E-companion for manuscript entitled "Smooth and flexible dual optimal inequalities"

Naveed Haghani Claudio Contardo Julian Yarkony

March 4, 2021

1 Proofs

In this appendix we present the proofs of the different lemmas and propositions stated in the main manuscript.

Proof of Proposition 1. Let (θ^*, ω^*) be an optimal solution to problem (4)-(7). If multiple optima exist, let (θ^*, ω^*) be the one attaining the lowest possible value of $\Sigma(\theta^*, \omega^*) = \sum\{\theta^*_l : l \in \Omega\} + \sum\{\omega^*_s : s \in \mathcal{S}\}$. We will prove, by contradiction, that no variable ω^*_s can take a strictly positive value. Let us assume that $\omega^*_s > 0$ for a certain $s \in \mathcal{S}$ and let us consider the following three scenarios:

- **There exists** $t \in \mathcal{S}$ **such that** $\omega^*_t > 0, s^- = t^+, s^+ \neq t^-$.

  Let $r = (t^-, s^+)$. Because of the triangle inequality, r lies in S and $\rho_r \leq \rho_s + \rho_t$. Let $\Delta = \min\{\omega^*_s, \omega^*_t\}$. We let $\omega'_s \leftarrow \omega^*_s - \Delta$, $\omega'_t \leftarrow \omega^*_t - \Delta$, $\omega'_r \leftarrow \omega^*_r + \Delta$ and $\omega'_k \leftarrow \omega^*_k$ for all $k \in \mathcal{S} \setminus \{s, t, r\}$. It is easy to see that (θ^*, ω') is primal feasible. The marginal contribution of replacing ω^* by ω' is $\Delta(\rho_r - \rho_s - \rho_t)$, which is non-positive. If negative (meaning that $\Delta(\rho_r - \rho_s - \rho_t) < 0$) we obtain a contradiction with the optimality of (θ^*, ω^*). If zero, on the other hand, the operation provides an alternate optimal solution (θ^*, ω') such that $\Sigma(\theta^*, \omega') < \Sigma(\theta^*, \omega^*)$, which is also a contradiction.

- **There exists** $t \in \mathcal{S}$ **such that** $\omega^*_t > 0, s^- = t^+, s^+ = t^-$.

  In this case we let $\Delta = \min\{\omega^*_s, \omega^*_t\}$ and let $\omega'_s \leftarrow \omega^*_s - \Delta$, $\omega'_t \leftarrow \omega^*_t - \Delta$, and $\omega'_k \leftarrow \omega^*_k$ for all $k \in \mathcal{S} \setminus \{s, t\}$. It is easy to see that the solution (θ^*, ω') is also primal feasible and the marginal contribution of this operation is $-\Delta(\rho_s + \rho_t)$.

The term $(\rho_s + \rho_t)$ is non-negative because of the following observation. From the definition of $\Omega(\cdot)$ we have that $l \in \Omega(s), l' = swap(l, s) \Leftrightarrow l' \in \Omega(t), l = swap(l', t)$. Then, for any such pair (l, l') we have —from the conditions satisfied by ρ— that $\rho_s + \rho_t \geq [c_{l'} - c_l] + [c_l - c_{l'}] \geq 0$. If $\rho_s + \rho_t > 0$, this operation results in a contradiction with the optimality of (θ^*, ω^*). If zero, on the other hand, the solution obtained is an alternate optimum and (θ^*, ω') is such that $\Sigma(\theta^*, \omega') < \Sigma(\theta^*, \omega^*)$, which is also a contradiction.

- **For every** $t \in \mathcal{S}$ **such that** $s^- = t^+$**,** $\omega_t = 0$.

  Since the primal problem is feasible and no $t \in \mathcal{S}$ exists satisfying $\omega_t^* > 0, s^- = t^+$, there must exist a column $l \in \Omega(s)$ such that $\theta_l^* > 0$. Let $\Delta = \min\{\omega_s^*, \theta_l^*\}$ and let $l' = swap(l, s)$. We construct new variables θ', ω' with all components equal to those of (θ^*, ω^*) except for $\omega_s' \leftarrow \omega_s^* - \Delta$, $\theta_l' \leftarrow \theta_l^* - \Delta$, $\theta_{l'}' \leftarrow \theta_{l'}^* + \Delta$. This operation entails an increase in the objective of $\Delta(c_{l'} - c_l - \rho_s)$ which by definition of ρ_s is non-positive. If negative, this would contradict the optimality of (θ^*, ω^*). If zero, on the other hand, it would entail an alternate optimum such that $\Sigma(\theta', \omega') < \Sigma(\theta^*, \omega^*)$ which is also a contradiction.

□

Proof of Proposition 2. Let (θ^*, ξ^*) be an optimal solution to problem (8)-(12). If multiple such optima exist, let (θ^*, ξ^*) be the one that minimizes $\Sigma(\theta, \xi) = \sum\{\theta_l : l \in \Omega\} + \sum\{\xi_{u\sigma} : u \in N, \sigma \in \Lambda_u\}$. We will prove by contradiction that if $\xi_{u\sigma}^* > 0$ for some $u \in N, \sigma \in \Lambda_u$ then either (θ^*, ξ^*) cannot be optimum, or that an alternate optimum (θ', ξ') exists such that $\Sigma(\theta', \xi') < \Sigma(\theta^*, \xi^*)$.

Let $\xi_{u\sigma}^* > 0$ for some $u \in N, \sigma \in \Lambda_u$. Constraints (10) assure the existence of at least one column $l \in \Omega$ such that $\beta_{ul\sigma} = 1$ and $\theta_l^* > 0$. Let $S = \{u \in N : a_{ul} = 1, \xi_{u\sigma_{ul}}^* > 0\}$ be the subset of items covered by l that are associated with a strictly positive value of $\xi_{u\sigma_{ul}}^*$. Let $\Delta = \min\{\min_{u \in S} \xi_{u\sigma_{ul}}^*, \theta_l^*\}$. Let $l' = remove(l, S)$ be the column resulting from removing all items in S from l. Now, let us consider a solution (θ', ξ') with all entries equal to those of (θ^*, ξ^*) except for the entries $\xi_{u\sigma_{ul}}' \leftarrow \xi_{u\sigma_{ul}}^* - \Delta$ for every $u \in S$, $\theta_l' \leftarrow \theta_l^* - \Delta$, $\theta_{l'}' \leftarrow \theta_{l'}^* + \Delta$. This operation entails a feasible solution with a marginal contribution to the objective equal to $\Delta(c_{l'} - c_l + \sum_{u \in S} \sigma_{ul})$. Either this quantity is negative —which would contradict the optimality of (θ^*, ξ^*)— or $\Sigma(\theta', \xi') < \Sigma(\theta^*, \xi^*)$ which is also not possible. □

Proof of Proposition 3. Let $(\theta^*, \omega^*, \xi^*)$ be an optimal solution to problem (13)-(18). If multiple optima exist, let it be one that minimizes $\Sigma(\theta, \omega, \xi) = \sum_{l \in \Omega} \theta_l + \sum_{s \in \mathcal{S}} \omega_s +$

$\sum_{u \in N, \sigma \in \Lambda_u} \xi_{u\sigma}$. The proof follows by applying the same arguments provided for the proofs of correctness for the S-DOI and F-DOI in sequence. First we assume that $\omega_s^* > 0$ for some $s \in \mathcal{S}$ to arrive to a contradiction. Then, assuming that $\omega_s^* = 0$ for every $s \in \mathcal{S}$, we assume that $\xi_{u\sigma}^* > 0$ for some $u \in N, \sigma \in \Lambda_u$ to arrive to another contradiction. □

Proof of Lemma 1. Let $\alpha^* \geq 0$ be such that $c_l - \sum_{u \in N} a_{ul}\alpha_u^* \geq 0$ for every $l \in \Omega$, which implies that α^* is feasible for the dual of (1)-(3), therefore $\sum_{u \in N} \alpha_u^* \leq z^*$. Because (α^*, γ^*) is feasible for the dual RMP of (13)-(18) when the constraint set is restricted to Ω_R, it follows that $\sum_{u \in N} \alpha_u^* \geq z^*$. Therefore $\sum_{u \in N} \alpha_u^* \leq z^* \leq \sum_{u \in N} \alpha_u^*$. □

Proof of Proposition 4. When one ignores the γ variables in the pricing process, two possible outcomes for the pricing are possible: i) We identify a column $l \in \Omega$ such that $c_l - \sum_{u \in N} a_{ul}\alpha_u^* < 0$. Because the γ-variables are ≤ 0, it follows that $\overline{c_l} < 0$ when computed using equation (19). Therefore, the set Ω_R can be enlarged by adding column l; ii) For every column $l \in \Omega$, $c_l - \sum_{u \in N} a_{ul}\alpha_u^* \geq 0$. In this case, we apply the result of Lemma 1 and stop the column generation. We have computed a valid dual bound for problem (1)-(3). □

Proof of Proposition 5. The first point is trivially true as swapping a customer u covered by a route by an uncovered customer v of lower demand is always possible. Let now $s = (u, v)$ be such that $u, v \in N, u \neq v, d_u \geq d_v$. Let $l \in \Omega(u, v)$ be a column containing u but not v. Let i, j be the predecessor and successor nodes of u in the route l. If nodes u, v were swapped and the ordering of the nodes not changed, the change in cost of this operation would be $\Delta = c_{iv} + c_{vj} - c_{iu} - c_{uj}$. This means that the cost $c_{l'}$ of route $l' = swap(l, s)$ is such that $c_{l'} \leq c_l + \Delta \leq c_l + \rho_{uv}$. It follows that $c_{l'} - c_l \leq \rho_{uv}$ for every route $l \in \Omega(u, v)$. □

Proof of Proposition 6. Let $l \in \Omega_R$, and let $S \subseteq N(l)$ be a subset of nodes covered by l. Let $(s_1, \ldots, s_n)$ be an arbitrary permutation of the elements of S, and let $S_i = \{s_k : k = 1 \ldots i\}$ be the subset of the first i elements in that permutation. Let $l'_i = remove(l, S_i)$ and $l'_0 = l$. Note that $c_l - c_{l'} = \sum_{i \in \{0, \ldots, n-1\}} c_{l'_i} - c_{l'_{i+1}}$. Note that by (31), $\sigma_{s_i l} \leq c_{l'_{i-1}} - c_{l'_i}$. Therefore $\sum_{u \in S} \sigma_{ul} \leq c_l - c_{l'}$. □

2 Computational performance and analysis on the CpMP and the CVRP

2.1 CpMP

In this appendix we present the computational study of our DOI on the CpMP described in Section 7.1. We test on the same datasets used in Section 8.2 where we set the facility count, p, for each problem to the facility count of the solution of the linear relaxation of the equivalent SSCFLP problem rounded up to the nearest integer. We also set each facility's opening cost to zero. For the Holmberg et al. and Yang et al. datasets we run the same capacity adjustment experiments described in Section 8.2.3. The algorithms have been coded in Matlab and we use CPLEX as our general-purpose MIP solver. Our machine is equipped with an 8-core AMD Ryzen 1700 CPU @3.0 GHz and 32 GB of memory running Windows 10. We notice that on some CpMP problems, unstabilized CG can take prohibitively long to converge. For these problems, we set the maximum iteration count of CG to 5000 iterations. If by the 5000th iteration, the unstabilized CG has not yet converged, we exit. We use the time elapsed up to that point as the convergence time but add a "+" as an indicator that the true convergence time is longer. We also add a "+" when doing associated speedup calculations. All other implementation parameters are equivalent to those described in Section 8.2.

2.1.1 Results on the Holmberg et al. and Yang et al. datasets

Numerical results for both datasets are shown in Table 1. On the Holmberg et al. dataset, we see that `sdoi` delivers the highest average and median speedup over all capacity levels, save for the average speedup at $L = 3$, where `sfdoi` provides the highest speedup. At higher capacity levels, `sfsdoi` generally provides the second highest speedup with average speedups of 12.7 and 14.9 at $L = 3$ and $L = 4$ respectively. On the Yang et al. dataset, `smsdoi` provides the fastest speedup at $L = 1$ and $L = 2$. When $L \geq 3$, `sfsdoi` provides the fasest speedup, with a median speedup of 32.7 at $L = 4$. Across both datasets that `fdoi` do not generally perform well at lower capacity levels, however they perform significantly well at higher capacity levels.

2.1.2 Results on newly generated random instances

CpMP results on the structured and unstructured synthetic datasets are shown in Tables 4 and 6 respectively. We see again that `sdoi` perform very well on the structured dataset

Instance			Time (sec) std	Speedup sdoi	fdoi	sfdoi	sm	smsdoi
Holmberg et al.	mean	$L = 1$	20.9	3.5	1.0	1.6	1.8	2.1
		$L = 2$	48.6	6.9	3.1	5.4	2.9	5.0
		$L = 3$	92.7	11.9	6.2	12.7	4.3	10.3
		$L = 4$	121.0	14.9	8.1	14.9	5.8	10.9
	median	$L = 1$	21.5	4.0	1.1	1.6	1.6	2.3
		$L = 2$	32.6	6.4	2.7	3.9	2.5	4.0
		$L = 3$	68.9	11.3	4.4	6.3	3.9	8.4
		$L = 4$	98.5	12.5	5.3	10.7	5.8	8.2
Yang et al.	mean	$L = 1$	38.0	1.5	0.2	0.3	1.7	1.1
		$L = 2$	249.2	2.6	1.7	3.0	3.8	5.7
		$L = 3$	1190.7	3.5	11.1	37.4	10.4	24.2
		$L = 4$	3682.1+	5.5+	59.5+	103.6+	10.4+	57.6+
	median	$L = 1$	32.7	1.6	0.2	0.3	1.6	0.9
		$L = 2$	138.1	2.0	1.4	2.0	1.7	2.7
		$L = 3$	523.5	3.3	3.4	7.3	4.2	7.1
		$L = 4$	847.8	3.5	16.0	32.7	4.5	9.0

Table 1: CpMP runtime results for increased capacity. New capacity $K'_i = LK_i$ for each facility

std	sdoi	fdoi	sfdoi	sm	smsdoi
0	**15**	**0**	**0**	**1**	**0**

Table 2: CpMP # of instances with lowest runtime, Holmberg et al. dataset ($|N| = 200$)

std	sdoi	fdoi	sfdoi	sm	smsdoi
0	**3**	**0**	**0**	**6**	**1**

Table 3: CpMP # of instances with lowest runtime, Yang et al. dataset ($|N| = 200$)

but still do not provide improvement on the unstructured dataset. In fact, only `fdoi` manage to provide a positive average speedup on the unstructured dataset. `sfdoi` has the highest average speedup on the structured dataset with a lower bound on the average speedup of 393.8. `sdoi` has the highest median speedup on the structured dataset with a median speedup of 132.5.

	Time (sec)	**Speedup**				
	`std`	`sdoi`	`fdoi`	`sfdoi`	`sm`	`smsdoi`
mean	8239.5+	329.2+	17.2+	393.8+	19.4+	291.8+
median	2212.3	132.5	5.9	69.5	10.1	78.8

Table 4: CpMP average runtime over 50 structured problem instances.

`std`	`sdoi`	`fdoi`	`sfdoi`	`sm`	`smsdoi`
0	**31**	**0**	**12**	**0**	**7**

Table 5: CpMP # of instances with lowest runtime, 50 structured problem instances

	Time (sec)	**Speedup**				
	`std`	`sdoi`	`fdoi`	`sfdoi`	`sm`	`smsdoi`
mean	52.7	0.7	1.2	0.8	3.0	1.0
median	50.3	0.7	1.3	0.8	3.0	1.0

Table 6: CpMP average runtime over 50 unstructured problem instances.

`std`	`sdoi`	`fdoi`	`sfdoi`	`sm`	`smsdoi`
1	**0**	**8**	**0**	**40**	**1**

Table 7: CpMP # of instances with lowest runtime, 50 unstructured problem instances

2.2 CVRP

In this section, we assess the stabilization capabilities of the S-DOI, F-DOI and SF-DOI on the CVRP. To this end, our analysis considers instances from two datasets:

1. A subset of classical instances from the CVRP literature, namely from the sets E, M, P and X (Uchoa et al., 2017). We restrict our analysis to instances with 200 customers or less and such that the ratio between the number of customers and the number of vehicles is ≥ 10. Note that the instances from the sets E, M and P assume that the number of available vehicles is limited, whereas for the instances from the set X the fleet us assumed to be of unlimited size.

2. A new dataset (denoted the H dataset) of 8 highly degenerate instances constructed as follows. The number of nodes $|N|$ is picked in $\{30, 40, 50, 60\}$. The position of each node is picked at random inside a square of dimensions 1000×1000. The demands are integers picked at random in the interval $[10, 100]$. The minimum number of vehicles k is either 2 or 3. However, we follow the convention established by the instances of the set X and assume that the fleet is of unlimited size. if D represents the sum of the customers' demands, the vehicle capacity for a given k is equal to $Q = \lfloor D/(k-1) \rfloor - 1$. The capacity Q is made in such a way that $k-1$ vehicles are short of one unit of capacity each to serve the entire demand.

Both datasets are made available in the website http://claudio.contardo.org/datasets-source-code/, including a Julia script to construct instances of the second dataset.

The CG algorithm has been coded in C++ and uses the IBM CPLEX 12.10 solver for the solution of the linear programs. Our CG considers several state-of-the-art acceleration techniques for the pricing subproblem, including two labeling heuristics and one metaheuristic (Contardo et al., 2020), bidirectional labeling (Righini and Salani, 2006), completion bounds (Baldacci et al., 2008; Contardo and Martinelli, 2014), and `ng`-routes embedded within a decremental state-space relaxation (DSSR) mechanism (Baldacci et al., 2011; Martinelli et al., 2014). We price on elementary routes only, but use DSSR to keep the computational burden under control. The MP is enhanced by the addition of rounded capacity cuts (Fukasawa et al., 2006). We initialize the RMP by executing a metaheuristic to construct near-optimal primal solutions. Every column from every such solution found by the metaheuristic is then used to initialize the RMP.

Differently from what we presented for the two previous problems, our implementation of the F-DOI considers fixed sets $\Lambda_u^r = \Lambda^r = \{0, 5, 25, 125, 625, 3125\}$ of size six. Every time that a column enters the RMP, the value of σ_{ul} is computed and truncated down from the values in Λ_r. Our implementation of the S-DOI is also different from that used for the facility location problems. We consider a S-DOI for every possible tuple $(u, v), u, v \in N, u \neq v, d_u \geq d_v$.

2.2.1 Results for the first dataset

For the instances from the sets E, P, M and X we solve the linear relaxation of problem (27)-(29) with the addition of capacity cuts and report, in Tables 8, 9 and 10, the total elapsed CPU time (in seconds), the number of CG iterations, and the number of columns generated. We highlight in bold characters the setting that achieved the lowest value for each measure. Note that the results for instance M-n121-k7 are not reported as all six settings had to be aborted after exceeding a time limit of one day. When looking at the CPU times, the results are not conclusive. The `std` setting is often the fastest, and the different stabilization schemes seem to provide little to no performance boost. We believe that this is mainly due to two factors:

- The fact that the pricing heuristics do not seem much affected by the choice of the stabilization, and regardless of the total number of CG iterations, the number of calls to the exact pricing routine remains almost constant throughout the different settings. An example of such behavior is instance M-n101-k10, where `sdoi` converges in about a 40% less CG iterations than `std`, but using a larger computing time.
- The little to no degeneracy of those instances. It is known that degeneracy in vehicle routing becomes a problem when excessively long routes are priced (≥ 25 customers per route) (see, for instance Costa et al., 2019). None of the instances from this dataset satisfy this criterion, which is also why we generated the second dataset, with much longer routes.

The stabilization impact of the different settings, but in particular of `smsdoi` is better seen when one looks at the number of columns generated, which are several times less than those generated by the `std` setting.

2.2.2 Results for the second dataset

We have executed the same experiment for the second dataset, but now we have given the algorithm a maximum computing time of 86,400 seconds (one day), before the optimization is aborted and the problem considered as unsolved. In Tables 11-13 we report the same data as for the previous tables, and also highlight in bold characters the minimum for each row.

Now, we can see that the differences in CPU time between the `std` setting and the different stabilization schemes are important. The reductions in terms of number of iterations are of up to a factor of 10 (instance H-n60-k2). Although there is no one setting that outperforms the others, it can be seen that the dual optimal inequalities are very robust and provide substantial gains with respect to `std` or even `sm`. When put together, we see that `sdoi`, `fdoi` and `sfdoi` are best (in terms of CPU time) in all cases but one. However, we do not fully understand yet the problem characteristics that favors one stabilization scheme over another. This is an important subject that deserves further digging in a future research.

Instance	$\lvert N \rvert$	k	`std`	`sm`	`sdoi`	`fdoi`	`sfdoi`	`smsdoi`
E-n76-k7	76	7	**60.44**	121.69	68.93	81.10	81.02	83.18
E-n101-k8	101	8	**174.56**	218.78	193.09	318.32	319.44	227.03
M-n101-k10	101	10	**192.85**	245.49	209.57	243.06	270.26	311.56
M-n151-k12	151	12	796.95	1170.36	**759.06**	872.54	766.36	962.22
M-n200-k16	200	16	**3720.56**	3746.90	4543.44	5475.67	4148.73	5483.90
M-n200-k17	200	17	2698.52	3776.41	2608.94	**2574.36**	4636.46	4297.49
P-n76-k4	76	4	183.71	228.37	200.01	207.78	**181.82**	253.12
P-n76-k5	76	5	120.94	149.21	**105.23**	111.43	109.43	181.45
P-n101-k4	101	4	**1218.94**	1612.26	1258.48	2198.87	1829.53	1723.62
X-n120-k6	120	6	1452.89	1008.98	**957.24**	1923.64	1248.14	1091.00
X-n139-k10	139	10	**680.33**	833.67	695.39	980.30	771.51	936.19
X-n143-k7	143	7	**14446.83**	23475.95	18949.61	16418.12	17412.14	21676.15
X-n157-k13	157	13	710.17	**636.54**	693.26	1143.55	1041.41	822.42
X-n162-k11	162	11	5694.35	5973.84	**5515.11**	7513.89	6593.95	5802.92
X-n167-k10	167	10	3987.99	3756.02	4269.62	**3645.37**	3935.80	3856.73
X-n186-k15	186	15	7359.41	8688.01	**7102.83**	7612.79	8416.60	8215.15
X-n190-k8	190	8	**19756.26**	23594.75	21540.90	24211.85	24029.39	25009.03

Table 8: CPU time spent on some classic CVRP instances

Instance	$\|N\|$	k	`std`	`sm`	`sdoi`	`fdoi`	`sfdoi`	`smsdoi`
E-n76-k7	76	7	**97**	128	112	136	121	108
E-n101-k8	101	8	**149**	159	180	294	303	167
M-n101-k10	101	10	766	711	**415**	463	488	463
M-n151-k12	151	12	**217**	275	228	250	251	241
M-n200-k16	200	16	673	765	684	**543**	674	817
M-n200-k17	200	17	226	236	225	252	287	**166**
P-n76-k4	76	4	173	177	191	187	163	**162**
P-n76-k5	76	5	125	101	**87**	116	111	128
P-n101-k4	101	4	512	712	**423**	684	459	751
X-n120-k6	120	6	1841	1031	1137	1503	1386	**865**
X-n139-k10	139	10	365	355	**316**	571	404	428
X-n143-k7	143	7	**816**	1219	855	1198	983	957
X-n157-k13	157	13	653	**515**	688	857	753	567
X-n162-k11	162	11	513	475	**465**	903	802	473
X-n167-k10	167	10	634	611	**582**	822	745	650
X-n186-k15	186	15	**263**	394	239	368	411	336
X-n190-k8	190	8	**2341**	2708	3312	3032	2922	2617

Table 9: CG iterations on some classic CVRP instances

Instance	$\|N\|$	k	`std`	`sm`	`sdoi`	`fdoi`	`sfdoi`	`smsdoi`
E-n76-k7	76	7	2380	**1139**	2253	1841	1604	1147
E-n101-k8	101	8	3649	**2231**	3801	3165	3225	2506
M-n101-k10	101	10	6270	3510	2714	**1926**	2207	2056
M-n151-k12	151	12	5268	3941	4859	3881	3611	**2732**
M-n200-k16	200	16	13975	10203	10398	14182	9094	**5555**
M-n200-k17	200	17	6711	4349	5506	5016	4643	**2642**
P-n76-k4	76	4	4786	3303	4327	4107	3410	**2974**
P-n76-k5	76	5	3277	1037	2652	2504	2544	**924**
P-n101-k4	101	4	18586	16352	14007	17217	**9816**	18143
X-n120-k6	120	6	14887	4686	5733	7081	5271	**2650**
X-n139-k10	139	10	7070	3192	5069	3950	3608	**2284**
X-n143-k7	143	7	22189	22604	23963	21315	**17114**	19506
X-n157-k13	157	13	5122	2076	3840	3125	2498	**1661**
X-n162-k11	162	11	9789	5405	7897	7788	6563	**4611**
X-n167-k10	167	10	11022	**6141**	10134	9289	7594	6274
X-n186-k15	186	15	6300	4220	5603	4276	4757	**3319**
X-n190-k8	190	8	42359	34043	47598	30314	31985	**26999**

Table 10: Total number of columns generated on some classic CVRP instances

Instance	$\|N\|$	k	`std`	`sm`	`sdoi`	`fdoi`	`sfdoi`	`smsdoi`
H-n30-k2	30	2	116.81	124.44	122.50	96.96	**46.38**	141.05
H-n30-k3	30	3	48.64	94.66	**31.18**	49.32	44.87	81.66
H-n40-k2	40	2	**341.42**	653.93	429.62	418.93	616.23	620.67
H-n40-k3	40	3	185.16	207.42	122.06	151.10	**98.62**	382.00
H-n50-k2	50	2	1610.59	1013.99	470.00	582.34	**366.70**	700.75
H-n50-k3	50	3	2410.23	2140.66	2563.37	**840.19**	896.54	1740.95
H-n60-k2	60	2	40392.25	–	90062.19	**8308.26**	20891.27	–
H-n60-k3	60	3	574.26	719.66	**485.25**	515.99	608.61	725.37

Table 11: CPU time spent on newly generated instances

Instance	$\|N\|$	k	`std`	`sm`	`sdoi`	`fdoi`	`sfdoi`	`smsdoi`
H-n30-k2	30	2	64	171	75	57	**25**	104
H-n30-k3	30	3	140	129	85	112	98	**78**
H-n40-k2	40	2	112	383	227	**48**	105	269
H-n40-k3	40	3	260	163	185	200	**97**	263
H-n50-k2	50	2	530	334	240	185	**155**	293
H-n50-k3	50	3	1448	770	1369	396	**267**	694
H-n60-k2	60	2	1865	–	1250	199	**184**	–
H-n60-k3	60	3	446	459	416	353	404	**286**

Table 12: CG iterations on newly generated instances

Instance	$\|N\|$	k	`std`	`sm`	`sdoi`	`fdoi`	`sfdoi`	`smsdoi`
H-n30-k2	30	2	1944	4626	1667	748	**379**	1266
H-n30-k3	30	3	1412	619	875	455	**336**	423
H-n40-k2	40	2	4219	7797	4184	**1391**	2059	4004
H-n40-k3	40	3	3287	1390	1905	1976	**620**	1021
H-n50-k2	50	2	12960	7298	4696	3827	3604	**3274**
H-n50-k3	50	3	21990	9119	19976	5808	**3489**	6929
H-n60-k2	60	2	27758	–	19499	6104	**3395**	–
H-n60-k3	60	3	8082	10687	11284	7047	6547	**4535**

Table 13: Total number of columns generated on newly generated instances

3 Detailed results

In this Section we provide detailed results for the two location problems considered in our study, namely the single-source capacitated facility location problem (SSCFLP) and the capacitated p-median problem (CpMP).

Inst.	lb	Total Runtime (sec)						Iterations						Number of Columns					
		std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
56	20972.1	5.1	3.0	9.2	6.5	2.2	3.7	182	54	57	36	155	105	3707	1351	1257	970	1814	1001
57	25819.7	5.1	3.7	16.7	15.5	1.8	3.9	169	64	75	53	119	99	3701	1458	1656	1187	1724	1043
58	36564.6	6.0	4.8	36.2	29.9	1.7	4.6	175	67	106	64	113	103	3789	1673	2360	1500	1720	1110
59	26973.8	8.1	6.1	31.5	21.4	5.1	10.5	210	103	112	62	303	260	4162	1847	2156	1374	2172	1316
60	20534.0	17.4	3.6	7.1	6.5	4.0	3.3	310	69	54	54	246	103	6022	1444	1033	763	1916	786
61	24454.0	27.9	4.9	12.2	6.2	7.5	6.4	398	84	81	42	315	186	7509	1830	1320	855	2437	1074
62	32287.6	15.5	6.1	31.4	30.2	3.5	5.2	267	83	105	63	173	121	5092	1691	2033	1344	2017	985
63	25084.1	36.2	6.4	15.0	8.5	6.1	5.1	462	118	87	61	308	151	7656	1716	1418	847	2171	892
64	20530.0	27.2	4.0	6.7	6.5	4.4	2.5	373	77	49	58	246	79	7356	1441	979	665	1993	684
65	24445.0	31.6	5.7	9.2	6.6	7.2	3.8	388	94	69	48	332	114	7715	1817	1051	712	2322	786
66	31264.2	26.7	6.8	22.6	25.9	3.3	4.9	330	91	95	71	158	110	6382	1651	1686	1192	2022	916
67	24803.1	38.4	41.4	16.4	10.8	10.4	11.7	489	428	85	62	586	269	7577	5951	1493	1093	2579	1680
68	20538.0	17.7	4.2	7.4	3.2	4.7	3.4	299	85	54	22	286	106	6065	1598	1076	629	2045	852
69	24532.0	28.1	5.3	10.8	9.2	8.8	5.3	383	91	64	69	365	156	7687	1855	1299	935	2431	1008
70	32230.1	34.7	8.9	49.8	21.8	6.3	6.9	427	139	154	86	291	171	7235	2109	2290	1206	2279	1002
71	25535.1	20.7	6.1	19.7	12.8	5.1	6.0	346	102	94	56	280	159	5958	1808	1691	1066	2087	929

Table 14: SSCFLP results, Holmberg et al. dataset ($|N| = 200$)

Inst.	lb	Total Runtime (sec)						Iterations						Number of Columns					
		std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
1	29835.8	38.6	23.9	212.2	140.8	13.5	18.7	500	257	356	242	567	384	6759	3937	5669	3596	3150	2241
2	28582.5	28.4	22.1	202.7	141.2	8.9	13.9	363	250	307	211	402	275	6645	4129	5727	3730	3040	2144
3	27946.3	38.2	22.8	210.4	131.8	7.4	12.8	477	248	347	219	375	257	6818	3632	5582	3272	2691	1825
4	27670.5	54.0	35.7	274.2	143.1	13.1	16.6	595	379	383	222	513	348	8422	4672	6458	3558	3051	1942
5	27586.5	106.0	67.8	580.3	291.2	38.8	21.2	1070	698	854	589	977	439	10003	5796	7872	4631	3876	2067
6	27855.9	34.6	21.4	207.6	139.9	12.4	11.3	353	199	275	168	228	172	7812	4983	6645	4703	3800	3018
7	29450.4	27.4	19.5	175.4	134.4	12.1	10.2	290	186	247	152	224	156	7048	4760	6249	4426	3742	2943
8	27754.5	41.2	27.0	218.6	158.9	18.5	19.0	398	247	281	178	320	286	8661	5659	6797	4879	3938	3282
9	27476.1	28.9	19.6	184.5	137.3	10.5	8.5	307	179	260	149	239	133	7069	4483	6151	4071	3367	2435
10	28988.5	36.4	25.8	207.6	140.9	23.4	13.9	408	275	285	179	441	218	7514	5079	6568	4528	3954	3125

Table 15: SSCFLP results, Yang et al. dataset ($|N| = 200$)

Inst.	lb	Total Runtime (sec)						Iterations						Number of Columns					
		std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
1	68.4	910.0	27.7	190.2	60.2	42.7	12.0	1567	197	260	132	387	174	27726	3276	3768	1780	4297	1166
2	69.7	5121.3	14.2	293.5	42.6	68.8	8.6	3300	158	405	145	478	148	40777	1839	4341	1291	4450	930
3	68.6	7652.9	36.0	179.3	29.9	96.8	10.8	3799	237	276	79	642	184	56168	4110	3985	1325	5290	1150
4	70.1	523.4	18.6	457.1	56.2	61.3	11.8	1154	161	344	151	522	187	21236	2473	5474	1681	4538	1192
5	69.3	1069.5	24.7	129.5	37.6	65.4	9.7	1512	191	225	93	548	155	27828	3251	3593	1533	4797	1134
6	69.3	743.2	12.9	115.3	23.8	37.4	6.7	1492	127	217	77	343	103	25182	1978	3380	1098	3952	979
7	68.7	15400.0	12.3	292.1	18.4	72.7	5.0	4991	161	285	77	653	103	84034	1768	4103	917	5083	772
8	69.4	645.9	29.2	369.2	82.0	77.1	20.2	1196	201	310	139	710	320	22194	3532	4962	2129	4984	1546
9	70.0	689.1	27.3	161.0	53.2	48.3	9.6	1156	234	242	138	439	155	22351	3249	3820	1702	4231	1070
10	69.9	475.5	14.8	97.3	28.7	36.9	7.5	1008	157	196	88	311	110	18999	1993	2825	1124	3549	982
11	68.9	288.7	16.2	285.6	75.0	32.8	9.0	882	142	259	128	333	119	16261	2304	4167	1957	3928	1127
12	67.7	3606.8	48.9	248.8	59.8	113.8	18.1	2232	300	265	114	745	297	41675	4865	4591	1649	5456	1416
13	69.0	300.6	20.1	131.6	51.4	23.8	7.9	839	198	244	151	249	109	15534	2356	2957	1522	3475	971
14	71.2	408.9	27.8	161.2	65.8	33.8	10.3	1028	199	245	135	313	133	17878	2826	3414	1835	3692	1027
15	69.3	459.0	14.9	227.6	37.7	40.7	9.1	1130	142	275	88	339	140	19398	2087	3906	1393	3883	990
16	70.8	235.0	13.1	93.9	43.9	30.7	7.6	768	125	163	115	270	109	14251	1956	2767	1481	3645	1022
17	68.5	3424.2	58.6	407.5	55.5	88.6	15.3	2727	332	255	123	465	247	40871	5032	4492	1619	4436	1326
18	70.1	264.8	15.8	91.0	42.8	32.7	9.9	895	138	197	110	352	144	16751	2231	2913	1507	3821	1078
19	68.9	400.2	25.4	123.9	62.0	32.4	10.5	1054	176	177	125	326	148	18637	2939	3126	1753	3995	1223
20	69.7	550.8	23.8	156.7	40.6	46.8	12.9	1229	188	243	108	509	203	21560	3090	3509	1518	4251	1242
21	69.9	823.9	40.9	245.5	65.0	68.0	11.1	1798	312	301	157	542	164	25458	3821	4050	1896	4535	1174
22	71.5	142.9	17.6	82.4	43.6	18.3	9.2	644	139	150	91	238	120	12134	2377	2656	1598	3456	1201
23	68.7	423.4	14.0	64.4	19.6	49.8	6.9	1004	117	159	61	444	118	19178	2114	2584	907	4083	916
24	68.2	2602.4	12.4	114.5	30.3	78.6	11.1	2275	125	181	90	491	179	42203	2009	3139	1265	4566	1129
25	71.4	1637.5	13.8	82.9	23.9	67.1	7.6	1851	147	187	95	556	137	32344	1992	2771	1114	4720	941

Table 16: SSCFLP results structured problem instances 1-25.

Inst.	lb	Total Runtime (sec)						Iterations						Number of Columns					
		std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
26	68.7	7726.4	21.7	243.6	10.8	117.3	9.3	3581	183	313	44	1041	195	66893	3334	4240	666	6536	1000
27	69.3	8311.1	18.4	153.2	41.9	99.9	9.6	4193	183	285	131	711	169	61711	2375	3744	1440	5180	1047
28	70.2	128.5	14.8	76.0	46.7	15.3	8.6	613	138	160	106	215	110	11800	2040	2626	1564	3315	1143
29	67.7	3105.1	36.8	361.5	18.2	101.1	9.9	2597	268	268	67	815	188	38682	3777	4236	956	5495	1129
30	68.7	212.7	20.5	185.0	68.5	24.7	9.3	850	156	235	126	278	118	15055	2437	3876	1908	3615	1025
31	67.9	1016.3	19.6	276.7	28.0	58.8	11.0	1519	189	326	97	531	209	26770	2484	4669	1227	4471	1064
32	70.2	433.4	19.4	118.8	58.0	38.9	9.4	1094	196	235	162	340	140	20438	2338	3173	1605	3877	1035
33	69.1	2404.7	33.2	90.0	22.8	76.5	8.6	2149	227	184	77	431	146	36636	3537	3047	1074	4275	983
34	68.9	926.0	36.0	183.4	47.5	80.3	15.4	1403	232	190	117	636	284	26051	4301	3299	1434	5015	1298
35	68.9	5642.5	23.6	200.6	45.4	77.1	11.8	3726	266	374	155	547	199	50456	2528	3943	1370	4695	1085
36	68.0	260.1	18.9	109.8	52.5	23.1	8.7	1111	163	216	120	270	119	15586	2192	2915	1605	3505	912
37	69.0	370.4	17.7	204.3	56.9	31.9	8.7	1064	196	283	155	298	142	18989	2009	3797	1562	3756	989
38	68.4	779.3	33.0	341.9	58.0	75.4	11.7	1574	246	367	190	668	199	21781	3447	4740	1624	4818	1134
39	69.5	223.8	22.3	196.5	67.8	39.1	14.3	906	166	249	123	397	207	15189	2764	4027	1902	3957	1238
40	69.1	453.6	25.9	185.1	65.1	39.8	14.5	1289	211	296	156	348	224	20940	2889	4058	1896	4154	1104
41	67.9	318.9	22.0	183.7	72.6	30.6	8.7	1269	163	282	171	317	124	16084	2749	3817	1993	3623	1017
42	70.3	1065.8	26.9	217.8	20.0	59.9	8.6	1485	199	236	60	526	152	28862	3290	3566	974	4600	959
43	70.4	188.1	17.6	151.8	53.4	23.9	9.5	756	135	184	92	265	110	13902	2462	3367	1738	3610	1194
44	69.7	520.0	28.1	339.5	64.7	41.3	10.1	1178	194	291	119	361	142	21342	3154	4723	1980	3923	1125
45	70.0	156.2	16.5	85.2	56.1	27.7	12.5	678	131	149	95	288	169	13224	2412	2761	1737	3773	1293
46	67.7	53021.6	14.6	80.2	13.0	83.1	8.5	8137	143	179	46	548	170	155872	2577	2918	725	5260	975
47	69.8	215.0	23.5	267.9	80.4	32.3	10.9	710	160	215	117	259	158	13842	2773	3775	2069	3818	1191
48	69.3	373.0	12.1	74.3	25.8	36.7	7.1	1022	120	170	74	304	99	16443	1780	2444	1167	3776	1010
49	69.2	448.7	26.8	376.9	96.5	32.6	11.5	1453	220	368	173	323	135	18449	2865	4554	2225	3889	1358
50	68.4	400.3	16.9	154.4	51.1	45.9	10.5	923	163	233	135	383	154	17347	2209	3477	1609	4078	1094

Table 17: SSCFLP results structured problem instances 26-50.

Inst.	lb	Total Runtime (sec)						Iterations						Number of Columns					
		std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
1	50.1	59.4	69.7	42.9	77.6	8.0	21.1	357	330	99	95	161	150	6952	6562	1836	1896	2611	2398
2	49.1	59.2	66.0	41.4	68.6	8.5	21.7	351	322	91	88	167	159	7093	6515	1886	1819	2655	2447
3	49.7	60.6	63.9	38.4	70.2	8.6	21.4	357	310	90	91	170	155	7188	6308	1825	1808	2594	2371
4	50.1	53.6	60.8	44.9	57.0	8.2	20.8	333	305	95	87	160	151	6754	6232	2013	1900	2506	2406
5	50.5	60.1	63.4	40.9	67.6	7.9	21.1	353	312	92	88	161	156	7204	6289	1926	1868	2530	2416
6	51.1	60.7	67.7	36.4	66.1	7.9	21.2	363	325	83	90	170	158	7339	6588	1794	1827	2526	2378
7	50.0	58.9	65.8	48.9	82.9	8.3	20.7	354	317	102	106	165	148	7036	6416	1988	2031	2600	2375
8	50.1	74.7	67.9	42.4	71.6	8.2	20.9	366	333	97	98	164	158	7283	6548	1904	1944	2484	2365
9	50.2	64.5	69.6	44.5	73.7	8.0	20.8	375	335	97	98	157	152	7436	6669	1897	1930	2505	2406
10	48.6	61.8	70.6	35.2	62.7	8.0	20.8	356	328	82	83	162	145	7193	6677	1733	1768	2473	2384
11	50.3	55.6	62.5	39.4	70.6	8.3	20.7	340	311	87	90	163	156	6891	6267	1881	1912	2593	2374
12	50.4	62.2	64.4	43.4	75.6	8.3	23.2	375	325	100	106	168	175	7301	6291	1872	1938	2505	2505
13	50.2	60.2	72.3	33.3	52.7	7.8	19.5	353	337	77	74	159	138	7194	6808	1608	1572	2471	2315
14	50.9	59.1	68.1	34.5	52.0	8.0	21.1	355	328	88	83	159	162	7058	6511	1648	1663	2555	2409
15	51.0	61.7	69.4	40.4	63.6	7.9	21.2	362	333	97	91	158	159	7235	6625	1785	1755	2418	2340
16	50.6	58.1	67.8	39.2	63.8	8.4	22.2	354	338	96	95	170	169	6958	6544	1762	1739	2505	2398
17	50.3	55.9	62.2	44.2	73.4	8.8	21.3	340	309	95	92	167	150	6915	6288	1994	1991	2727	2429
18	50.1	55.8	62.2	40.7	70.4	8.2	21.5	343	310	85	94	159	157	6845	6287	1886	1930	2565	2460
19	51.1	59.1	65.9	33.9	57.3	8.1	21.2	354	319	85	84	152	154	7019	6342	1647	1684	2514	2381
20	50.1	52.6	57.1	50.7	78.0	7.9	20.8	335	298	104	98	160	155	6676	5973	2091	2029	2503	2349
21	51.3	58.2	66.5	47.4	71.9	8.3	22.0	357	332	105	96	169	161	7068	6440	1931	1886	2572	2423
22	50.4	56.9	63.2	43.6	73.1	8.3	20.4	344	307	93	90	163	149	7053	6335	1953	1919	2577	2408
23	50.0	58.4	67.2	41.2	69.6	8.1	21.0	349	326	91	95	160	151	7042	6441	1835	1885	2456	2407
24	49.2	57.5	62.5	40.8	73.7	8.5	22.0	351	317	92	92	166	158	7048	6276	1857	1917	2619	2477
25	51.2	59.9	68.0	35.2	58.3	8.0	21.4	355	327	85	79	161	154	7130	6559	1714	1628	2496	2415

Table 18: SSCFLP results unstructured problem instances 1-25.

Inst.	lb	Total Runtime (sec)						Iterations						Number of Columns					
		std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
26	49.9	60.4	67.3	40.7	68.5	7.8	21.0	358	322	89	89	156	151	7205	6567	1886	1858	2456	2427
27	50.5	56.4	62.0	52.2	89.8	8.0	22.1	344	308	107	108	166	158	6879	6205	2106	2143	2587	2462
28	49.7	59.9	69.5	42.8	71.2	8.0	21.2	354	326	92	91	158	151	7193	6616	1924	1935	2554	2441
29	50.2	60.3	68.2	47.3	71.5	8.3	19.9	365	330	101	96	154	147	7167	6573	1938	1933	2609	2341
30	50.0	60.5	64.7	35.5	55.5	7.9	20.7	356	318	83	76	160	151	7175	6412	1732	1639	2504	2423
31	50.8	62.5	69.7	31.7	46.6	7.9	20.5	366	330	81	76	165	159	7239	6586	1578	1528	2472	2308
32	50.6	61.4	66.5	36.2	61.6	7.9	20.2	358	322	82	82	160	150	7267	6517	1749	1740	2496	2352
33	49.6	58.1	66.5	43.3	69.2	7.6	21.0	353	321	96	90	160	158	7101	6554	1884	1869	2459	2416
34	50.4	52.8	64.4	48.2	78.7	7.7	21.1	334	323	100	101	153	156	6738	6399	1988	1963	2448	2371
35	51.1	54.2	59.6	42.5	73.9	7.7	21.1	338	302	92	93	158	156	6868	6181	1974	1943	2560	2427
36	51.6	60.5	63.3	36.4	61.7	8.3	21.8	356	311	85	80	162	155	7254	6315	1774	1742	2548	2392
37	49.8	58.1	63.8	38.9	55.3	8.0	20.5	352	312	89	87	157	150	7123	6329	1775	1799	2541	2389
38	50.8	57.5	66.3	52.2	93.3	8.4	21.7	358	328	107	115	172	161	6942	6536	2124	2154	2736	2441
39	50.8	58.0	64.6	44.9	64.3	8.1	21.3	359	321	99	97	168	155	7119	6370	1968	1989	2573	2413
40	51.1	61.3	66.3	39.1	69.3	8.0	20.8	363	321	91	94	162	155	7222	6405	1808	1818	2454	2342
41	50.3	59.7	63.7	32.1	56.6	8.2	21.1	351	308	73	75	166	156	7157	6286	1609	1634	2572	2404
42	50.6	56.8	62.0	35.1	61.2	8.2	21.4	344	305	82	81	169	161	6935	6174	1710	1718	2542	2396
43	51.1	57.1	63.8	38.1	62.9	8.1	20.7	343	312	88	83	166	159	7040	6393	1826	1794	2561	2368
44	50.9	58.9	67.9	39.1	66.7	8.7	21.7	351	330	92	90	169	156	7076	6512	1720	1806	2721	2428
45	50.5	58.7	62.1	46.0	77.9	8.2	21.8	353	313	96	98	165	162	7068	6247	1975	1992	2599	2441
46	50.2	60.0	64.6	32.3	49.8	8.1	21.1	355	315	80	82	170	162	7125	6332	1615	1638	2509	2383
47	49.9	59.6	64.8	48.6	68.6	8.0	21.3	366	321	109	105	160	157	7099	6359	1946	1963	2541	2426
48	51.0	59.1	64.5	35.9	65.3	8.3	22.2	352	313	83	89	170	160	7017	6260	1724	1773	2579	2452
49	50.9	56.9	63.6	42.7	56.9	8.1	21.8	349	317	95	94	165	162	7011	6287	1965	1928	2535	2402
50	50.4	58.6	66.3	38.1	60.5	7.6	20.4	354	319	90	84	157	152	7065	6442	1797	1723	2447	2324

Table 19: SSCFLP results unstructured problem instances 26-50.

Inst.	lb	Total Runtime (sec)						Iterations						Number of Columns					
		std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
56	11992.6	4.6	3.7	16.5	10.2	4.3	6.1	171	57	68	49	227	146	3535	1358	1460	1030	1912	1062
57	13067.8	7.0	3.7	56.4	23.2	5.0	7.1	187	57	124	47	239	154	3966	1457	2582	1188	2208	1191
58	13067.8	6.9	3.8	52.3	21.1	4.6	7.1	187	57	124	47	239	154	3966	1457	2582	1188	2208	1191
59	13067.8	6.9	3.7	50.7	20.8	4.8	7.0	187	57	124	47	239	154	3966	1457	2582	1188	2208	1191
60	13034.0	15.8	3.8	13.2	10.7	8.4	7.4	278	72	87	76	430	200	5533	1465	1171	935	2286	1008
61	14854.0	27.5	7.0	25.0	13.3	16.8	11.1	369	115	97	66	577	276	7096	2169	1736	1113	3069	1295
62	15943.2	20.8	5.2	48.1	22.9	11.1	9.7	304	80	125	64	456	211	5679	1566	2225	1194	2777	1189
63	14854.0	27.5	7.0	25.1	13.4	16.9	11.2	369	115	97	66	577	276	7096	2169	1736	1113	3069	1295
64	13030.0	21.3	3.8	9.2	8.9	8.8	5.8	321	70	58	62	486	154	6385	1450	1006	809	2289	803
65	14845.0	32.2	5.6	13.2	8.5	17.6	10.5	375	90	69	40	723	275	7527	1797	1254	771	3107	1265
66	17930.7	28.3	7.0	29.2	25.2	8.8	11.1	369	103	108	84	417	246	6312	1677	1736	1168	2486	1162
67	14619.0	29.4	27.7	29.3	24.0	18.6	19.8	372	238	99	83	633	397	7169	4770	1846	1355	3197	2166
68	13038.0	15.0	4.5	12.7	7.7	10.6	6.3	272	86	73	46	534	171	5293	1630	1241	803	2613	990
69	14932.0	27.3	6.0	21.5	13.6	13.7	11.1	338	97	83	80	523	273	6883	1990	1579	1030	2714	1317
70	17258.5	42.7	9.6	56.6	25.9	20.6	18.5	481	150	148	73	672	397	7636	2049	2526	1239	3309	1504
71	14167.0	21.6	4.7	13.3	7.1	15.6	9.3	319	81	66	39	700	246	6373	1732	1336	821	3003	1313

Table 20: CpMP results, Holmberg et al. dataset ($|N| = 200$)

		Total Runtime (sec)						Iterations						Number of Columns					
Inst.	**lb**	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
1	5963.4	35.5	21.3	201.1	159.1	21.0	37.2	392	208	273	187	685	652	6828	3802	4918	3298	3544	3328
2	5361.6	22.5	17.0	119.9	107.5	11.0	14.6	276	161	185	118	499	282	5743	3523	4013	2877	3062	2285
3	5097.4	43.9	24.7	172.9	153.4	12.5	25.7	496	250	288	199	541	478	6680	3550	4230	2879	3081	2484
4	5582.7	32.7	21.1	141.3	101.1	10.3	34.9	374	188	239	143	414	624	6577	3481	4357	2669	3106	3130
5	6810.6	90.8	69.1	462.2	298.5	71.8	58.9	720	531	551	352	1260	959	9595	6188	7286	4121	5285	3758
6	5039.4	32.7	20.4	128.5	106.7	19.6	40.8	306	178	182	139	372	559	7164	4871	4723	3831	4069	4340
7	5335.3	29.8	21.1	203.1	174.2	68.9	51.5	274	184	223	142	884	626	6786	4701	5856	4236	6080	4277
8	4855.4	34.8	21.7	168.3	119.5	33.1	54.9	312	185	196	133	542	698	7576	5029	5445	4055	4461	4615
9	4709.2	28.0	19.3	93.2	79.8	19.0	27.0	313	206	151	117	394	395	6509	4289	4040	3327	3847	3437
10	4507.2	29.7	18.2	182.5	128.1	32.0	37.3	285	172	186	122	492	522	6798	4543	5278	4091	4601	4027

Table 21: CpMP results, Yang et al. dataset ($|N| = 200$)

Inst.	lb	Total Runtime (sec)						Iterations						Number of Columns					
		std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
1	39.5	1084.6	14.8	192.1	38.0	87.4	24.8	1054	122	243	84	928	391	20285	2203	3763	1428	4790	1766
2	39.7	29907.3	48.2	884.5	26.2	3033.4	38.9	5000	274	439	75	7973	672	98239	5294	5005	1024	27699	2327
3	39.5	1503.4	15.2	262.0	28.1	217.6	20.8	1531	138	427	90	1674	375	27249	2086	3766	1070	7641	1516
4	40.3	9427.9	22.8	1044.4	29.0	462.6	35.8	3075	181	365	73	2486	556	56292	2985	4906	1136	10570	2018
5	40.1	758.6	11.2	120.6	35.9	91.0	22.0	1140	99	173	71	1073	320	21992	1840	3044	1288	5507	1758
6	39.4	6817.7	15.2	402.4	25.2	1089.5	19.7	2684	132	346	81	4745	379	48680	2251	4190	1090	18120	1541
7	38.7	29941.4	30.0	663.7	23.5	1103.2	20.7	5000	213	384	82	5369	398	99416	3843	4310	886	20488	1757
8	40.1	19280.1	43.7	321.0	18.0	294.0	26.1	4397	245	197	47	2123	497	87164	4768	3225	935	9657	2228
9	40.1	28076.5	36.5	532.6	27.2	1423.4	41.1	5000	230	282	74	4471	637	97296	4067	4124	1196	17687	2361
10	40.8	406.9	10.8	69.3	32.0	118.6	26.6	927	103	153	85	1114	390	18005	1593	2365	1121	5266	1511
11	39.4	597.7	36.7	324.9	48.9	168.4	28.6	998	209	230	85	1478	433	19590	3982	3744	1540	7100	2121
12	42.7	32913.9	47.0	6929.0	96.2	9905.1	40.7	5000	269	684	116	12700	516	89163	4373	12127	1745	39769	2427
13	44.1	394.1	18.1	243.3	56.1	41.2	16.2	800	163	315	123	488	216	15131	2146	3957	1532	3957	1324
14	46.2	514.3	23.5	253.1	82.8	224.1	42.8	989	201	293	151	1194	510	17603	2366	4220	1776	6593	1985
15	39.9	4199.1	27.9	485.3	27.3	187.5	26.1	2242	182	267	72	1614	456	44380	3521	3887	938	7708	2026
16	40.9	1062.1	22.7	460.8	66.5	430.8	38.9	1313	158	261	123	2223	474	22422	2653	3653	1795	9088	2109
17	39.6	859.7	14.2	88.6	44.0	103.2	41.8	1208	111	154	85	997	529	20664	2072	2608	1433	4895	2034
18	40.6	552.0	16.5	115.9	43.4	132.8	31.9	1017	182	247	127	1265	478	19494	2020	2762	1220	5950	1663
19	39.7	709.1	27.5	265.2	54.1	145.9	34.4	981	152	184	91	1209	520	18968	3013	3268	1565	6278	1970
20	39.8	7095.6	18.9	408.0	17.2	1507.3	21.3	2835	136	264	47	6593	326	51016	2733	3862	903	20584	1618
21	40.2	2379.4	14.9	613.2	36.0	150.4	23.8	2108	137	582	90	1535	343	30843	2291	5705	1339	6332	1524
22	41.7	561.1	27.2	202.4	61.5	103.8	28.1	964	172	254	122	1061	403	18281	3193	3787	1748	5780	1885
23	38.9	6829.2	11.6	187.8	22.7	345.3	18.3	2712	124	341	81	1838	290	44449	1722	3292	893	8324	1231
24	38.7	23792.9	31.7	993.4	14.5	830.8	45.8	5000	191	487	49	3912	699	100046	3797	5972	813	16259	2446
25	41.4	28618.2	29.5	758.9	21.3	353.4	21.2	5000	203	363	84	2258	345	97538	3500	4347	960	9996	1376

Table 22: CpMP results structured problem instances 1-25.

		Total Runtime (sec)						Iterations						Number of Columns					
Inst.	**lb**	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
26	38.7	28343.8	16.5	583.9	22.8	789.5	30.6	5000	136	297	64	3563	501	99760	2637	3487	952	15154	1904
27	39.3	4054.3	19.9	188	55.6	190.7	26.2	2409	203	260	122	1546	397	38235	2507	3944	1509	6861	1575
28	40.5	362.9	29.3	149.1	43.8	85.5	23.9	838	203	220	91	861	352	14369	3180	3095	1404	5178	1621
29	38.4	17804.2	17	300.2	15.6	152.3	18.7	4371	149	336	49	1386	303	79873	2524	3707	869	7388	1628
30	43.7	405.3	23.9	588.2	103.4	260.2	45.1	845	159	366	150	1445	471	14951	2499	6442	2094	8518	2029
31	38.7	2000.9	13.4	98.5	19.9	132.2	31.3	1597	144	155	45	1306	422	31131	2025	2590	883	6647	1788
32	40.7	832.7	16.2	144.6	55.8	95.2	22.8	1411	139	244	112	976	310	22717	2175	3317	1432	5262	1445
33	39.8	14031.5	38.6	394.8	14.3	178.5	28.4	3806	217	264	39	1317	463	74220	3951	3779	817	6816	1893
34	39.1	11372.8	33.6	474.4	14.2	627.3	31.7	3393	204	519	48	3509	501	59829	3947	4215	864	14208	2007
35	39.2	26085.3	40.4	940	15.8	804.7	41.4	5000	247	372	48	3738	656	96355	4630	4611	898	14602	2323
36	38.8	1682.9	18.3	120.4	35	145.8	40.9	1972	152	192	74	1433	498	27317	2427	3157	1279	5749	1802
37	40	344.5	14.2	141.9	41.4	46.9	16.6	832	99	163	73	540	213	15435	2004	2995	1305	3780	1268
38	38.8	4860.7	29.3	822.3	27.3	360.9	53.2	2423	191	470	77	2079	650	44121	3700	4994	1065	9297	2367
39	39.9	676	32.1	421.1	97.3	490.8	54.8	957	186	290	152	2265	619	17949	3136	4635	2180	10231	2337
40	39.8	2903.5	23.3	537.9	36.1	206.9	41.5	2136	204	456	88	1473	545	33054	3025	5018	1224	7263	2042
41	38.9	573	19.4	259.5	45.9	344.6	21.5	980	128	283	95	1730	314	17951	2445	4442	1369	8252	1537
42	40.6	5820	17.5	269.4	29.8	230	30.1	2609	141	184	67	1424	433	50747	2570	2837	1022	6814	1762
43	40.7	593.5	31.8	317.9	90.9	126.7	40.2	975	173	215	130	1027	439	18222	3344	3521	1834	5337	2124
44	40.2	16057.4	45	303.1	17.1	768	28	3988	247	283	45	2721	437	78181	5034	4245	854	12245	1938
45	40.4	270.6	13.6	82.5	42.8	96.3	22.3	751	114	159	91	955	295	13897	1961	2565	1383	5134	1477
46	38.6	28802.3	11.3	345.7	11.9	306.6	19.6	5000	103	150	40	2241	319	100107	1955	2728	654	10592	1464
47	40.4	757.7	24	149.2	48	122.7	30.9	869	172	196	94	1328	418	17121	3063	3217	1574	5473	1737
48	39.4	2438.4	29.4	525.4	26.3	126.6	28.6	1647	218	388	86	1151	440	32187	3570	4296	1139	6103	1610
49	40.4	573.6	31.7	672.4	86.9	99	40.5	1088	212	535	175	978	487	19019	3752	5853	2139	5030	2096
50	39.1	2045.2	14.6	356	31.2	197.7	23.6	1583	125	189	84	1382	355	29938	2119	3274	1233	7330	1850

Table 23: CpMP results structured problem instances 26-50.

		Total Runtime (sec)						**Iterations**						**Number of Columns**					
Inst.	**lb**	`std`	`sdoi`	`fdoi`	`sfdoi`	`sm`	`smsdoi`	`std`	`sdoi`	`fdoi`	`sfdoi`	`sm`	`smsdoi`	`std`	`sdoi`	`fdoi`	`sfdoi`	`sm`	`smsdoi`
1	21.0	51.9	79.8	42.2	65.4	9.8	40.9	316	299	79	75	217	241	6227	5816	1573	1486	2560	2355
2	24.1	88.3	104.4	650.2	168.4	36.8	67.5	394	311	191	136	366	400	7998	6274	3970	2872	3489	3107
3	24.8	76.1	108.2	114.3	126.2	173.4	134.8	365	316	131	110	990	703	7304	6360	2651	2230	6171	4694
4	21.0	51.6	78.4	36.9	64.3	17.1	55.8	311	285	69	72	314	345	6326	5849	1522	1514	3152	2542
5	21.6	53.3	81.9	37.7	62.0	11.6	70.8	313	290	67	70	276	434	6413	5925	1490	1491	2616	3051
6	21.9	50.7	79.2	38.5	64.8	9.7	46.1	299	284	67	68	247	312	6135	5833	1474	1458	2422	2371
7	21.1	47.4	75.6	42.3	67.2	47.2	63.9	305	289	83	84	644	418	5896	5672	1550	1540	3772	2896
8	21.1	50.4	83.5	44.7	73.3	18.3	79.4	311	313	91	96	462	530	6120	5996	1520	1505	2969	3141
9	21.4	49.9	76.0	37.9	61.7	68.0	119.7	305	289	75	74	782	763	6117	5731	1512	1472	4370	4383
10	23.6	82.3	117.6	102.1	107.8	21.4	47.9	369	318	113	98	360	303	7537	6448	2350	1998	3187	2637
11	21.1	51.4	76.4	40.8	64.4	34.3	93.9	309	289	70	70	540	629	6205	5837	1533	1509	3553	3834
12	21.6	45.6	74.5	36.1	61.9	38.3	70.7	288	283	64	64	584	482	5880	5741	1410	1400	3679	3008
13	21.3	45.4	70.4	32.2	57.7	9.9	32.6	289	273	59	65	219	193	5849	5596	1306	1330	2378	2195
14	21.9	52.4	77.3	37.1	66.8	13.5	34.8	320	287	77	83	346	214	6238	5724	1452	1468	2714	2276
15	22.0	52.1	80.7	38.2	61.5	40.5	61.6	315	313	74	72	641	410	6290	6039	1447	1408	3648	2860
16	21.5	50.5	81.0	35.6	61.1	24.6	67.1	311	295	71	63	510	475	6211	5984	1395	1369	3196	3007
17	21.2	50.3	78.9	42.8	66.8	56.5	41.0	315	289	74	75	657	254	6268	5835	1544	1485	4061	2504
18	20.9	50.1	79.3	41.0	66.6	12.4	36.9	302	285	73	68	279	224	6189	5820	1581	1507	2860	2378
19	26.1	71.7	110.3	76.9	105.8	48.5	58.8	354	307	105	95	462	353	7128	6226	2129	1909	3748	2848
20	20.8	52.5	78.0	48.3	75.8	10.2	35.8	308	284	74	71	224	217	6304	5702	1615	1544	2702	2316
21	22.3	49.0	74.1	35.3	57.5	16.1	37.1	307	287	71	75	404	216	6076	5744	1464	1416	2976	2403
22	21.5	49.4	74.8	40.1	64.9	10.4	38.3	304	288	72	71	241	234	6207	5799	1486	1477	2654	2702
23	21.1	50.2	77.5	37.8	64.6	12.0	43.7	311	295	71	77	306	312	6189	5778	1420	1443	2578	2508
24	20.2	52.7	76.0	39.0	64.6	99.3	119.4	311	288	73	75	874	864	6145	5767	1480	1417	5286	4627
25	22.2	50.6	73.9	35.3	55.6	12.4	38.0	312	285	76	72	277	253	6085	5765	1453	1427	2786	2154

Table 24: CpMP results unstructured problem instances 1-25.

Inst.	lb	Total Runtime (sec)						Iterations						Number of Columns					
		std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi	std	sdoi	fdoi	sfdoi	sm	smsdoi
26	21.0	49.2	75.8	36.6	62.7	50.9	69.7	301	282	65	69	637	469	6060	5804	1479	1498	4058	3207
27	21.4	50.2	75.5	44.3	72.8	8.8	34.0	299	278	76	72	203	202	6047	5643	1590	1552	2381	2267
28	20.8	51.6	78.6	44.6	65.3	14.0	61.2	318	299	81	82	331	425	6214	5867	1510	1481	2898	2871
29	21.3	45.9	75.0	41.7	63.9	13.6	64.5	294	281	79	73	342	467	5889	5752	1539	1472	2709	2945
30	21.0	47.8	74.7	35.5	63.0	24.2	79.8	300	287	67	63	506	552	6076	5711	1402	1369	3214	3461
31	25.8	70.9	116.6	87.8	116.0	31.5	60.1	355	318	100	84	414	368	7123	6410	2024	1787	3557	2928
32	21.6	45.9	76.8	36.6	60.3	15.4	71.8	293	287	70	63	397	478	5935	5803	1452	1405	2818	2993
33	20.6	49.8	76.3	36.3	60.2	16.8	41.2	300	276	67	64	438	290	6154	5694	1484	1444	2912	2278
34	21.4	50.0	74.7	40.1	60.6	11.0	42.3	309	282	75	77	242	262	6112	5715	1517	1480	2599	2554
35	21.8	48.8	72.1	43.6	63.4	18.4	47.5	293	281	73	67	418	333	6042	5777	1597	1526	3079	2614
36	22.6	50.0	73.6	35.7	56.6	12.4	70.9	309	297	74	80	267	481	6101	5951	1446	1416	2700	3149
37	20.8	47.3	71.2	36.6	55.0	14.4	42.6	296	282	68	67	356	269	6052	5754	1475	1442	2799	2683
38	21.7	55.7	76.2	60.8	87.4	9.8	35.7	339	303	104	105	244	208	6482	6014	1762	1709	2306	2412
39	21.7	51.9	74.4	42.1	63.5	63.9	96.9	315	298	79	74	753	623	6337	5925	1602	1518	4448	4286
40	22.1	52.7	72.0	33.5	51.8	60.5	82.7	304	279	65	60	696	583	6179	5672	1407	1346	4183	3467
41	21.2	45.3	74.1	32.2	54.2	10.0	37.7	293	291	70	65	222	223	5885	5821	1405	1420	2444	2389
42	21.6	46.1	71.6	33.1	54.6	21.6	49.2	291	281	66	62	453	325	5938	5727	1428	1368	3139	2572
43	21.9	52.1	72.5	39.5	59.5	10.7	38.9	306	286	70	68	244	238	6185	5880	1510	1512	2679	2493
44	22.0	48.3	73.3	32.3	53.8	11.1	55.5	294	293	63	63	280	402	5969	5876	1415	1391	2575	2508
45	21.4	53.7	77.4	39.2	60.5	16.3	51.2	310	295	68	65	353	349	6250	5918	1480	1411	3052	2657
46	21.2	49.5	72.6	31.2	51.7	14.5	54.0	300	283	64	61	388	363	6081	5758	1374	1345	2820	2857
47	20.9	48.4	71.6	40.0	58.6	15.9	56.4	297	284	79	67	374	387	6079	5732	1559	1495	3023	2808
48	21.8	53.3	75.0	35.5	52.2	27.6	47.1	324	305	74	69	508	288	6329	5848	1418	1384	3356	2730
49	21.9	45.8	74.3	41.2	53.9	10.1	32.9	288	291	95	72	207	197	5837	5743	1549	1520	2546	2396
50	21.4	48.8	70.5	36.1	57.1	32.0	58.7	307	278	74	73	557	409	6038	5532	1383	1382	3408	2832

Table 25: CpMP results unstructured problem instances 26-50.

References

Baldacci R, Christofides N, Mingozzi A (2008) An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming* 115(2):351–385.

Baldacci R, Mingozzi A, Roberti R (2011) New route relaxation and pricing strategies for the vehicle routing problem. *Operations research* 59(5):1269–1283.

Contardo C, Martinelli R (2014) A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints. *Discrete Optimization* 12:129–146.

Contardo C, Martinelli R, Vidal T (2020) Efficient metaheuristic pricing for vehicle routing In progress.

Costa L, Contardo C, Desaulniers G (2019) Exact branch-price-and-cut algorithms for vehicle routing. *Transportation Science* Forthcoming.

Fukasawa R, Longo H, Lysgaard J, de Aragão MP, Reis M, Uchoa E, Werneck RF (2006) Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical programming* 106(3):491–511.

Holmberg K, Rönnqvist M, Yuan D (1999) An exact algorithm for the capacitated facility location problems with single sourcing. *European Journal of Operational Research* 113(3):544–559.

Martinelli R, Pecin D, Poggi M (2014) Efficient elementary and restricted non-elementary route pricing. *European Journal of Operational Research* 239(1):102–111.

Righini G, Salani M (2006) Symmetry helps: Bounded bi-directional dynamic programming for the elementary shortest path problem with resource constraints. *Discrete Optimization* 3(3):255–273.

Uchoa E, Pecin D, Pessoa A, Poggi M, Vidal T, Subramanian A (2017) New benchmark instances for the Capacitated Vehicle Routing Problem. *European Journal of Operational Research* 257(3):845–858, URL `http://dx.doi.org/10.1016/j.ejor.2016.08.01`.

Yang Z, Chu F, Chen H (2012) A cut-and-solve based algorithm for the single-source capacitated facility location problem. *European Journal of Operational Research* 221(3):521–532.