Stochastic Vehicle Routing: an Application of Stochastic Optimization Models

Michel Gendreau
CIRRELT and MAGI
École Polytechnique de Montréal

Summer School on Dynamic Discrete Choice Models: Econometric Models and Operations Research Methods
Montreal – June 10-12, 2015

Outline

1. Introduction
2. Basic Concepts in Stochastic Optimization
3. Modeling Paradigms
4. Problems with Stochastic Demands
5. Problems with Stochastic Customers
6. Problems with Stochastic Service or Travel Times
7. Conclusion and perspectives

Acknowledgements

- Walter Rei (CIRRELT and UQÀM)
- Ola Jabali and Gilbert Laporte (CIRRELT and HEC Montréal)
- Duygu Tas, Nico Dellaert, Tom van Woensel and Ton de Kok (School of Industrial Engineering, Eindhoven University of Technology)
- Charles Gauvin andLouis-Martin Rousseau (CIRRELT and École Polytechnique)
- Guy Desaulniers (GERAD and École Polytechnique)
- Fausto Errico (CIRRELT and ETS)

Introduction

Vehicle Routing Problems

- Introduced by Dantzig and Ramser in 1959
- One of the most studied problem in the area of logistics
- The basic problem involves delivering given quantities of some product to a given set of customers using a fleet of vehicles with limited capacities.
- The objective is to determine a set of minimum-cost routes to satisfy customer demands.

Vehicle Routing Problems

Many variants involving different constraints or parameters:

- Introduction of travel and service times with route duration or time window constraints
- Multiple depots
- Multiple types of vehicles
- …

What is Stochastic Vehicle Routing?

Basically, any vehicle routing problem in which one or several of the parameters are not deterministic:

- Demands
- Travel or service times
- Presence of customers
- …

Main classes of stochastic VRPs

- VRP with stochastic demands (VRPSD)
  - A probability distribution is specified for the demand of each customer.
  - One usually assumes that demands are independent (this may not always be very realistic...).
- VRP with stochastic customers (VRPSC)
  - Each customer has a given probability of requiring a visit.
- VRP with stochastic travel times (VRPSTT)
  - The travel times required to move between vertices, as well as sometimes service times, are random variables.

Basic Concepts in Stochastic Optimization

Dealing with uncertainty in optimization

- Very early in the development of operations research, some top contributors realized that :
  - In many problems there is very significant uncertainty in key parameters;
  - This uncertainty must be dealt with explicitly.
- This led to the development of :
  - Stochastic programming with recourse (1955)
  - Dynamic programming (1958)
  - Chance-constrained programming (1959)
  - Robust optimization (more recently)

Information and decision-making

In any stochastic optimization problem, a key issue is:

- How do the revelation of information on the uncertain parameters and decision-making (optimization) interact?
  - When do the values taken by the uncertain parameters become known?
  - What changes can I (must I) make in my plans on the basis of new information that I obtain?

Stochastic programming with recourse

- Proposed separately by Dantzig and by Beale in 1955.
- The key idea is to divide problems in different stages, between which information is revealed.
- The simplest case is with only two stages. The second stage deals with **recourse actions**, which are undertaken to adapt plans to the realization of uncertainty.
- Basic reference:

  J.R. Birge and F. Louveaux, Introduction to Stochastic Programming, 2nd edition, Springer, 2011.

Dynamic programming

- Proposed by Bellman in 1958.
- A method developed to tackle effectively sequential decision problems.
- The solution method relies on a time decomposition of the problem according to stages. It exploits the so-called *Principle of Optimality*.
- Good for problems with limited number of possible **states** and **actions**.
- Basic reference:

  D.P. Bertsekas, Dynamic Programming and Optimal Control, 3rd edition, Athena Scientific, 2005.

Chance-constrained programming

- Proposed by Charnes and Cooper in 1959.
- The key idea is to allow some constraints to be satisfied only with some probability.

E.g., in VRP with stochastic demands,

Pr{total demand assigned to route $r \leq capacity$ } $\geq 1-\alpha$

Robust optimization

- Here, uncertainty is represented by the fact that the uncertain parameter vector must belong to a given polyhedral set (without any probability defined)
  - E.g., in VRP with stochastic demands,

    having set upper and lower bounds for each demand, together with an upper bound on total demand.
- Robust optimization looks in a minimax fashion for the solution that provides the best “worst case”.

Modelling paradigms

Real-time optimization

Also called re-optimization

- Based on the implicit assumption that information is revealed over time as the vehicles perform their assigned routes.
- Relies on Dynamic programming and related approaches (Secomandi et al.)
- Routes are created piece by piece on the basis on the information currently available.
- Not always practical (e.g., recurrent situations)

A priori optimization

- A solution must be determined beforehand;
  this solution is "confronted" to the realization of the stochastic parameters in a second step.
- Approaches:
  - Chance-constrained programming
  - (Two-stage) stochastic programming with recourse
  - Robust optimization
  - ["Ad hoc" approaches]

Chance-constrained programming

- Probabilistic constraints can sometimes be transformed into deterministic ones (e.g., in in VRP with stochastic demands, when one imposes that

  Pr{total demand assigned to route *r* ≤ *cap.* } ≥ 1-α,

  if customer demands are independent and Poisson).

- This model completely ignores what happens when things do not "turn out correctly".

Robust optimization

- Not used very much in stochastic VRP up to now, but papers have been appearing in the last few years for node and arc routing problems.
- Model may be overly pessimistic.

Stochastic programming with recourse

- **Recourse** is a key concept in a priori optimization
  - What must be done to “adjust” the a priori solution to the values observed for the stochastic parameters!
  - Another key issue is deciding when information on the uncertain parameters is provided to decision-makers.
- Solution methods:
  - Integer L-shaped (Laporte and Louveaux)
  - Column generation (Branch & Price)
  - Heuristics (including metaheuristics)
- Probably closer to actual industrial practices, if recourse actions are correctly defined!

VRP with stochastic demands

VRP with stochastic demands (VRPSD)

- A probability distribution is specified for the demand of each customer.
- One usually assumes that demands are independent
(this may not always be very realistic...).
- Probably, the most extensively studied SVRP:
  - Under the reoptimization approach (Secomandi)
  - Under the a priori approach (several authors) using both the chance-constrained and the recourse models.

VRP with stochastic demands

- Classical recourse strategy:
  - Return to depot to restore vehicle capacity
  - Does not always seem very appropriate or “intelligent”

- Other recourse strategies are possible, however, and often closer to actual industrial practices.
  - Fixed threshold policies
  - Variable threshold policies
  - Preventive restocking (Yang, Mathur, Ballou, 2000)
  - Pairing routes (Erera et al., 2009)

VRP with stochastic demands

- Approximate solutions can be obtained fairly easily using metaheuristics
(e.g., Tabu Search, as in Gendreau et al., 1996).

- Computing effectively the value of the recourse function still remains a challenge.

A branch-and-cut approach (direct formulation)

The following material is taken from

- O. Jabali, W. Rei, M. Gendreau, G. Laporte (2014). New Valid Inequalities for the Multi-Vehicle Routing Problem with Stochastic Demands. *Discrete Applied Mathematics,* 177, 121-136.

Model

Input

$G(V, E)$ = undirected graph, $V = \{v_1, \ldots, v_n\}$ and $E = \{(v_i, v_j) \colon v_i, v_j \in V, i < j\}$
ξ_i = demand of client i, where $i = 2, \ldots, n$
$C = (c_{ij})$ travel cost matrix
D = capacity of each vehicle

Decision variables

$$
\begin{aligned}
x_{ij} &= \{0, 1\} && \text{for } i, j > 1 \\
x_{1j} &= \{0, 1, 2\} && \text{for } \quad j > 1
\end{aligned}
$$

The considered case

Client demands are independent
$\xi_j \sim N(\mu_j, \sigma_j)$ and $\xi_j \in (0, D)$, $j = 2, \ldots, n$
Recourse rules $\Rightarrow$ return to depot only when failure occurs
For a given route $(v_{r_1} = v_1, v_{r_2}, \ldots, v_{r_{t+1}} = v_1)$

$$
\begin{aligned}
\mathcal{Q}^{1,r} &= 2 \sum_{i=2}^{t} \sum_{l=1}^{i-1} P\left(\sum_{s=2}^{i-1} \xi_{r_s} \leq lD < \sum_{s=2}^{i} \xi_{r_s} \right) c_{1 r_i} \\
\mathcal{Q}(x) &= \sum_{i=1}^{m} \min\{\mathcal{Q}^{k,1}, \mathcal{Q}^{k,2}\},
\end{aligned}
$$

Model

$$\text{(VRPSD)} \qquad \text{Minimize} \sum_{i<j} c_{ij} x_{ij} + \mathcal{Q}(x)$$

subject to

$$\begin{aligned}
\sum_{j=2}^{n} x_{1j} &= 2m, && \\
\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} &= 2, && (k = 2, \dots, n), \\
\sum_{v_i, v_j \in S} x_{ij} \leq |S| - \left\lceil \sum_{v_i \in S} \mathbf{E}(\xi_i)/D \right\rceil &, && S \subset V \setminus \{v_1\}, 2 \leq |S| \leq n-2 \\
0 \leq x_{ij} &\leq 1 && 1 \leq i < j < n), \\
0 \leq x_{0j} &\leq 2 && (j = 2, \dots, n), \\
x &= (x_{ij}) && \text{integer.}
\end{aligned}$$

A branch-and-cut approach

- Computational results on problems with independent truncated Normal demands
  - 30 instances for each combination of *m* and *n*
  - 10 hours of CPU time for each run.

n	m	Solved	Runtime (min)	Gap
60	2	19	227	0.2%
70	2	14	372	0.4%
80	2	14	366	0.5%
50	3	9	458	0.8%
60	3	8	473	1.1%
70	3	11	419	1.6%
40	4	5	519	2.5%
50	4	4	529	3.9%
60	4	3	552	2.9%

A branch-and-price approach (set covering formulation)

The following material is taken from

- C. Gauvin, G. Desaulniers, M. Gendreau (2014). A Branch-Cut-and-Price Algorithm for the Vehicle Routing Problem with Stochastic Demands, *Computers & Operations Research*, 50, 141-153.

Litterature

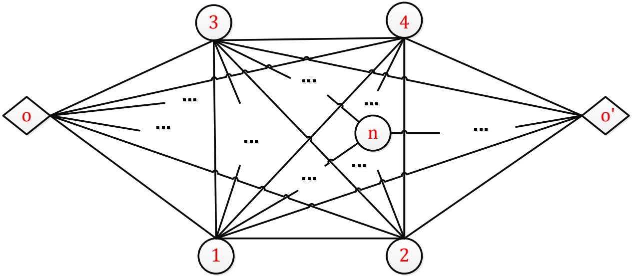
- **Heuristics**
  - Tillman (1965) - Savings-based heuristics for multi-depot and Poisson demands
  - Golden, Stewart (1983)
  - Gendreau, Séguin (1996) - Tabou search
- **Integer L-shaped method for problems with simple recourse**
  - Laporte, Louveaux (1993)
  - Gendreau, Laporte, Séguin (1995)
  - Hjorring, Holt (1999)
  - Laporte, Louveaux, Van Hamme (2001)
- **Column Generation**
  - Christiansen, Lysgaard (2007) Branch-and-price

Our Contribution

Develop a competitive branch-cut-and-price algorithm for the VRPSD based on the work of Christiansen et Lysgaard (2007).

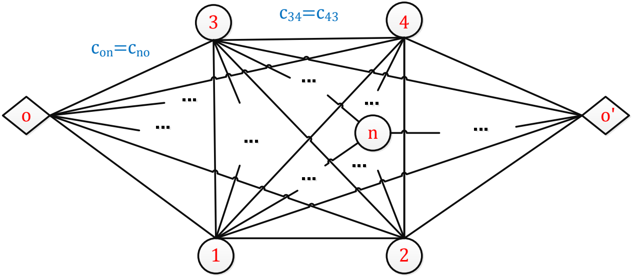
Notation

- $\mathcal{G} = (\mathcal{N}' = \mathcal{N} \cup \{o, o'\}, \mathcal{A})$: undirected graph
- $\mathcal{N} = \{1, 2, ..., n\}$: set of clients
- $\mathcal{A} = \{(i,j) | i,j \in \mathcal{N}; i \neq j\} \cup \{(o,j) | j \in V\} \cup \{(j, o') | j \in \mathcal{N}\}$: set of arcs



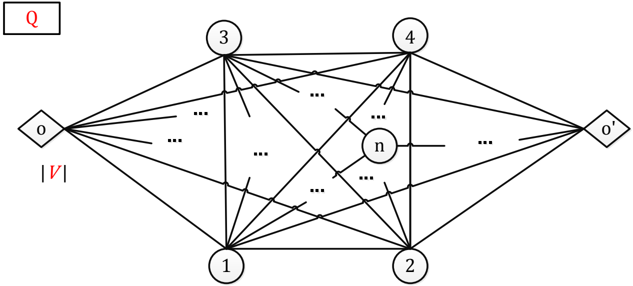
Notation

- c_{ij} : determinist travel cost from $i \in \mathcal{N}'$ à $j \in \mathcal{N}'$
- $p = (i_1, ..., i_h)$: route where $i_j \in \mathcal{N}'$ for $j \in \{1, ..., h\}$
- c_p : determinist travel cost of route p
- $\hat{c}_p$: total expected failure cost of route p
- α_{ip} : binary parameter indicating if route p visits client $i \in \mathcal{N}$ or not
- $\mathscr{P}$: set of all routes from o to o' which are feasible on average



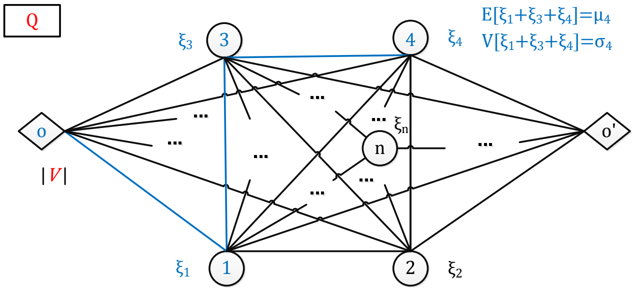
Notation

- $\mathcal{V}$: set of $|\mathcal{V}|$ identical vehicles
- Q : capacity of a vehicle



Notation

- $\xi_i \sim \Psi(\mathbb{E}_\xi[\xi_i], \mathbb{V}_\xi[\xi_i])$: random variable indicating demand of client $i \in \mathcal{N}$ following distribution Ψ ;
- Given path $p = (i_1, i_2, ... i_h)$
  - $\sum_{j=1}^{h} \mathbb{E}_\xi\left[\xi_{i_j}\right] = \mu_{i_h}$: expected cumulated demand at i_h
  - $\sum_{j=1}^{h} \mathbb{V}_\xi\left[\xi_{i_j}\right] = \sigma_{i_h}$: expected cumulated variance at i_h



Notation - decision variables

First stage

- x_{ij} : binary variable indicating if a vehicle follows the arc $(i,j) \in \mathcal{A}$.
- λ_p : binary variable indicating if we choose route p or not

Second stage

- $\mathcal{Q}(x)$: recourse function (expected failure cost of a given route)

Master Problem and Subproblem - VRPSD

$$
\begin{aligned}
\min_{\lambda} \quad & \sum_{p \in \mathscr{P}} \hat{c}_p \lambda_p \\
\text{s. t. :} \quad & \sum_{p \in \mathscr{P}} \alpha_{ip} \lambda_p = 1 \forall i \in \mathcal{N} \\
& \lambda_p \in \{0,1\} \quad \forall p \in \mathscr{P}
\end{aligned}
\tag{MP}
$$

$$
\begin{aligned}
\min_{x} \quad & \sum_{i \in \mathcal{N}'} \sum_{j \in \mathcal{N}'} \bar{c}_{ij} x_{ij} + \mathcal{Q}(x) \\
\text{s. à :} \quad & \sum_{(i,j) \in \delta^-(\{j\})} x_{ij} - \sum_{(j,i) \in \delta^+(\{j\})} x_{ji} = \begin{cases} -1 & \text{if } j = o \\ 0 & \text{else } \forall j \in \mathcal{N}' \\ 1 & \text{if } j = o' \end{cases} \\
& \sum_{i \in \mathcal{N}'} \sum_{j \in \mathcal{N}'} \mathbb{E}_{\xi} \left[\xi_j \right] x_{ij} \leq Q \\
& x_{ij} \in \{0,1\} \quad \forall i,j \in \mathcal{N}'
\end{aligned}
\tag{SP}
$$

Reduced cost

$$\bar{c}_{ij} = c_{ij} - \pi_j \text{ if } j \neq o'$$

Computation of $\mathcal{Q}(x)$

Given

- $p = (i_1, i_2, \ldots i_{h-1}, i_h) \subseteq \mathcal{N}'$ from $o = i_1$ to i_h

[1] Expected failure cost at client i_h for p

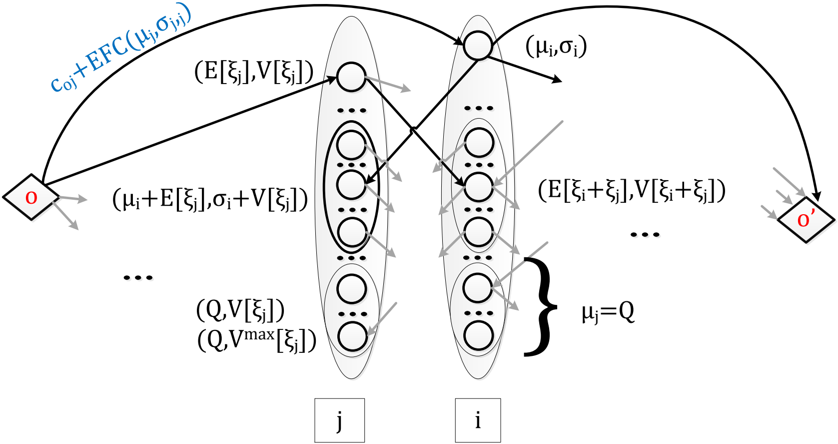
$$\text{EFC}(\mu_{i_h}, \sigma_{i_h}, i_h) = 2c_{oi_h} \sum_{u=1}^{\infty} (\mathbb{P}_\xi \left[\sum_{l=1}^{h-1} \xi_{i_l} \leq uQ < \sum_{l=1}^{h} \xi_{i_l} \right])$$

1-Dror & Trudeau (1987), Laporte et al. (2002), Christiansen & Lysgaard (2007)

Total cost of route p= determinist cost p + expected failure cost p

$$\hat{c}_p = \sum_{j=1}^{h} \left(c_{i_h i_{h+1}} + \text{EFC}(\mu_{i_{h+1}}, \sigma_{i_{h+1}}, i_{h+1}) \right)$$

Creation of the state-space graph $\mathcal{GS} = (\mathcal{NS}, \mathcal{AS})$



Numerical results

- Max alloted time : 20 minutes,
- 3,4 GHz and 16 GB of RAM vs 1,5GHz and 480 MB of RAM ⇒ scaling factor : 1.98
- Poisson demands + A, E, P instances

Instance	Base (This work)						Christiansen & Lysgaard (2007)			Acceleration factor
	Time	$\frac{\text{LB root}}{\text{UB}}$	# B & B Nodes	# Rounds of cuts	# Cuts CC	# Cuts SRI	Time	$\frac{\text{LB root}}{\text{UB}}$	# B & B Nodes	
A-n32-k5	24.1	0.98	0	7	6	90	142.4	0.96	2467	5.9
A-n33-k5	5.3	1	0	6	29	15	4	0.99	117	0.8
A-n33-k6	4.9	1	0	4	3	20	24.7	0.98	909	5.1
A-n34-k5	9.0	0.98	0	6	27	28	#	0.97	16059	$\geq$ 67.3
A-n36-k5	46.9	0.98	0	10	30	90	#	0.92	8035	$\geq$12.9
A-n37-k5	23.4	0.98	0	7	12	72	#	0.97	9191	$\geq$25.9
A-n37-k6	22.8	0.99	2	11	56	62	#	0.98	16195	$\geq$26.6
A-n38-k5	42.8	0.97	6	11	22	74	#	0.95	14499	$\geq$14.2
A-n39-k5	5.0	1	0	0	0	0	1.5	1	9	0.3
A-n39-k6	19.3	0.99	0	8	28	45	140.9	0.97	2431	7.3
A-n44-k6	125.0	0.99	12	21	21	189	#	0.98	11077	$\geq$4.9
A-n45-k6	86.3	0.98	2	11	9	120	#	0.9	9313	$\geq$7.0
A-n45-k7	30.5	1	0	9	8	70	445.5	0.99	5365	14.6
A-n46-k7	18.7	0.99	0	4	58	30	#	0.98	8149	$\geq$32.4
A-n48-k7	27.5	0.99	0	6	42	30	#	0.93	7729	$\geq$22.0
A-n53-k7	512.4	0.99	12	23	37	240	#	0.93	5385	$\geq$1.2

Numerical results (cont'd)

Instance	Base (This work)						Christiansen & Lysgaard (2007)			Acceleration factor
	Time	LB root / UB	# B & B Nodes	# Rounds of cuts	# Cuts CC	# Cuts SRI	Time	LB root / UB	# B & B Nodes	
A-n54-k7	310.5	0.99	6	18	55	126	#	0.94	5925	≥2.0
A-n55-k9	33.2	0.99	0	7	50	45	#	0.91	9979	≥18.3
A-n60-k9	#	0.42	20	35	49	219	#	0.93	6889	#
E-n22-k4	0.1	1	0	0	0	0	0.5	1	9	5
E-n33-k4	32.6	1	0	0	0	0	43.4	0.99	73	1.3
E-n51-k5	#	0.95	4	29	23	212	#	0.97	2771	#
P-n16-k8	0.0	1	0	1	1	0	0	1	17	#
P-n19-k2	9.7	0.94	0	7	4	60	77.3	0.94	1815	8.0
P-n20-k2	128.8	0.95	6	15	5	130	177.8	0.95	3191	1.4
P-n21-k2	2.2	1	0	1	3	0	2.5	0.99	27	1.1
P-n22-k2	46.7	0.97	0	8	3	90	110.6	0.97	1335	2.4
P-n22-k8	0.0	1	0	2	1	5	0	1	65	#
P-n23-k8	0.0	1	0	0	0	0	0.5	1	0	#
P-n40-k5	6.9	1	0	1	4	0	4.5	1	35	0.7
P-n45-k5	1193.8	0.98	12	34	22	272	#	0.95	4561	#
P-n50-k10	54.6	0.99	16	28	29	119	#	0.95	18901	≥11.1
P-n50-k7	40.4	0.99	0	11	26	75	#	0.95	5311	≥15.0
P-n50-k8	35.7	0.99	2	11	9	98	#	0.91	10409	≥17.0
P-n51-k10	10.0	0.99	0	8	21	56	217.2	0.99	5431	21.7
P-n55-k10	26.4	0.99	0	10	23	60	#	0.92	11257	≥23.0
P-n55-k15	19.9	1	36	32	23	61	400	0.99	20027	20.1
P-n55-k7	103.8	0.99	0	14	22	120	#	0.94	2441	≥5.8
P-n60-k10	417.5	0.99	34	52	42	273	#	0.95	4969	≥1.5
P-n60-k15	7.0	1	0	7	37	26	#	1	19029	≥86.6
Average	91.7	0.97	4.25	11.9	21	80.55	99.6	0.96	6284.93	13.6

A robust optimization approach

- C.E. Gounaris, W. Wiesemann, C.A. Floudas (2013). The Robust Capacitated Vehicle Routing Problem Under Demand Uncertainty. *Operations Research*, 61 (3), 677-693.

- The authors consider the generic case where the customer demands are supported on a polyhedron.
- Robust solutions must be feasible for all demand vectors in the polyhedron.
- Several interesting references.

VRP with stochastic customers

VRP with stochastic customers (VPRSC)

- Each customer has a given probability of requiring a visit.
- Problem grounded in the pioneering work of Jaillet (1985) on the Probabilistic Traveling Salesman Problem (PTSP).
- At first sight, the VRPSC is of no interest under the reoptimization approach.

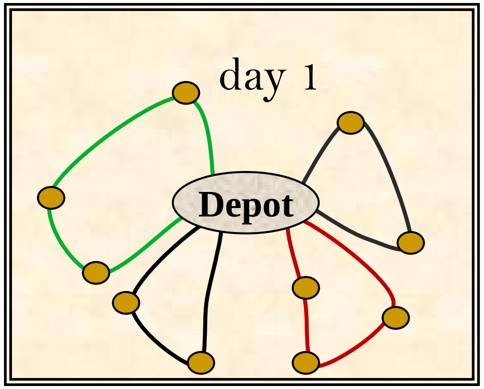
VRP with stochastic customers (VPRSC)

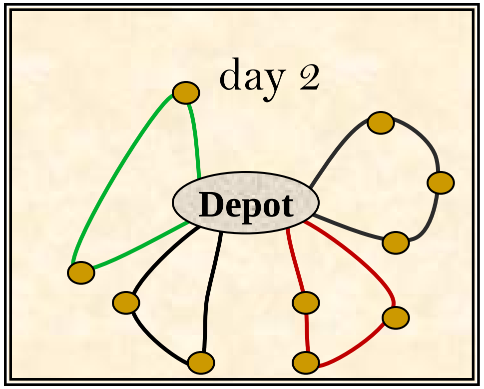
- Recourse action:
  - "Skip" absent customers
- Has been extensively studied by Gendreau, Laporte and Séguin in the 1990's:
  - Exact and heuristic solution approaches
- Can also be used to model the Consistent VRP (working paper with Ola Jabali and Walter Rei).

The Consistent VRP with Stochastic Customers

The consistent vehicle routing problem

- First introduced by Groër, Golden, and Wasil (2009)
  - Have the same driver visiting the same customers at roughly the same time each day that these customers need service
  - Focus is on the customer
  - Planning is done for D periods, known demand, m vehicles
  - Arrival time variation is no more than L
- Minimize travel time over D periods





Problem definition

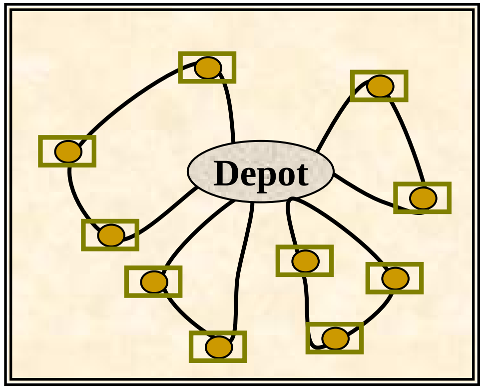
The consistent vehicle routing problem with stochastic customers

- Each customer has a probability of occurring
  - Same driver visits the same customers
  - A delivery time window is quoted to the customer
    $\rightarrow$ (Self-imposed TW)
- Cost structure
  - Penalties for early and late arrivals
  - Travel times

a priori approach

Stage 1
Plan routes and set targets

Stage 2
Compute travel times and penalties



Problem data

- An undirected graph $G=(V,A)$
  - $V=\{v_1,..,v_n\}$ is a set of vertices
  - $E=\{(v_i,\ v_j)$: $v_i,\ v_j\ \ V$, $i<j\}$ is a set of edges
  - Vertex v_1 corresponds to the depot
  - Vertices $v_2,..,v_n$ correspond to the potential clients
  - c_{ij} is the travel time between i and j
- m is the number of available vehicles
- A vehicle can travel at most λ hours
- p_i is the probability that client i places an order
- Ω is the set of possible scenarios associated with the occurrences for all customers

Model

- First-stage decision variables :
  - $x_{ij} = 1$, if client *j* is visited immediately after client *i* for $2 \leq i<j\leq n$, and 0 otherwise
  - x_{1j} can take the values 0,1 or 2
  - t_i, target arrival time at customer *i*
- ξ: a random vector containing all Bernoulli random variables associated with the customers.
- For each scenario $\omega \in \Omega$, let $\xi(\omega)^T = [\xi_2(\omega), \ldots, \xi_n(\omega)]$
  - $\xi_i(\omega) = 1$, if customer *i* is present and 0 otherwise.
- $\mathcal{Q}(x)$: second-stage cost (recourse)

Model

$$\min_x \mathrm{Q}(x) = E_\xi Q(x, \xi(\omega))$$

$$\sum_{i=1}^{n} x_{1i} = 2m$$

$$\sum_{i<k} x_{ik} + \sum_{j<k} x_{ik} \leq 2 \qquad v_k \in V \setminus v_1$$

$$\sum_{i,j} x_{ij} \leq |S| - \left\lceil \frac{l(S^+)}{\lambda} \right\rceil \qquad S \subset V \setminus v_1,\ 2 \leq |S| \leq n-2$$

$$0 \leq x_{0j} \leq 2 \qquad v_j \in V \setminus v_1$$

$$0 \leq x_{ij} \leq 1 \qquad 1 < i < j \leq n$$

$$x_{ij} \text{ integer} \qquad 1 \leq i < j \leq n$$

Model

Reformulation the objective function:

$$\min_{x,t} \tilde{c}^{T} x + \tilde{Q}(x)$$

$\tilde{c}^{T} x$ is a lower bound on the expected travel time
Gendreau, Laporte and Séguin (1995)

And

$$\tilde{Q}(x) = Q(x) - \tilde{c}^{T} x$$

Model

- Assumption: early arrivals do not wait for the time window
- Evaluation of the second stage cost

  $Q^{r,\delta}$: expected recourse cost corresponding to route r if orientation δ is chosen

  $Q_P{}^{r,\delta}$: total average penalties associated with time window deviations for route r if orientation δ is chosen

  $Q_T{}^{r}$: total average travel time for route for route r

$$Q^{r,\delta} = Q_T^r + Q_P^{r,\delta} \qquad\qquad Q(x) = \sum_{r=1}^{m} \min\{Q^{r,1}, Q^{r,2}\}$$

Model

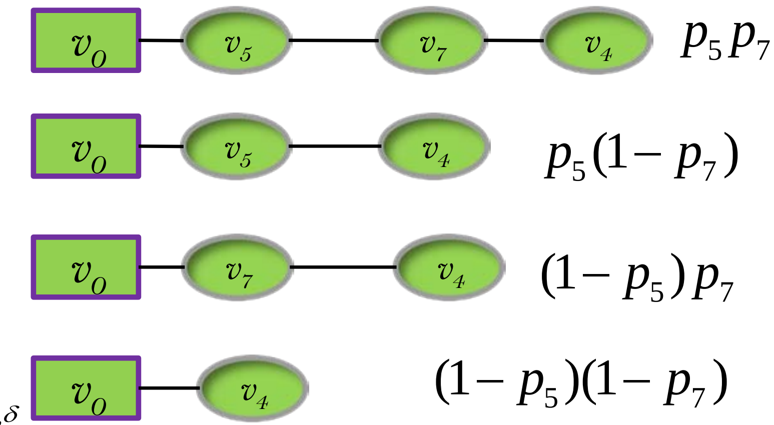
Given a route r, we relabel the vertices on the route according to a given orientation δ as follows:

$$(v_1 = v_{1_r}^{\delta}, v_{2_r}^{\delta}, \dots, v_{t_r}^{\delta}, v_{t_r+1}^{\delta} = v_1)$$

$\phi^{\delta}(v_{i_r})$: the minimum expected penalty associated with customer $v_{i_r}^{\delta}$

$$Q_P^{r,\delta} = \sum_{i=1}^{t_r+1} \phi^{\delta}(v_{i_r})$$

Model



Setting of $t_{i_r}^{\delta}$ and evaluation of $\phi^{\delta}(v_{i_r})$

Parameters:

$A_{i_r^\delta}$ – the collection of random events where customer $v_{i_r}^{\delta}$ requires a visit

w – half length of the time window

p_ω – probability of $\omega \in A_{i_r^\delta}$

$a_{i_r^\delta}(\omega)$ – arrival time at $v_{i_r}^{\delta}$ considering $\omega \in A_{i_r^\delta}$

β – late arrival penalty

Variables:

$t_{i_r}^{\delta}$ – target arrival time at customer $v_{i_r}^{\delta}$

$e_{i_r}^{\delta}$ $t_{i_r}^{\delta}$– early arrival at customer $v_{i_r}^{\delta}$ by $\omega \in A_{i_r^\delta}$

$l_{i_r}^{\delta}$ – late arrival at customer $v_{i_r}^{\delta}$ by $\omega \in A_{i_r^\delta}$

$$
\begin{aligned}
\phi^{\delta}(v_{i_r}) = \min \sum_{\omega \in A_{i_r^\delta}} & p_\omega\left(e_{i_r}^{\delta}(\omega) + \beta l_{i_r}^{\delta}(\omega)\right) \\
\text{s.t. } [t_{i_r}^{\delta} - w] - a_{i_r}^{\delta} \le e_{i_r}^{\delta}(\omega) & \quad \forall \omega \in A_{i_r^\delta} \\
a_{i_r}^{\delta} - [t_{i_r}^{\delta} + w] \le l_{i_r}^{\delta}(\omega) & \quad \forall \omega \in A_{i_r^\delta} \\
e_{i_r}^{\delta}(\omega), l_{i_r}^{\delta}(\omega) \ge 0 & \quad \forall \omega \in A_{i_r^\delta} \\
t_{i_r}^{\delta} \ge 0 &
\end{aligned}
$$

Solution procedure

Based on the Integer 0-1 L-Shaped Method proposed by Laporte and Louveaux (1993)

- Variant of branch-and-cut
  - Assumption 1: $Q(x)$ is computable
  - Assumption 2: There exists a finite value L = general lower bound for the recourse function.
- Operates on the current problem (CP) on each node of the search tree
  - In the VRP context, CP is relaxed:
    1. Integrality constraints
    2. Subtour elimination and route duration constraints
    3. $c^t x + Q(x) \rightarrow c^t x + \theta$

Preliminary results

Experimental sets:

- Vertices were generated similar to Laporte, Louveaux and van Hamme (2002)
- p values are randomly generated within 0.6 and 0.9
- 20 customers with 4 vehicles or 15 with 3 vehicles

Preliminary results

Set	N	Initial best integer	Initial best node	Initial GAP	Final solution	Final GAP	Run time
1	15	356.7	303.1	15.0%	332.8	<1%	64
2	15	352.5	264.3	25.0%	285.2	<1%	55
3	15	397.3	360.3	9.3%	388.1	<1%	263
4	15	363.3	303.0	16.6%	312.8	<1%	69
5	15	407.7	357.0	12.4%	393.3	<1	97
6	20	616.2	554.1	10.1%	597.2	<1%	879
7	20	486.4	486.4	6.2%	461.0	<1%	340
9	20	476.5	405.4	14.9%	451.3	<1%	27564
10	20	520.0	414.0	20.4%	455.1	<1	64638
11	20	449.4	368.5	18.0%	397.8	<1	67501
12	20	526.9	475.3	9.8%	478.6	<1	2242
13	20	474.0	436.5	7.9%	448.0	<1	1244
15	20	571.9	472.1	17.5%	486.7	9.35%	25200
16	20	444.3	397.7	10.5%	416.2	<1	25200
17	20	442.5	390.9	11.7%	414.2	3.13%	25200
8	20	494.4	401.0	18.9%	433.1	3.41%	86400
14	20	522.8	449.5	14.0%	503.5	1.53%	86400

VRP with stochastic service or travel times

VRP with stochastic service or travel times

- The travel times required to move between vertices and/or service times are random variables.
- The least studied, but possibly the most interesting of all SVRP variants.
- Reason: it is much more difficult than others, because delays “propagate” along a route.
- Usual recourse:
  - Pay penalties for soft time windows or overtime.
- All solution approaches seem relevant, but present significant implementation challenges.

VRP with stochastic service or travel times and soft time windows

The following material is taken from

- D. Tas, M. Gendreau, N. Dellaert, T. van Woensel, A.G. de Kok (2014). Vehicle Routing with Soft Time Windows and Stochastic Travel Times: A Column Generation and Branch-and-Price Solution Approach.

  European Journal of Operational Research, 236(3), 789-799.

- Considered model (introduced in Taş et al. 2013)
  - Distinguishes between transportation costs and service costs
  - Transportation costs: total distance + number of vehicles + total expected overtime
  - Service costs: early and late arrivals
- Solution approach
  - Column generation procedure
  - Master problem: set partitioning problem
  - Pricing subproblem: Elementary Shortest Path Problem with Resource Constraints (ESPPRC)
  - To generate an integer solution: embed our column generation procedure within a branch-and-price method

- $G = (N, A)$ where $N = \{0, 1, ..., n\}$ and $A = \{(i,j)|i,j \in N, i \neq j\}$
- Each customer $i \in N \setminus \{0\}$ has
  - a known demand ($q_i \geq 0$),
  - a fixed service duration ($s_i \geq 0$), and
  - a soft time window ($[l_i, u_i]$, $l_i \geq 0$, $u_i \geq 0$).
- No waiting!
- Weight on each arc $(i,j) \in A$, d_{ij}
- Capacity of each vehicle $v \in V$, Q

$$\min \quad \sum_{v\in V}\left[\rho\frac{1}{C_1}\left(c_d\sum_{j\in N}D_{jv}(\mathbf{x}) + c_e\sum_{j\in N}E_{jv}(\mathbf{x})\right)\right.$$
$$\left.+(1-\rho)\frac{1}{C_2}\left(c_t\sum_{i\in N}\sum_{j\in N}d_{ij}x_{ijv} + c_f\sum_{j\in N\setminus\{0\}}x_{0jv} + c_oO_v(\mathbf{x})\right)\right] \quad (1)$$

subject to
$$\sum_{j\in N}\sum_{v\in V}x_{ijv} = 1, \qquad i\in N\setminus\{0\}, \quad (2)$$
$$\sum_{i\in N}x_{ikv} - \sum_{j\in N}x_{kjv} = 0, \qquad k\in N\setminus\{0\}, v\in V, \quad (3)$$
$$\sum_{j\in N}x_{0jv} = 1, \qquad v\in V, \quad (4)$$
$$\sum_{i\in N}x_{i0v} = 1, \qquad v\in V, \quad (5)$$
$$\sum_{i\in N\setminus\{0\}}q_i\sum_{j\in N}x_{ijv} \le Q, \qquad v\in V, \quad (6)$$
$$\sum_{i\in B}\sum_{j\in B}x_{ijv} \le |B|-1, \qquad B\subseteq N\setminus\{0\}, v\in V, \quad (7)$$
$$x_{ijv}\in\{0,1\}, \qquad i\in N, j\in N, v\in V. \quad (8)$$

- Arrival time of vehicle v at node j:

$$Y_{jv} = \sum_{(l,k) \in A_{jv}} T_{lk} \tag{9}$$

- Random traversal time spent for one unit of distance $\rightarrow$ Gamma distributed
- Uncertainty per km
- Compute expected delay, earliness and overtime exactly

- Master problem: A set partitioning problem

$$\min \quad \sum_{p \in P} K_p y_p \tag{10}$$

$$\text{subject to} \sum_{p \in P} a_{ip} y_p = 1, \qquad i \in N \setminus \{0\}, \tag{11}$$

$$y_p \in \{0,1\}, \qquad p \in P. \tag{12}$$

- P: set of all feasible vehicle routes that start from and end at the depot,
- K_p: total weighted cost of route p,
- a_{ip}: equal to 1 if customer i is served by route p and 0, otherwise.

- Pricing subproblem: For each vehicle v, an Elementary Shortest Path Problem with Resource Constraints (ESPPRC)

$$\min \quad \overline{K}_p \tag{13}$$

$$\text{subject to} \quad (3)-(8). \tag{14}$$

- p: route of vehicle v,
- $\overline{K}_p$: reduced cost of route p.

$$\begin{aligned}\overline{K}_p =& K_p - \sum_{i\in N\setminus\{0\}} a_{ip}u_i \\ =& \rho\frac{1}{C_1}\left(c_d\sum_{j\in N} D_{jv}(\mathbf{x}) + c_e\sum_{j\in N} E_{jv}(\mathbf{x})\right) \\ &+(1-\rho)\frac{1}{C_2}\left(c_t\sum_{i\in N}\sum_{j\in N} d_{ij}x_{ijv} + c_f\sum_{j\in N\setminus\{0\}} x_{0jv} + c_oO_v(\mathbf{x})\right) - \sum_{i\in N\setminus\{0\}} a_{ip}u_i,\end{aligned} \tag{15}$$

- u_i, $i\in N\setminus\{0\}$: dual price associated with covering constraints (11).

TABLE: Results of problem instances in RC set with 25 customers with BF method

Instance	RootLB	RootUB	BestLB	BestUB	CPU	Gap%	Tree
RC101	2663.82	2688.85	2669.81	2682.70	1524.3	0.48	8
RC102	2656.18	2678.14	2663.84	2676.58	1399.7	0.48	8
RC103	2652.06	2675.63	2658.32	2670.58	1464.8	0.46	8
RC104	2651.93	2675.57	2658.23	2670.48	4321.4	0.46	10
RC105	2657.93	2684.02	2664.00	2677.30	1594.5	0.50	8
RC106	2651.72	2674.03	2659.46	2673.26	10800.0	0.52	12
RC107	2648.38	2672.19	2655.92	2669.05	2113.2	0.49	8
RC108	2648.18	2670.92	2654.35	2667.37	1956.2	0.49	8
RC201	2708.91	2716.07	2708.91	2716.07	18.5	0.26	0
RC202	2683.89	2689.12	2683.89	2689.12	17.0	0.19	0
RC203	2662.52	2674.81	2662.52	2674.81	16.9	0.46	0
RC204	2660.69	2674.31	2660.69	2673.94	126.8	0.50	4
RC205	2686.93	2697.12	2686.93	2697.12	22.3	0.38	0
RC206	2684.94	2696.37	2684.94	2696.37	15.5	0.43	0
RC207	2657.66	2683.02	2664.92	2680.39	10800.0	0.58	11
RC208	2648.18	2669.69	2654.06	2667.19	2047.8	0.49	8

TABLE: Results of problem instances in RC set with 25 customers with DF method

Instance	RootLB	RootUB	BestLB	BestUB	CPU	Gap%	Tree
RC101	2663.82	2895.13	2664.57	2687.43	10800.0	0.86	34
RC102	2656.18	2885.91	2656.18	2678.37	10800.0	0.84	25
RC103	2652.06	2903.37	2652.06	2677.50	10800.1	0.96	29
RC104	2651.93	2902.25	2651.93	2670.48	10800.6	0.70	31
RC105	2657.93	2886.28	2657.93	2681.08	10800.0	0.87	25
RC106	2651.72	2913.33	2651.84	2679.39	10800.3	1.04	29
RC107	2648.38	2908.49	2648.42	2674.13	10800.2	0.97	36
RC108	2648.18	2901.36	2648.18	2677.44	10800.0	1.10	37
RC201	2708.91	3050.71	2709.69	2721.21	142.3	0.43	8
RC202	2683.89	2994.67	2683.89	2689.62	105.7	0.21	5
RC203	2662.52	2929.48	2662.52	2674.46	3019.5	0.45	19
RC204	2660.69	2929.48	2660.69	2673.72	549.1	0.49	16
RC205	2686.93	2983.00	2686.93	2699.54	379.7	0.47	18
RC206	2684.94	2989.09	2686.64	2693.69	121.4	0.26	8
RC207	2657.66	2937.62	2657.66	2683.87	10800.0	0.99	28
RC208	2648.18	2902.54	2648.18	2677.90	10800.2	1.12	40

TABLE: Average results of problem instances in C, R and RC sets with 20, 25, 50 and 100 customers obtained by BF and DF methods

Set	Method	Avg. Gap%	Set	Method	Avg. Gap%
C1-20	BF	9.61	C1-20	DF	10.87
C2-20	BF	7.91	C2-20	DF	10.45
R1-20	BF	8.74	R1-20	DF	11.14
R2-20	BF	8.77	R2-20	DF	11.18
RC1-20	BF	0.46	RC1-20	DF	0.47
RC2-20	BF	0.40	RC2-20	DF	0.40
C1-25	BF	7.80	C1-25	DF	9.22
C2-25	BF	7.30	C2-25	DF	10.15
R1-25	BF	3.30	R1-25	DF	5.00
R2-25	BF	2.97	R2-25	DF	6.36
RC1-25	BF	0.49	RC1-25	DF	0.92
RC2-25	BF	0.41	RC2-25	DF	0.55
C1-50	BF	4.42	C1-50	DF	9.07
C2-50	BF	4.24	C2-50	DF	22.78
R1-50	BF	2.80	R1-50	DF	4.27
R2-50	BF	2.71	R2-50	DF	5.63
RC1-50	BF	2.76	RC1-50	DF	4.40
RC2-50	BF	2.67	RC2-50	DF	6.07
C1-100	BF	2.31	C1-100	DF	9.40

- Problem instances with 100 customers
- Limit for total CPU is set to 8 hours
- Applied strategy $\rightarrow$ DF method

TABLE: Average results of problem instances in C, R and RC sets with 100 customers obtained by DF method with 8 hour CPU limit

Set	Method	Avg. Gap%
C1-100	DF	7.26
C2-100	DF	22.45
R1-100	DF	4.01
RC1-100	DF	2.82

VRP with stochastic service times: a chance-constrained formulation

Following material from

- F. Errico, G. Desaulniers, M. Gendreau, W. Rei, L.-M. Rousseau. The Vehicle Routing Problem with hard time windows and stochastic service times.

  Forthcoming (hopefully…!)

Context

We consider a VRP with

- Stochastic service times
- Hard time windows
- No demands, nor vehicle capacity
- VRPTW-ST

Several applications :

- Dispatching of technicians or repairmen :
  - Perform specific services at the customers
  - Details of the service to perform are unknown beforehand
- Energy production planning :
  - Several power plants are connected in a network
  - Maintenance operations (implying outage) must planned in specific hard time windows (technicians are not available otherwise)
  - Duration of the operations is unknown beforehand

Related literature

- Several papers on VRP/m-TSP with stochastic travel times and customer deadlines or soft time windows (see Adulyasak and Jaillet, 2014)
- TSP with hard time windows and stochastic travel times in Jula et al. (2006), Chang et al. (2009)
  - Heuristic methods
- VRP with stochastic travel times, demand uncertainty and customer deadlines in Lee et al. (2012)
  - Robust optimization approach
- TSP with customers deadlines and stochastic customers in Campbell and Thomas (2008)
- **With respect to previous works, we aim to**
  - 1. Use chance-constrained stochastic model
    2. Use two-stage stochastic programming with recourse
  - Develop an exact solution method

Notation and assumptions

- A directed graph $G = (V, A)$, where
  - $V = \{0, 1, \ldots, n\}$ is the node set
    - 0 represents a depot
    - $V_c = \{1, \ldots, n\}$ the customer set,
  - $A = \{(i, j) \mid i, j \in V\}$ is the arc set.
- A non-negative travel cost c_{ij} and travel time t_{ij} are associated with each arc (i, j) in A.
- A hard time window $[a_i, b_i]$, $i \in V_c$
- A stochastic service time s_i, $i \in V_c$.
- Service time probability functions are supposed to be known and :
  - Discrete with finite support
  - Mutually independent

The VRPTW-ST with Chance Constraint

Definition (Successful Route)

Given a service time realization, a route is said Successful if :

(i) Route starts and ends in node 0 ;

(ii) Service at customers starts within the given time windows.

- Vehicles may arrive before the beginning of a time window.
- Late service time is not allowed

The VRPTW-ST finds a set of route such that :

1. Routes start and end in node 0 ;
2. Routes induce a proper partition of all customers
3. The global probability that the route plan is Successful is higher than a given reliability threshold $0 < \alpha < 1$;
4. The travel cost is minimized.

Formulation

- $\mathcal{R}$: set of all possible routes.
- $a_{ir} = 1$ prameter if route r visits customer i and 0 otherwise.
- c_r the cost associated with route r
- $x_r = 1$ binary variable if route r is chosen, 0 otherwise

Formulation :

$$\min \sum_{r\in\mathcal{R}} c_r x_r \tag{1}$$

$$s.t. \sum_{r\in\mathcal{R}} a_{ir} x_r = 1 \qquad \forall i \in V_c \tag{2}$$

$$\Pr\{\text{All routes are Successful}\} \geq \alpha \tag{3}$$

$$x_r \in \{0,1\} \qquad \forall r \in \mathcal{R}, \tag{4}$$

Linearization

Mutually independent service time $\Rightarrow$

Proposition

Let $\mathcal{R}'$ denote a set of routes inducing a proper partition of the customers set V_c. Given any two routes $r_1, r_2 \in \mathcal{R}'$, the success probability of r_1 is independent from the success probability of r_2.

Linearization

Mutually independent service time $\Rightarrow$

Proposition

Let $\mathcal{R}'$ denote a set of routes inducing a proper partition of the customers set V_c. Given any two routes $r_1, r_2 \in \mathcal{R}'$, the success probability of r_1 is independent from the success probability of r_2.

This can be used to linearize constraint (3) :

$$\Pr\{ \text{ All routes are Successful} \} \geq \alpha$$

Linearization

Mutually independent service time $\Rightarrow$

Proposition

Let $\mathcal{R}'$ denote a set of routes inducing a proper partition of the customers set V_c. Given any two routes $r_1, r_2 \in \mathcal{R}'$, the success probability of r_1 is independent from the success probability of r_2.

This can be used to linearize constraint (3) :

$$\prod_{r \in \mathcal{R}: x_r = 1} \Pr\{ \text{ Route } r \text{ is Successful } \} \geq \alpha,$$

Linearization

Mutually independent service time $\Rightarrow$

Proposition

Let $\mathcal{R}'$ denote a set of routes inducing a proper partition of the customers set V_c. Given any two routes $r_1, r_2 \in \mathcal{R}'$, the success probability of r_1 is independent from the success probability of r_2.

This can be used to linearize constraint (3) :

$$\sum_{r \in \mathcal{R}} x_r \ln(\Pr\{\text{ Route } r \text{ is Successful }\}) \geq ln(\alpha)$$

Linearization

Mutually independent service time $\Rightarrow$

Proposition

Let $\mathcal{R}'$ denote a set of routes inducing a proper partition of the customers set V_c. Given any two routes $r_1, r_2 \in \mathcal{R}'$, the success probability of r_1 is independent from the success probability of r_2.

This can be used to linearize constraint (3) :

$$\sum_{r \in \mathcal{R}} \beta_r x_r \leq \beta,$$

where

$$\begin{aligned} \beta_r &:= -\ln(\Pr\{\text{ Route } r \text{ is Successful }\}) \\ \beta &:= -ln(\alpha) \end{aligned}$$

Computing the route success probability (1)

Observations :

- Consider a route $r = (v_0, \ldots, v_q, v_{q+1})$ where v_0 and v_{q+1} are 0
- Consider $\bar{t}_{v_i}$ the random variable for the service starting time at customer v_i
- r is successful $\Leftrightarrow a_{v_i} \leq \bar{t}_{v_i} \leq b_{v_i}$, for all customers in r
- To compute the route success probability we need the probability distributions of $\bar{t}_{v_i}$
- $\bar{t}_{v_i}$ are sums of independent random variables
    - Their distribution can be computed by convolution
    - Under certain hypothesis, convolutions have nice properties (closed forms, etc)
    - **Not in hour case** : Time windows truncate/modify the distributions
- $\Rightarrow$ We actually need to carry out computations

Computing the route success probability (2)

- Starting service times $\bar{t}_{v_i}$ are linked to arrival times t_{v_i} :

$$\bar{t}_{v_i} = \begin{cases} a_{v_i} & t_{v_i} < a_{v_i} \\ t_{v_i} & a_{v_i} \leq t_{v_i} \leq b_{v_i} \end{cases}$$

- For the corresponding probability mass functions $m^t_{v_i}$ and $\bar{m}^t_{v_i}$

$$\bar{m}^t_{v_i}(z) = \begin{cases} 0 & z < a_{v_i}, \\ \sum_{l \leq a_{v_i}} m^t_{v_i}(l) & z = a_{v_i}, \\ m^t_{v_i}(z) & a_{v_i} < z \leq b_{v_i}, \\ 0 & z > b_{v_i}. \end{cases}$$

- Observe that for a given v_i :

$$\Pr\{r \text{ is Successful up to } v_i\} = \sum_{z \in \mathcal{N}} \bar{m}^t_{v_i}(z)$$

Computing the route success probability (3)

Simple iterative procedure :

1. for $(i = 1, \ldots, q-1)$ do
   a Truncation Step : Starting from $m^t_{v_i}$ obtain $\bar{m}^t_{v_i}$
   b Convolution Step : Compute $m^t_{v_{i+1}}(z) = (\bar{m}^t_{v_i} * m^s_{v_i})(z - t_{v_i, v_{i+1}}), \ \forall z \in \mathcal{N}$
2. Compute : $\Pr\{r \text{ is successful}\} = \sum_{z \in \mathcal{N}} \bar{m}^t_{v_q}(z)$

Critical point : algorithmic complexity depends on

- The quality of the time discretization
- The customer time windows widths
- The amplitude of the distribution supports

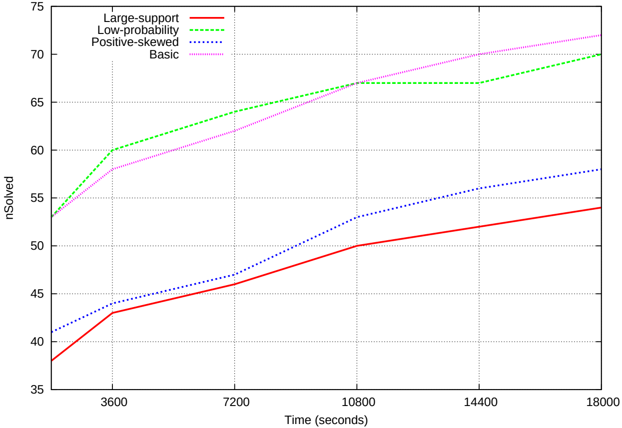
Instance set

Instances derived from the VRPTW database of Solomon (1987) :

- We build 4 instance families :
  1. **Basic** :
     - Symmetric triangular distributions
     - Median corresponding to original values : 100 for R and RC, 900 for C
     - Support : $[80, 120]$ for R and RC, $[700, 1100]$ for C.
     - Minimum success probability : $\alpha = 95\%$
  2. **Low-probability** :
     - Similar to Basic case, but the minimum success probability is $\alpha = 85\%$
  3. **Large-support** :
     - Similar to Basic case, but larger support : $[50, 150]$ for R and RC, $[450, 1350]$ for C
  4. **Positive-skewed** :
     - Similar to Large-support case, but different median values : 70 for R and RC, 630 for C
- Capacity and demand are disregarded
- Number of customers : 25 and 50 for R1, RC1, C1 ; 25 for R2, RC2, C2. (85 X 4 = 340 Total)
- Max CPU time : 5h on Intel i7-2600 3.40GHz, 16G RAM

Performance on benchmark instances (1)

Number of optimally solved instances (over 85) :



- Instance families with larger support are more difficult
- Approx. 80% of the instances are solved within the first hour

Deterministic VS Stochastic Model (1)

Deterministic (**Median values**) VS Stochastic (**Large-support**)

class	PercCostDAvg	PercVehDAvg	PercSuccDAvg	count
1	-6.8	-56.4	-44.91242	39
2	-0.1	0.0	-5.12083	15
	-5.0	-40.7	-33.85920	54

- General tendency : modest cost decrease $\Longleftrightarrow$ consistent decrease of success probability ($-5.0 \Longleftrightarrow -33.9\%$)
- Some differences :
  - Family 1 : $-6.8 \Longleftrightarrow -44.9\%$
  - Family 2 : $0.1 \Longleftrightarrow 5\%$
- Stochastic model is convenient

Deterministic VS Stochastic Model (2)

Deterministic (**Worst-case values**) VS Stochastic (**Large-support**)

class	PercCostDAvg	PercVehDAvg	PercSuccDAvg	count
1	9.6	74.4	2.93339	39
2	1.7	0.0	0.06861	15
	7.4	53.7	2.13762	54

- General tendency : relevant cost increase $\Longleftrightarrow$ small increase of success probability ($+7.4 \Longleftrightarrow +2.1\%$)
- Some differences :
  - Family 1 : $+9.6 \Longleftrightarrow +2.9\%$
  - Family 1 : $+1.7 \Longleftrightarrow +0.07\%$
- Stochastic model is still convenient

Conclusions and perspectives

Conclusion and perspectives

- Stochastic vehicle routing is a rich and promising research area.
- Much work remains to be done in the area of recourse definition.
- SVRP models and solution techniques may also be useful for tackling problems that are not really stochastic, but which exhibit similar structures
- Up to now, very little work on problems with stochastic travel and service times, while one may argue that travel or service times are uncertain in most routing problems!
- Correlation between uncertain parameters is possibly a major stumbling block in many application areas, but almost no one seems to work on ways to deal with it.