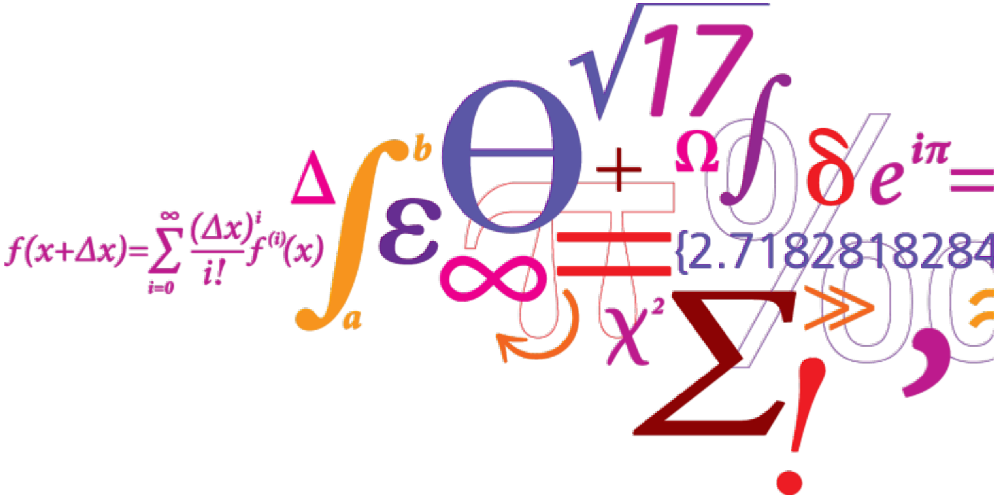
VRP with stochastic demands

Route 2014
June 2nd, 2014

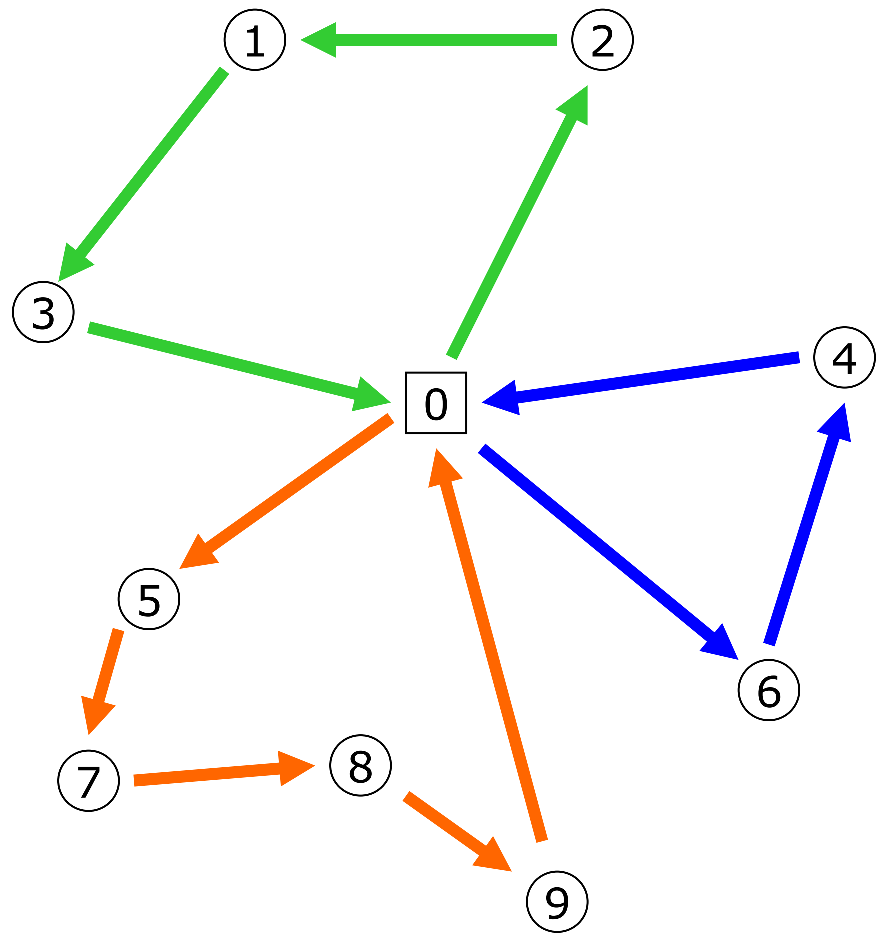
Trine Krogh Boomsma
Maria Schiøtt Eckhausen
Stefan Røpke



DTU Management Engineering
Department of Management Engineering

Capacitated vehicle routing problem static version

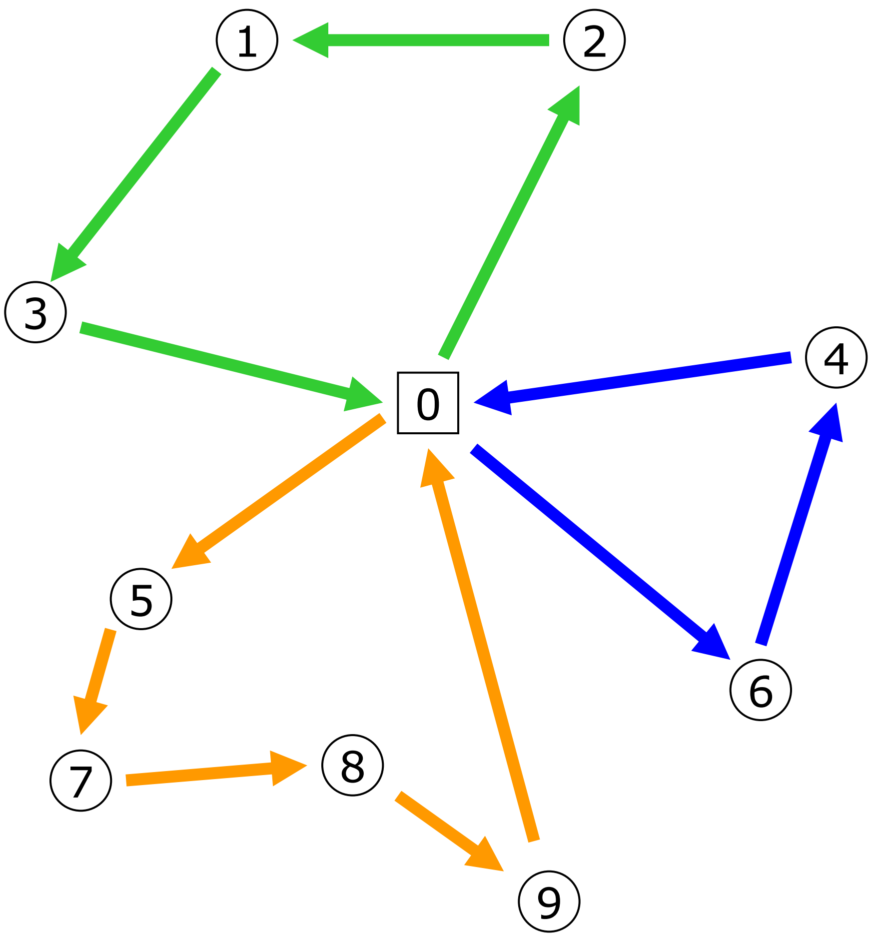
Capacitated vehicle routing problem (CVRP):
•Visit all customers exactly once.
•Each customer has a demand q_i.
•Each vehicle has a capacity Q.
•Sum of demands on a route ≤ vehicle capacity
•**Objective:** Minimize the cost of edges used in the solution.



VRP with stochastic demands (VRPSD)

vehicle routing problem with stocastic demand (VRPSD)

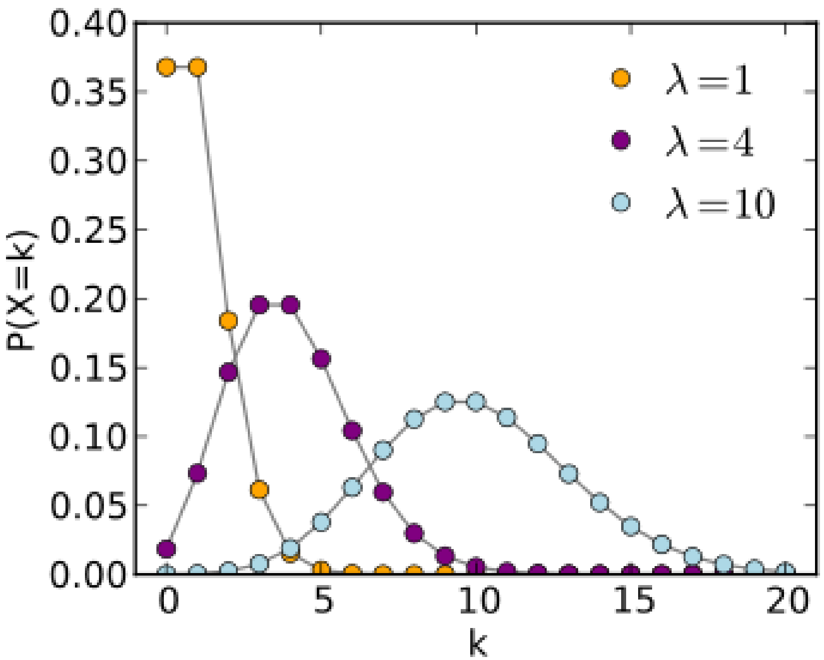
- Visit all customers.
- The demand of each customer *i* is given by a random variable q_i. Its probability distribution is assumed to be known. We assume the variables are independent.
- Each vehicle has a capacity Q.
- We have to plan routes in advance. When executing the plan it may turn out that we the planned route violate the capacity constraint. In that case the vehicle returns to the depot to empty/restock the vehicle.
- Expected demand on each route cannot exceed Q.
- **Objective:** Minimize the expected cost of the solution.

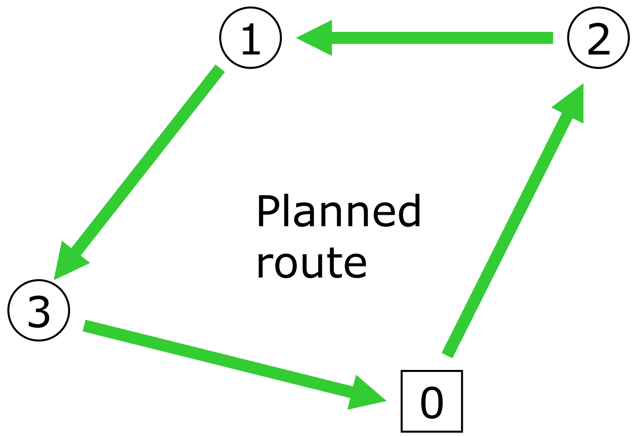


VRP with stochastic demands (VRPSD)

Vehicle capacity: 6

Poisson distribution - probability mass function



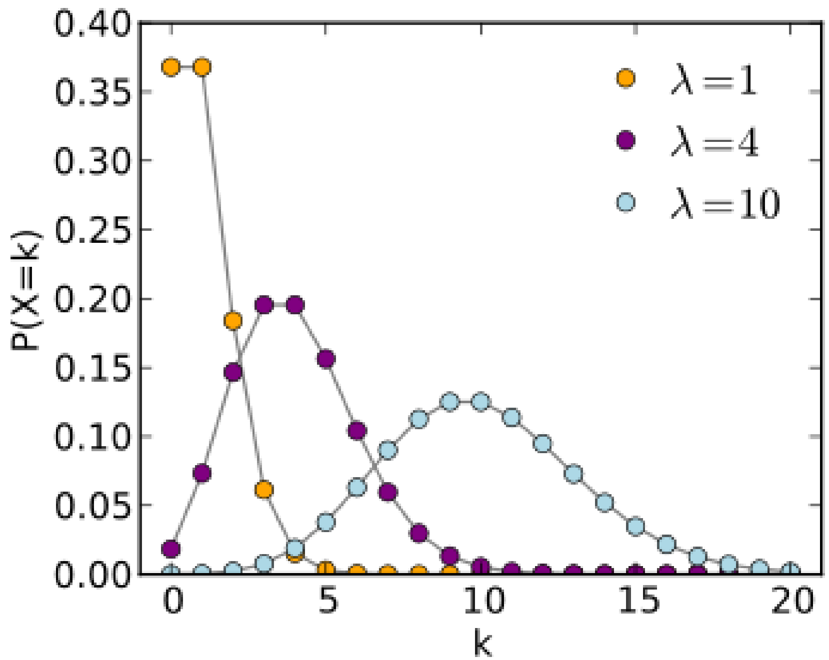


Assume all customers have expected demand 2.

One realization of the random variables: 3, 2, 2

VRP with stochastic demands

Poisson distribution - probability mass function

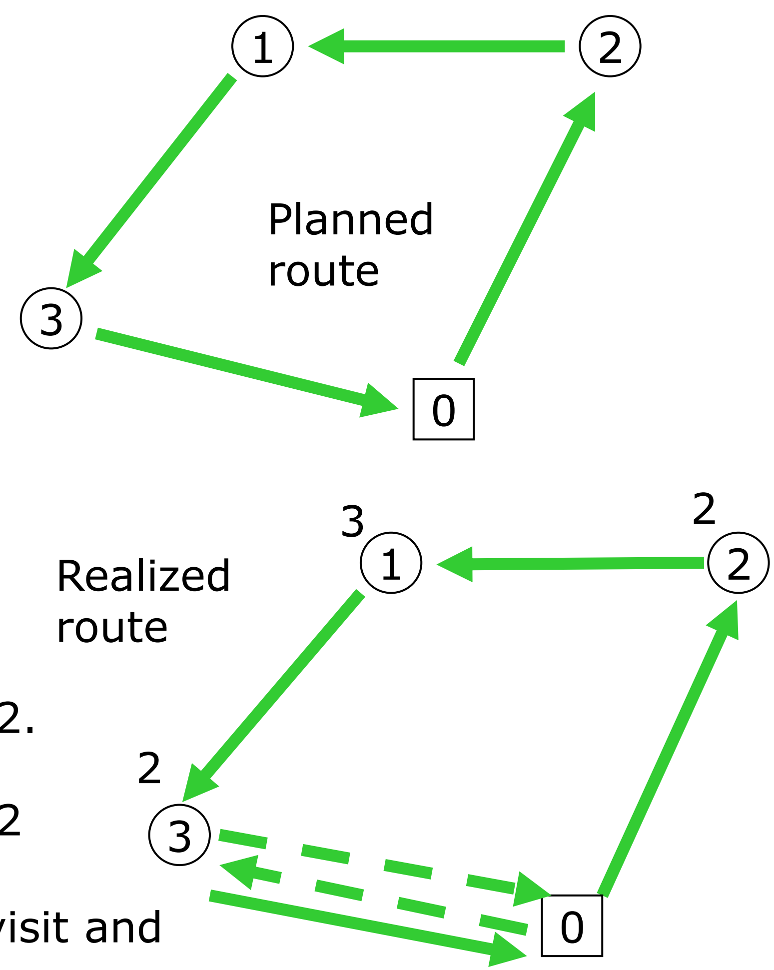


Assume all customers have expected demand 2.

One realization of the random variables: 3, 2, 2

We deliver one unit to customer 3 at the first visit and another at the second visit.

Vehicle capacity: 6



VRP with stochastic demands (VRPSD)

Vehicle capacity: 6

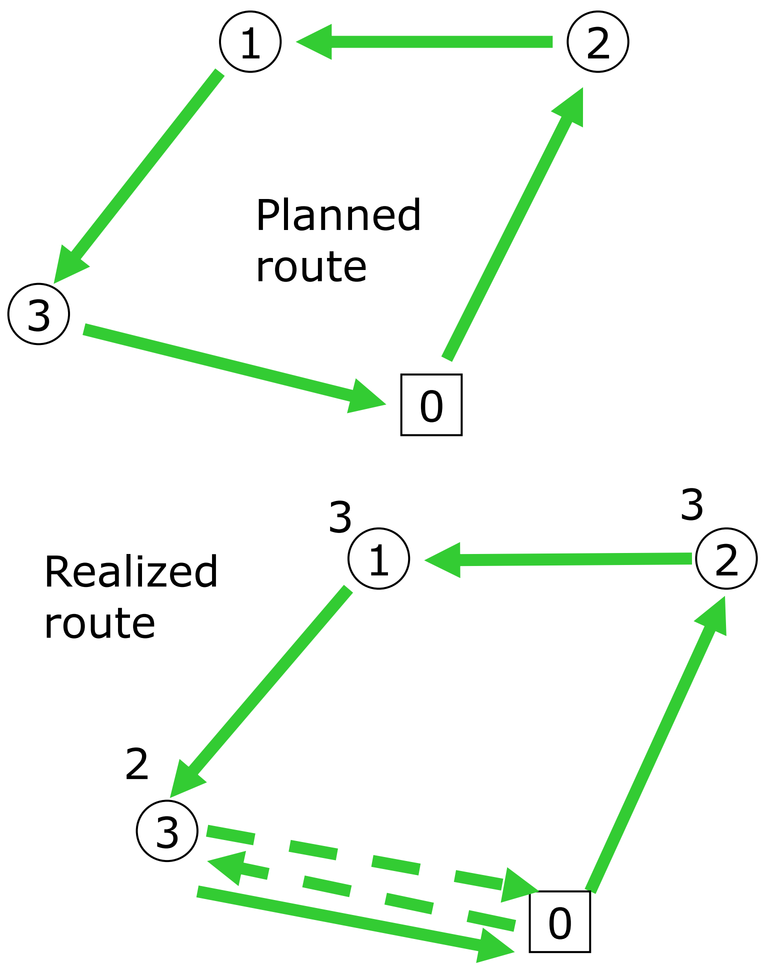
The extra trip back to the depot is the "recourse action".

We make decisions in stage 1 where we plan the routes for our vehicles.

In stage 2 the routes are followed blindly and any infeasibilities are "repaired" using the recourse action.

This type of problem is a *two-stage stochastic problem with recourse.*

Cost of extra trips to the depot is called *recourse cost.*

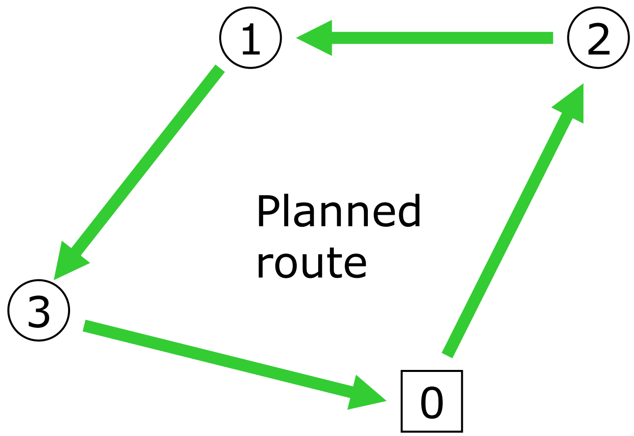


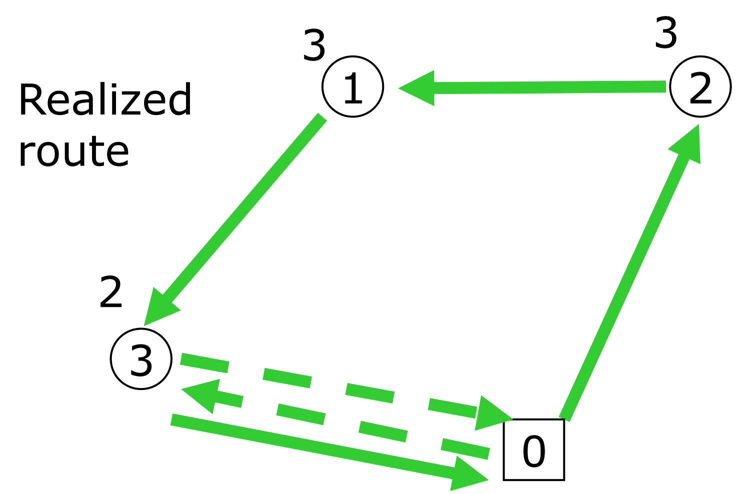
VRP with stochastic demands (VRPSD)

Vehicle capacity: 6

What if realization of the random variables was:
3, 2, 2 ?

In that case we will still continue to node 3 even though the vehicle is full after visiting node 1.





Recent exact methods

Paper	Method	Test data	Demand <= Q
Gendreau, Laporte, Seguin	Integer L-shaped method (branch and cut)	Uniform distributions in different intervals	Yes
Laporte Louveaux, Van Hamme	Integer L-shaped method (branch and cut)	Poisson and normal distribution	Yes
Jabali, Rei, Gendreau, Laporte, 2013	Integer L-shaped method (branch and cut)	Truncated normal distribution	yes
Christiansen & Lysgaard 2007	Branch and price	Poisson distribution	Yes

Formulating VRPSD using a set partitioning model

- $V_c = \{1, \ldots, n\}$. Set of customers.
- Ω : set of all feasible VRPSD routes.
- c_p : expected cost of route $p \in \Omega$.
- a_{ip} : constant that is 1 if customer i is visited by route p and 0 otherwise.
- y_p : binary variable that is 1 if and only if path $p \in \Omega$ is used in the solution.

$$\min \sum_{p \in \Omega} c_p y_p$$

Subject to:

$$\sum_{p \in \Omega} a_{ip} y_p = 1 \qquad \forall i \in V_c$$

$$y_p \in \{0, 1\} \qquad \forall p \in \Omega$$

Formulating VRPSD using a set partitioning model

- $V_c = \{1, \ldots, n\}$. Set of customers.
- Ω : set of all feasible VRPSD routes.
- c_p : expected cost of route $p \in \Omega$.
- a_{ip} : constant that is 1 if customer i is visited by route p and 0 otherwise.
- y_p : binary variable that is 1 if and only if path $p \in \Omega$ is used in the solution.

Expected cost consist of:

- Normal cost of traversed arc +
- Expected cost of recourse actions

min

Subject to:

$$\sum_{p \in \Omega} a_{ip} y_p = 1 \qquad \forall i \in V_c$$

$$y_p \in \{0, 1\} \qquad \forall p \in \Omega$$

Properties of set partitioning approach

- LP relaxation of set partitioning problem is solved by column generation
- ... and the entire problem is solved by branch-(and-cut)-and-price.
- Stochasticity is ”removed” from master problem and ”hidden” away in definition of variables.
- If we can compute expected recourse cost correctly when solving the pricing problem then we can also solve the entire problem.
- We can reuse many techniques developed for static CVRP since stochasticity is hidden away in sub-problem.

- Approach was first proposed in:
- C.H. Christiansen and J. Lysgaard. *A branch-and-price algorithm for the capacitated vehicle routing problem with stochastic demands*. Operations Research Letters, 35:773?781, 2007

Calculating expected recourse cost of a route

- Let $p(s)$ be the customer at position s in the route and let ξ_i be the random variable associated with the demand of customer i.

- The probability of having the first failure at customer j is:

$$\mathbb{P}\Big(\sum_{s=1}^{j-1}\xi_{p(s)} \leq Q < \sum_{s=1}^{j}\xi_{p(s)}\Big)$$

The probability of having the l'th failure at customer j is:

$$\mathbb{P}\Big(\sum_{s=1}^{j-1}\xi_{p(s)} \leq lQ < \sum_{s=1}^{j}\xi_{p(s)}\Big)$$

the expected recourse cost associated with customer j is

$$\sum_{l=1}^{\infty}\mathbb{P}\Big(\sum_{s=1}^{j-1}\xi_{p(s)} \leq lQ < \sum_{s=1}^{j}\xi_{p(s)}\Big)2c_{0,p(j)}$$

the expected recourse cost of the route is (the number of customers on the route is t)

$$\mathcal{Q}_r = \sum_{j=1}^{t}\sum_{l=1}^{\infty}\mathbb{P}\Big(\sum_{s=1}^{j-1}\xi_{p(s)} \leq lQ < \sum_{s=1}^{j}\xi_{p(s)}\Big)2c_{0,p(j)}$$

Calculating expected recourse cost of a route

(from last slide): The expected recourse cost of the route is (the number of customers on the route is t)

$$\mathcal{Q}_r = \sum_{j=1}^{t} \sum_{l=1}^{\infty} \mathbb{P}\left(\sum_{s=1}^{j-1} \xi_{p(s)} \le lQ < \sum_{s=1}^{j} \xi_{p(s)} \right) 2c_{0,p(j)}$$

Define

$$F^j(x) = \mathbb{P}\left(\sum_{s=1}^{j} \xi_{p(s)} \le x \right)$$

Notice that F^j is the cumultative distribution function of the random variable $\sum_{s=1}^{j} \xi_{p(s)}$. The expected recourse cost of the route can be rewritten

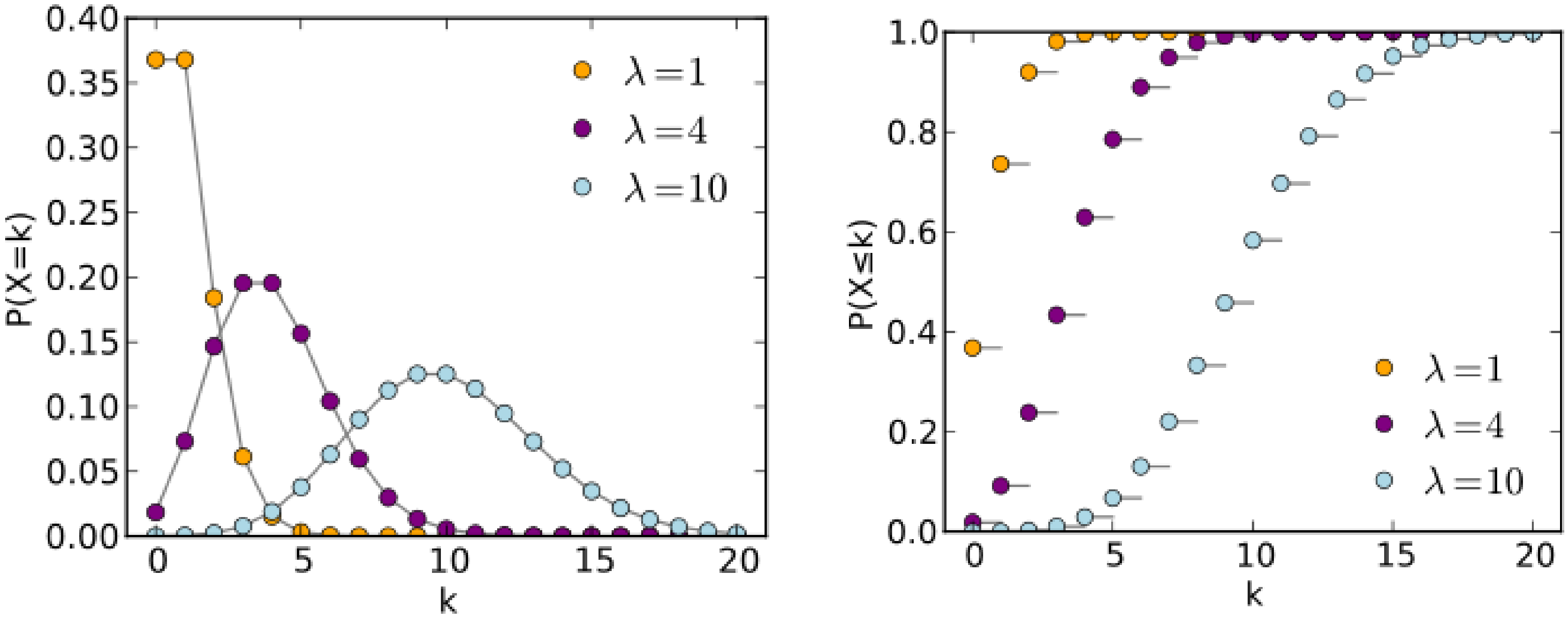
$$\mathcal{Q}_r = 2 \sum_{j=1}^{t} \sum_{l=1}^{\infty} \left(F^{j-1}(lQ) - F^j(lQ) \right) c_{0i_j}$$

Assumptions needed

- Customer demands are independent random variables
- All demands should follow the same distribution Ψ.
- The distribution (Ψ) should have the accumulative property
  - That is, the sum of two or more Ψ-distributed random variables is itself a Ψ –distributed random variable
  - We should be able to describe the distribution of each random variable by a number of parameters
- Examples of distributions that have the accumulative property: normal, Gamma, poisson, exponential distribution.

- In the following we use the poison distribution

Poisson distribution



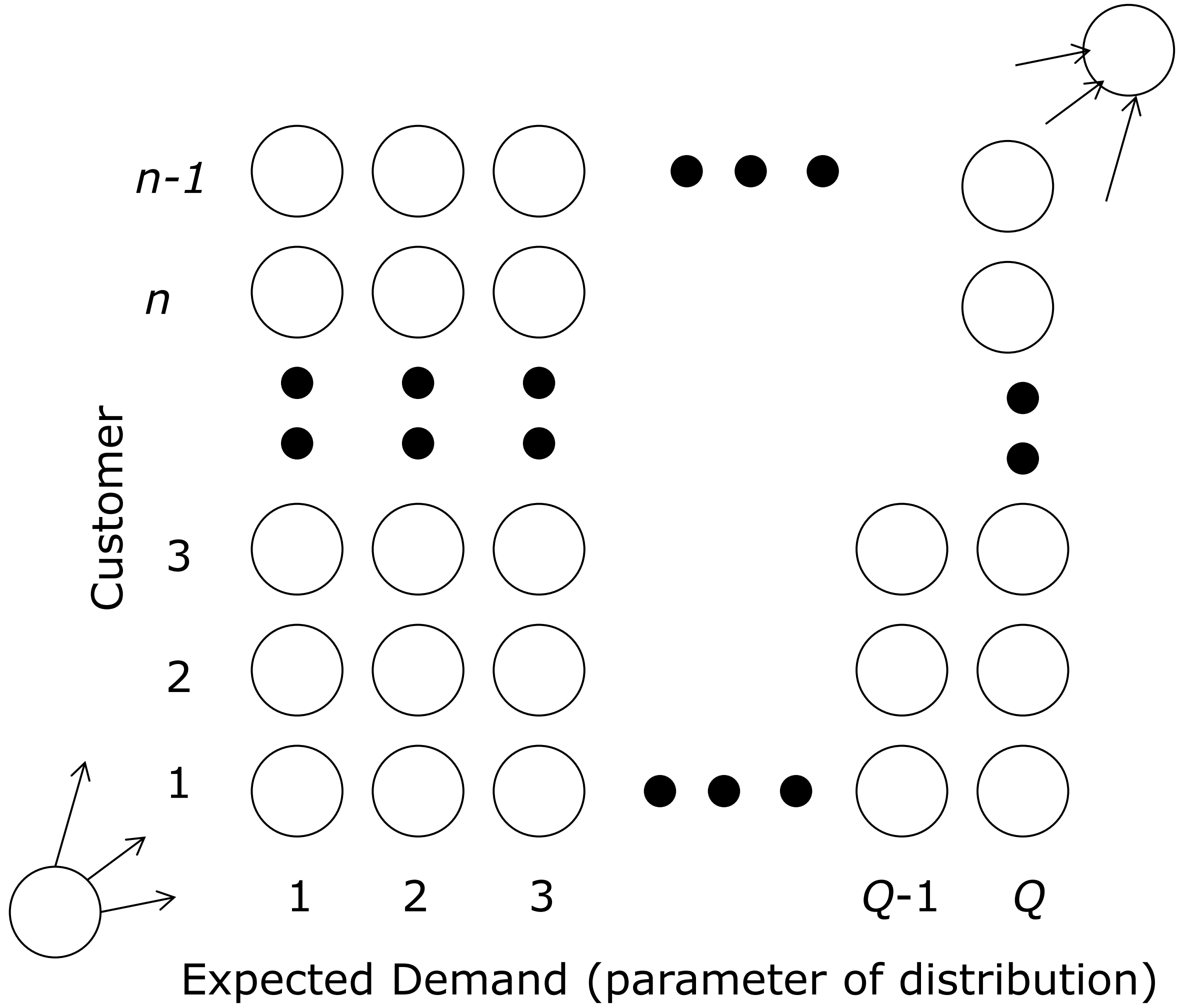
if ξ_i and ξ_j have a Poisson distribution with parameter λ_i and λ_j, respectively then $\xi_i + \xi_j$ is Poisson distributed with parameter $\lambda_i + \lambda_j$

Christiansen/Lysgaard insight

- Let P_1 and P_2 be two different partial routes from the depot to customer i.
- Let $\xi(P_j)$ be the random variable corresponding to the sum of the variables corresponding customers visited by route P_j.
- Assume that $\xi(P_1)$ and $\xi(P_2)$ have the same distribution.
- Assume further that the expected cost of P_1 is lower than that of P_2 and that any extension of P_2 also is feasible for P_1
- In this case P_1 dominates P_2.

Christiansen/Lysgaard pricing algorithm (Poisson distribution)

- Shortest path computation in an expanded graph.
- Special care taken to eliminates two-cycles (ordinary shortst path computation would allow them.
- Higher order cycles are possible.



Christiansen/Lysgaard pricing algorithm - interpretted as labeling algorithm

- The Chrisitansen / Lysgaard approach can be implemented with a labeling algorithm with labels
$$L = (i, c, \lambda)$$
with i being node, c expected cost of the partial path represented by label, λ parameter for distribution. A label is feasible if $\lambda \leq Q$.

- Dominance: Label L_1 dominates L_2 if
$$i_1 = i_2 \wedge c_1 \leq c_2 \wedge \lambda_1 = \lambda_2$$

- **We show that this can be improved to**:
$$i_1 = i_2 \wedge c_1 \leq c_2 \wedge \lambda_1 \leq \lambda_2$$
for the poisson distribution and
$$i_1 = i_2 \wedge c_1 \leq c_2 \wedge \frac{1}{\lambda_1} \leq \frac{1}{\lambda_2}$$
for the exponential distribution.

Improved dominance for Poisson distribution

- Remember, the expected recourse cost of a route can be written

$$\mathcal{Q}_r = 2\sum_{j=1}^{t}\sum_{l=1}^{\infty}\left(F^{j-1}(lQ) - F^{j}(lQ)\right) c_{0i_j}$$

with

$$F^{j}(x) = \mathbb{P}\left(\sum_{s=1}^{j}\xi_{p(s)} \le x\right)$$

Improved dominance for Poisson distribution

- New notation: $F^{poi}(x, \lambda)$ is the cumulative distribution function for Poisson distribution with parameter λ.

- We extend L_1 by a partial route with total expected demand $d + \Delta$ with Δ being the difference in expected demand between last and second to last customer on extension. Probability of l'th failure at last customer of extension:

$$F^{poi}(lQ, \lambda_1 + d) - F^{poi}(lQ, \lambda_1 + d + \Delta)$$

Similar for label L_2:

$$F^{poi}(lQ, \lambda_2 + d) - F^{poi}(lQ, \lambda_2 + d + \Delta)$$

If we can show

$$F^{poi}(lQ, \lambda_1+d)-F^{poi}(lQ, \lambda_1+d+\Delta) \leq F^{poi}(lQ, \lambda_2+d)-F^{poi}(lQ, \lambda_2+d+\Delta)$$

for all $l \in \mathbb{N}, d \geq 0, \Delta > 0, \lambda_2 + d + \Delta \leq Q$ then probability of failure for all feasible extensions of L_2 is higher than the same extension of L_1.

Improved dominance for Poisson distribution

- New notation: … ion for Poisson distribution …
- We extend L_1 b… $d + \Delta$ with Δ being the dif… d second to last customer o… customer of extension:

this follows from $f(\lambda) = F^{poi}(x, \lambda)$ (x is a constant here) being concave and nondecreasing in the interval]0,x]

$$F^{poi}(lQ, \lambda_1 + d) - F^{poi}(lQ, \lambda_1 + d + \Delta)$$

Similar for label L_2:

$$F^{poi}(lQ, \lambda_2 + d) - F^{poi}(lQ, \lambda_2 + d + \Delta)$$

If we can show

$$F^{poi}(lQ, \lambda_1+d)-F^{poi}(lQ, \lambda_1+d+\Delta) \leq F^{poi}(lQ, \lambda_2+d)-F^{poi}(lQ, \lambda_2+d+\Delta)$$

for all $l \in \mathbb{N}, d \geq 0, \Delta > 0, \lambda_2 + d + \Delta \leq Q$ then probability of failure for all feasible extensions of L_2 is higher than the same extension of L_1.

Other improvements compared to Christiansen/Lysgaard approach

- Eliminating more cycles using ng-routes
- Adding valid inequalities known from the CVRP.
- Strong branching

Results (all with Poisson distributed demands)

	BCP				Christiansen & Lysgaard			
	Time (s)	Opt	Root LB	BB	Time (s)	Opt	LB root	BB
A-32-5	7	X	843.80	853.60	282	X	817.31	853.6
A-33-5	2	X	702.64	704.20	8	X	700.01	704.2
A-33-6	2	X	792.81	793.90	49	X	775	793.9
A-34-5	5	X	822.78	826.87	-		803.26	825.26
A-36-5	9	X	851.74	858.71	-		838.83	852.09
A-37-5	27	X	700.06	708.34	-		687.4	707.54
A-37-6	6	X	1025.66	1031.16	-		1007.98	1030.44
A-38-5	24	X	768.30	775.14	-		739.19	761.12
A-39-5	1	X	869.18	869.18	3	X	866.92	869.18
A-39-6	9	X	873.24	876.60	279	X	850.09	876.6
A-44-6	52	X	1019.01	1025.48	-		1007.55	1021.29
A-45-6	20	X	1017.18	1026.73	-		984.38	1006.88
A-45-7	19	X	1261.44	1264.83	882	X	1254.23	1264.83
A-46-7	9	X	999.47	1002.22	-		986.39	999.87
A-48-7	14	X	1183.99	1187.14	-		1160.52	1180.22
A-53-7	123	X	1112.89	1124.27	-		1093.64	1109.34
A-54-7	137	X	1280.69	1287.07	-		1262.49	1279.93
A-55-9	15	X	1175.88	1179.11	-		1148.4	1173.56

BCP. TL 7200 sec, Intel Xeon 2.66Ghz
C&L. TL 1200 sec, Pentium Centrino 1.5Ghz. Guestimate: 3 times slower than Xeon.

Results (all with Poisson distributed demands)

	BCP				Christiansen & Lysgaard			
	Time (s)	Opt	Root LB	BB	Time (s)	Opt	LB root	BB
A-60-9	7215	-	1508.45	1522.83	-		1489.82	1503.65
E-22-4	0	X	411.57	411.57	1	X	409.86	411.57
E-33-4	5	X	850.27	850.27	86	X	844.35	850.27
E-51-5	1298	X	542.12	553.26	-		538.75	544.63
P-16-8	0	X	512.819	512.819	0	X	511.27	512.82
P-19-2	6	X	222.564	224.062	153	X	210.9	224.06
P-20-2	46	X	225.279	233.054	352	X	221.11	233.05
P-21-2	0	X	218.962	218.962	5	X	217.75	218.96
P-22-2	27	X	225.757	231.341	219	X	223.67	231.26
P-22-8	0	X	681.06	681.06	0	X	677.97	681.06
P-23-8	0	X	619.527	619.527	1	X	619.52	619.52
P-40-5	2	X	472.497	472.497	9	X	471.47	472.5
P-45-5	203	X	525.948	533.522	-		519.03	527.77
P-50-7	17	X	578.419	582.371	-		573.66	581.19
P-50-8	40	X	663.973	669.23	-		659.67	666.9
P-50-10	21	X	752.082	758.764	-		750.27	756.52
P-51-10	6	X	807.477	809.7	430	X	802.58	809.7
P-55-7	73	X	585.134	588.563	-		582.12	585.47
P-55-10	42	X	736.556	742.41	-		734.69	740.02
P-55-15	10	X	1064.57	1068.92	792	X	1062.67	1068.05
P-60-10	220	X	796.201	803.604	-		793.71	798.63
P-60-15	8	X	1083.9	1085.49	-	X	1080.85	1085.12

BCP. TL 7200 sec, Intel Xeon 2.66Ghz
C&L. TL 1200 sec, Pentium Centrino 1.5Ghz. Guestimate: 3 times slower than Xeon.

More tests

- Tested on 91 instances based on standard CVRP instances.
- 69 could be solved to optimality within 2 hours.
- Largest solved instance: 100 customers.
- Smallest unsolved instance: 33 customers.

Conclusion

- Significant improvement compared to existing branch-and-price algorithm.
- Approach is promising – at least for some distributions

- **Future work (to name a few directions)**
- Compare to integer L-shaped methods (branch and cut).
- Check if improved dominance work for more distributions
- Understand why some small instances are difficult (poor lower bounds).
- Understand the role of each improvement suggested.