconds	Number of best	Average CPU seconds	Number of best	Average CPU seconds	Number of best	Average CPU seconds
25	100	2	60.45	8	49.46	1	0.5	0	0.28
	200	7	110.5	3	110.65	0	6.63	0	2.45
	300	4	123.6	1	122.45	1	5.54	4	10.52
50	100	5	156.1	0	191.8	7	4.08	0	0.81
	200	4	357.39	5	225.94	1	14.51	0	47.05
	300	2	342.47	3	331.61	0	15.41	5	19.7
75	100	5	616.13	4	294.68	1	20.24	0	3.24
	200	5	889.82	4	521.03	0	52.65	1	12.46
	300	0	1058.82	5	574.74	0	70.34	5	24.85
100	100	2	1198.66	5	352.64	2	46.68	1	7.68
	200	0	923.22	7	645.36	0	67.16	3	31.84
	300	0	1226.33	5	742.64	0	79.68	5	26.43



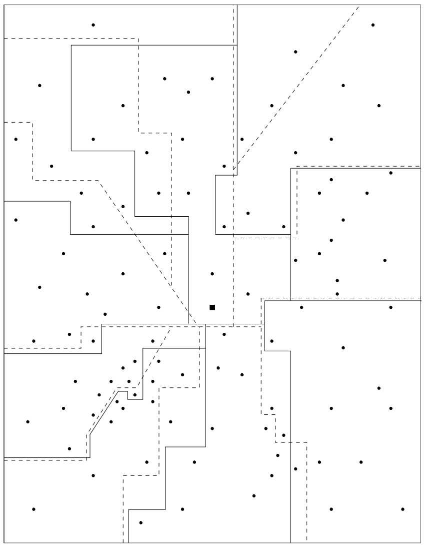
(a) 25 customers



(b) 50 customers



(c) 75 customers



(d) 100 customers

Figure 1: Districts obtained with 25, 50, 75 and 100 customers. The dotted lines correspond to the initial solution and the full lines correspond to the final solution.

possibilities for moving the customers between the districts when demands are large. Again, DS requires much less computation time than TS.

7 Conclusion

We have introduced, formulated and solved the problem of designing districts for vehicle routing problems with stochastic demands. In the unconstrained case, a district serving all customers will be an optimal solution. However, in practical contexts, districts are constrained by an upper bound on the total travel cost. Two procedures for generating initial solutions and two improvement methods were designed. One is based on tabu search and the other is a descent heuristic. Both are based on simple relocation moves, and tabu search allows intermediate infeasible solutions. We have conducted experiments on instances with small and large districts by varying the upper bound on the travel cost and by also varying a and b. Because the problem is new, comparisons can only be made between the heuristics we have developed. The combination of a sweep based initial solution procedure coupled with tabu search yields the best results.

Acknowledgements

This work was supported by the Research Council of Norway under grant 127533/432 and by the Canadian Natural Sciences and Engineering Research Council under grant OGP0039682. This support is gratefully acknowledged.

References

Blais, M., Lapierre, S. D. and Laporte, G. (2003), 'Solving a Home Care Districting Problem in an Urban Setting', *Journal of the Operational Research Society* **54**(11), 1141–1147.

Bozkaya, B., Erkut, E. and Laporte, G. (2003), 'A Tabu Search Heuristic and Adaptive Memory Procedure for Political Districting', *European Journal of Operational Research* **144**(1), 12–26.

Cordeau, J.-F., Gendreau, M. and Laporte, G. (1997), 'A Tabu Search Heuristic for Periodic and Multi-Depot Vehicle Routing Problems', *Networks* **30**(2), 105–119.

Daganzo, C. F. and Newell, G. F. (1986), 'Configuration of Physical Distribution Networks', *Networks* **16**(2), 113–132.

D'Amico, S. J., Wang, S.-J., Batta, R. and Rump, C. M. (2002), 'A Simulated Annealing Approach to Police District Design', *Computers & Operations Research* **29**(6), 667–684.

Drexl, A. and Haase, K. (1999), 'Fast Approximation Methods for Sales Force Deployment', *Management Science* **45**(10), 1307–1323.

Ferland, J. A. and Guénette, G. (1990), 'Decision Support System for a School Districting Problem', *Operations Research* **38**(1), 15–21.

Fleischmann, B. and Paraschis, J. (1988), 'Solving a Large Scale Districting Problem: A Case Report', *Computers & Operations Research* **15**(6), 521–533.

Table 3: Comparison of the heuristics for different demand variations

		H1		H2		H3		H4	
a	b	Number of best	Average CPU seconds	Number of best	Average CPU seconds	Number of best	Average CPU seconds	Number of best	Average CPU seconds
0.05	0.1	0	923.22	7	645.36	0	67.16	3	31.84
0.1	0.15	4	271.33	5	199.41	0	14.37	1	3.75
0.2	0.25	8	67.34	2	21.05	0	1.81	2	0.54
0.3	0.35	1	16.18	5	10.42	0	0.59	4	0.33
0.4	0.45	0	9.54	10	7.09	0	0.31	0	0.23

Gendreau, M., Hertz, A. and Laporte, G. (1994), 'A Tabu Search Heuristic for the Vehicle Routing Problem', *Management Science* **40**(10), 1276–1290.

Gendreau, M., Laporte, G. and Potvin, J.-Y. (2002), Metaheuristics for the Capacitated VRP, *in* P. Toth and D. Vigo, eds, 'The Vehicle Routing Problem', SIAM Society for Industrial and Applied Mathematics.

Gendreau, M., Laporte, G. and Séguin, R. (1996), 'Stochastic Vehicle Routing', *European Journal of Operational Research* **88**(1), 3–12.

Glover, F. (1986), 'Future Paths for Integer Programming and Links to Artificial Intelligence', *Computers & Operations Research* **13**(5), 533–549.

Kall, P. and Wallace, S. (1994), *Stochastic Programming*, Wiley, Chichester.

Laporte, G. and Semet, F. (2002), Classical Heuristics for the Capacitated VRP, *in* P. Toth and D. Vigo, eds, 'The Vehicle Routing Problem', SIAM Society for Industrial and Applied Mathematics.

Lin, S. (1965), 'Computer Solutions of the Traveling Salesman Problem', *Bell System Technical Journal* **44**, 2245–2269.

Pezzella, F., Bonanno, R. and Nicoletti, B. (1981), 'A System Approach to the Optimal Health Care Districting', *European Journal of Operational Research* **8**(2), 139–146.

Skiera, B. and Albers, S. (1998), 'Costa: Contribution Optimizing Sales Territory Alignment', *Marketing Science* **17**(3), 196–213.

Solomon, M. M. (1987), 'Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints', *Operations Research* **35**(2), 254–265.

Taillard, É. D. (1993), 'Parallel Iterative Search Methods for Vehicle Routing Problems', *Networks* **23**(8), 661–673.