



DALARNA
UNIVERSITY

**Working papers in transport, tourism, information technology and microdata
analysis**

Confidence in heuristic solutions?

**Författare 1: Kenneth Carling
Författare 2: Xiangli Meng
Editor: Hasan Fleyeh**

Nr: 2014:12

Working papers in transport, tourism, information technology and microdata analysis

ISSN: 1650-5581

© Authors

Confidence in heuristic solutions?

Authors[♦]: Kenneth Carling and Xiangli Meng^{*}

Abstract: Solutions to combinatorial optimization problems frequently rely on heuristics to minimize an objective function. The optimum is sought iteratively and pre-setting the number of iterations dominates in operations research applications, which implies that the quality of the solution cannot be ascertained. Deterministic bounds offer a mean of ascertaining the quality, but such bounds are available for only a limited number of heuristics and the length of the interval may be difficult to control in an application. A small, almost dormant, branch of the literature suggests using statistical principles to derive statistical bounds for the optimum. We discuss alternative approaches to derive statistical bounds. We also assess their performance by testing them on 40 test p -median problems on facility location, taken from Beasley's OR-library, for which the optimum is known. We consider three popular heuristics for solving such location problems; simulated annealing, vertex substitution, and Lagrangian relaxation where only the last offers deterministic bounds. Moreover, we illustrate statistical bounds in the location of 71 regional delivery points of the Swedish Post. We find statistical bounds reliable and much more efficient than deterministic bounds provided that the heuristic solutions are sampled close to the optimum. Statistical bounds are also found computationally affordable.

Key words: p -median problem, deterministic bounds, statistical bounds, jackknife, discrete optimization, extreme value theory

[♦] Kenneth Carling is a professor in Statistics and Xiangli Meng is a PhD-student in Micro-data analysis at the School of Technology and Business Studies, Dalarna university, SE-791 88 Falun, Sweden.

^{*} Corresponding author. E-mail: xme@du.se. Phone: +46-23-778509.

1. Introduction

Consider the challenge of finding a solution to the minimum of an intractable objective function of a combinatorial optimization problem. This challenge arises frequently in operations research (OR) dealing with issues like automation, facility location, routing, and scheduling. Usually a good, but biased, solution (called heuristic solution) is sought by applying one of the many iterative (heuristic) methods available. None of the heuristics guarantees that the solution will coincide with the optimum and, hence, many solutions to real world OR-problems are plagued with an uncertainty about the quality of the solution.

It goes without saying that many heuristic users find this uncertainty unappealing and try to take measures to control it. There are four measures for doing so. The first is to carefully choose the heuristic amongst them thoroughly tested on various problems of similar kind, the second is to set the number of iterations large or run until no improvements are encountered, the third is to use methods that provide deterministic bounds, and the fourth is to compute statistical bounds. We assume that the first measure (as well as reasonable data-reduction of the problem) is taken by the user and we will not discuss it further. The second measure is common practice, whereas the third measure occasionally is taken. Deterministic bounds can be informative on the quality of the solution, but often at a high computational cost. The fourth measure is rarely taken by heuristic users in spite of its potential powerfulness. One reason for heuristic users hesitating in taking the fourth measure is the lack of systematic investigation of statistical bounds. The aim of this paper is to systematically investigate statistical bounds for the fundamental location-allocation problem being the p -median problem. In the investigation we vary heuristic method, sample size, estimator of the bounds, the number of iterations (computing time), and the complexity of the problem. We present results on the reliability of the statistical bounds and the length of the interval compared with the length of deterministic bounds. The computer experiment is accompanied with an illustrating problem concerning the location of regional delivery points of the Swedish Post.

This paper is organized as follows: section two presents statistical optimum estimation techniques. In the third section we present the p -median problem and discuss three solution

methods used in the investigation. In section four details about the experiments are given concerning the implementation of the solution methods, factors varied in the experiment and factors kept constant, as well as outcome variables of interest. The fifth section gives the results and the sixth section illustrates the use of statistical bounds in a practical problem of locating 71 delivery points of the Swedish Post in a region. The seventh section concludes the paper.

2. Statistical optimum estimation techniques

First some notations used throughout the paper: z_k = the k :th feasible solution to the combinatorial problem; A = the set of all feasible solutions $A = \{z_1, z_2, \dots, z_K\}$; $g(z_k)$ = the objective function of solution z_k ; $\theta = \min_A g(z_k)$. The challenge is to identify θ and the corresponding solution z_k . The abbreviation SOET (statistical optimum estimation techniques) was recently introduced by Giddings, Rardin, and Uzoy (2014) and it refers to techniques both for point and interval estimating θ .

In short, the statistical approach is to estimate the value of the minimum based on a sample of heuristic solutions and put bounds (or confidence limits) around it. Golden and Alt (1979) did pioneering work on statistical bounds followed by others in the 1980:s, but thereafter the statistical approach has been dormant. In fact, Akyüz, Öncan, and Altinel (2012) state that the statistical approach had not been used in location problems since 1993 to the best of their knowledge.

There are four approaches to estimating statistically the optimum according to the review article of Giddings et al (2014). The dominating one is based on Extreme Value Theory (EVT) and another is the truncation-point approach which we will refer to as the Weibull (W) estimator and the Jackknife (JK) estimator, respectively. Giddings et al (2014) also mention the Limiting-distribution approaches as well as the Multinomial approaches. Neither of them is used or seems promising and we will disregard them.

As a point of departure in discussing the SOET:s, we note that even a very large random sample of $g(z_p)$ is almost useless for identifying θ . The crux is that a sample of the

gigantic size of 10^9 will be hard pressed to contain a feasible solution close to θ even for combinatorial optimization problems of modest complexity (Meng and Carling, 2014). Fortunately, as several authors have pointed out, if the starting values are picked at random, repeated heuristic solutions mimic a random sample in the tail (see e.g. McRoberts, 1971 and Golden and Alt, 1979). Thereby random values in the tail can be obtained at a much less computational effort and used for estimating θ . To discuss estimation of θ and statistical bounds for the parameter we need additional notations: $\hat{\theta}$ = an estimator of θ ; n = the number of replicates of the heuristic algorithm with random starting values; \tilde{x}_i = the heuristic solution of i^{th} replication $i = 1, 2, \dots, n$.

The Weibull point estimator of θ is $\hat{\theta}_W = \tilde{x}_{(1)}$ where $\tilde{x}_{(1)}$ is the best (smallest) heuristic solution in all the n replicates.. The JK-estimator is introduced by Quenouille (1956):

$$\hat{\theta}_{JK} = \sum_{i=1}^{M+1} (-1)^{(i-1)} \binom{M+1}{i} \tilde{x}_{(i)}$$

where M is the order. Dannenbring (1977) and Nydick and Weiss (1988) suggest to use the first order, i.e. $M = 1$, for point estimating the minimum. The first order JK-estimator is more biased than higher order ones, but its mean square error is lower compared with higher orders as shown by Robson and Whitlock (1964). Carling and Meng (2014) however consider both the first and the second order JK-estimator and note that the second order JK-estimator performs quite well. The JK point-estimators are $\hat{\theta}_{JK}^{(1)} = 2\tilde{x}_{(1)} - \tilde{x}_{(2)}$ and $\hat{\theta}_{JK}^{(2)} = 3\tilde{x}_{(1)} - 3\tilde{x}_{(2)} + \tilde{x}_{(3)}$ respectively.

As for interval estimation, the upper bounds of the Weibull and the Jackknife estimators are the same, $\tilde{x}_{(1)}$. However their lower bounds differ. The Weibull lower bound with a confidence of $100(1 - e^{-n})\%$ is $[\tilde{x}_{(1)} - \hat{b}]$ where \hat{b} is the estimated shape parameter of the Weibull distribution (Wilson, King and Wilson, 2004). There are several ways of estimating the shape parameter, including the maximum likelihood estimation technique. We as others have found the following simple estimator to be fast, stable, and giving good results:

$$\hat{b} = \tilde{x}_{[0.63(n+1)]} - (\tilde{x}_{(1)}\tilde{x}_{(n)} - \tilde{x}_{(2)}^2) / (\tilde{x}_{(1)} + \tilde{x}_{(n)} - 2\tilde{x}_{(2)})$$

integer value of the function (Derigs, 1985).

The Jackknife estimator, and extensions of it (Dannenbring, 1977), has only served the purpose of point estimating θ . However, Meng and Carling (2014) suggests a lower bound computed by means of bootstrapping the point estimator (Efron, 1979). The lower bound is $[\hat{\theta}_{JK} - 3\sigma^*(\hat{\theta}_{JK})]$ where $\sigma^*(\hat{\theta}_{JK})$ is the standard deviation of $\hat{\theta}_{JK}$ obtained from bootstrapping the n heuristic solutions (1,000 times was found sufficient). With the scalar 3, the confidence is 99.9% provided that the sampling distribution of the JK-estimator is approximately Normal.

Giddings et al (2014) provide a critique of the SOET:s which deserves to be re-stated briefly. Firstly, all the estimators presume a continuous distribution of the heuristic solutions for being theoretically justified. However, the discrete, feasible solutions amounts to a finite K and a strong heuristic method might produce a sample of clustered, discrete heuristic solutions. Thus, this assumption is strictly speaking false.¹ Secondly, clustering of heuristic solutions suggests a violation of the independence assumption upon which the probability statement of the interval hinges. Although several authors (see e.g. Wilson et al, 2004) have proposed goodness-of-fit tests for checking the assumed Weibull distribution and the assumed independence, one cannot expect the power of the tests to be high from a small sample in the extreme tail of the distribution. A failure of rejecting the null hypotheses of Weibull distribution and independence is perhaps a stronger indication of low power of the test than an indication of the correctness of the assumptions.

A conclusion from Giddings' et al (2014) critique is that theoretically perceived statistical properties of the SOET:s is possibly misleading and the properties' applicability is specific to the heuristic and the combinatorial problem due to the improper assumptions. As a consequence, the theoretical properties need to be empirically checked for various problems and heuristic methods.

¹ We believe less complex problems to be more amenable to discreteness and consequent improper statistical bounds, whereas highly complex problems have a large number of service and demand points rendering the parent distribution almost continuous. Exact solutions is usually feasible for non-complex problems, while deterministic or statistical bounds are critical for complex problems.

3. The p -median problem and heuristic methods

Location theory is an important part of operations research and it is concerned with the issue of locating facilities and assigning demand points to them in some optimal way. Continuous problems like the original Weber problem deals with location in the plane, whereas discrete problems deals with location on networks with vertices (or nodes) connected by edges. The p -median problem is the most important of the four primary discrete location problems with the other three being the p -center, the uncapacitated facility location, and the quadratic assignment problems. A nice virtue of operations research is the vast availability of test problems for which the optimum either is known or consistently is updated as improved solutions emerge. Consequently, the SOET:s can be checked with respect to known optimums. We focus on the p -median problem because of its fundamental role in location theory and the large number of existing test problems in the OR-library (Beasley, 1990).

The problem is to allocate P facilities to a demand geographically distributed in Q demand points such that the weighted average or total distance to its nearest service center is minimized for the demand points. Hakimi (1964) considered the task of locating telephone switching centers and showed later (Hakimi, 1965) that, in a network, the optimal solution of the p -median model exists at the nodes (vertices) of the network. If V is the number of nodes, then there are $K = \binom{V}{p}$ feasible solutions for a p -median problem where the largest test problem we consider in the computer experiment has about $2.5 * 10^{164}$ feasible solutions.

As enumerating all feasible solutions is not possible as the problem size grows, much research has been devoted to efficient methods to solve the p -median model (see Handler and Mirchandani, 1979 and Daskin, 1995 as examples). Reese (2006) reviews the literature for solution methods² to the p -median problem. Lagrangian relaxation (LR) is the most used method and since it gives deterministic bounds we will use it as a benchmark for SOET:s (Beasley, 1993).

² Reese (2006) uses the term solution methods to indicate any approach to find (approximately) the optimum. Heuristics and meta-heuristics are subsets of solutions methods in his sense; however we will use solution method and heuristic interchangeably in what follows.

In the class of heuristic methods, vertex substitution (VS) is the most frequent. VS starts with a (random) p^* , say, and seeks a local improvement by examining all local nodes for the first facility, then the second and so on until it reaches the last facility up on which an iteration is completed. After reaching the last facility, it returns to the first facility and repeats the procedure until no further improvement is encountered. The algorithm has no inherent randomness as it updates the solution according to a deterministic scheme. Randomness in the heuristic solution comes from either selecting the starting nodes, p^* , at random or by randomizing the order of examination of the nodes in p^* .

Another class (the largest) of solution methods to the p -median problem is metaheuristics with a domination of genetic algorithms³ and simulated annealing (SA). SA does *not* apply a deterministic scheme in its search for a good heuristic solution. Hence, randomness in the heuristic solution will come both from random starting points and inherent randomness in the algorithm. SA starts with p^* and selects at random one facility and evaluates a move of it to another node picked at random. The facility will be moved if it implies an improvement, but it may also be moved with a small probability in spite of no improvement. One random pick including the evaluation constitutes an iteration. SA runs until a pre-specified (usually large) number of iterations are met.

We limit the study to VS, SA, and LR. There are of course details regarding the implementation of the algorithms and we therefore implement all of them in the statistical language R and provide the code in the Appendix in the interest of making our study replicable.⁴ The virtue of SA is that the algorithm will gradually iterate towards the optimum since the inherent randomness implies that eventually all feasible solutions will be evaluated. Hence, by increasing the number of iterations better solutions are to be expected. VS and LR are more susceptible to be trapped in local minima and thereby never approaching the optimum however large the number of iterations. Such a fundamental difference in the characteristics of the methods may very well have implication on the functioning on the SOET:s.

³ See for instance Michalewicz and Janikow (1991).

⁴ (www.r-project.org).

4. The computational experiment

The first factor varied in the computational experiment is the estimator. The point and the interval Weibull-estimator are defined in section 2 except for the confidence level. The confidence level will be kept at $\alpha = 0.9987$ throughout the experiments which means that the lower bound will be calculated as $[\tilde{x}_{(1)} - \hat{b}/s]$, where $s = (-n/\ln\alpha)^{1/\hat{a}}$ and $\hat{a} = (\tilde{x}_{(1)}\tilde{x}_{(n)} - \tilde{x}_{(2)}^2)/(\tilde{x}_{(1)} + \tilde{x}_{(n)} - 2\tilde{x}_{(2)})$ being an estimator of the shape parameter of the Weibull distribution (Giddings et al, 2014). The first and the second Jackknife estimators are also defined in section 2 where $\sigma^*(\hat{\theta}_{JK})$ is calculated from 1,000 bootstrap samples of the n heuristic solutions.

The second factor is complexity, usually understood as the number of feasible solutions, of the p -median test problems. Carling and Meng (2014) shows that the normal distribution gives a close fit to $g(z_k)$ for most of the 40 test problems and computes complexity as the distance between the optimum and the estimated center of $g(z_k)$ in terms of standard deviations. With this definition of complexity the test problems ranges from 2.97 to 14.93.

The third factor is the heuristic methods varied. The implementation of them is explicitly given by the R-code provided in the Appendix. Our implementation of the solution methods follows closely Densham and Rushton (1992) (VS), Al-Khedhairi (2008) (SA), and Daskin (1995) (LR)⁵.

In the early literature on statistical bounds little was said on the size of n , whereas Akyüz et al (2012) and Luis, Sahli, and Nagy (2009) advocate n to be at least 100. However, Brandeau and Chiu (1993) as well as Carling and Meng (2014) find $n = 10$ to work well. We will examine as the fourth factor $n = 10, 25$. However, running the heuristics SA and VS is computationally costly and repeating them is of course even more costly. For this reason we run the algorithms 100 times per problem to obtain 100 heuristic solutions. Thereafter,

⁵ Han (2013) implemented LR for the test problems and, by pre-testing, found the mean of the columns of the distance matrix divided by eight to yield good starting values for the algorithm. In the implementation of the algorithm, we follow his approach.

heuristic solutions are sampled with replacement from the set of 100 solutions. LR is run only once per experimental combination as its initial values are deterministically determined by the distance matrix of the problem.

The fifth, and the last, factor is the running time of the solution methods. The time per iteration varies by the size and complexity of the problems and it therefore reasonable to assign more running time to the more difficult problems. We run the algorithms for $2 * \left(\frac{V}{100}\right)$, $20 * \left(\frac{V}{100}\right)$, $60 * \left(\frac{V}{100}\right)$ seconds per replicate where, again, V is the number of nodes of the problem. The nodes vary from 100 to 900 in the test problem so the running time of the algorithms varies between 2 second to 9 minutes. Running time refers to CPU time on Intel i5-2500 and 3.30GHz.

We present results on the reliability of the SOET:s by the bias ($\hat{\theta} - \theta$), the proportion of intervals containing the optimum (hereafter referred to as coverage), the length of the intervals, and the proportion of intervals shorter than the length of the deterministic bounds.

5. Results

Our computer experiment is a full factorial experiment of $3 \times 40 \times 2 \times 2 \times 3$ (estimator, complexity, heuristic, n , and running time) resulting in 1440 experimental combinations for which bias, coverage, length of interval, and the proportion of intervals shorter than the length of the deterministic bounds are outcome variables.⁶ We begin by checking for which experimental combinations it is meaningful to compute statistical bounds. Thereafter we check the bias of the three estimators in point estimating the optimum. Finally, we give results on the coverage of the intervals as well as their length.

5.1. Statistical bounds and the quality of heuristic solutions

Monroe and Sielken (1984) observe that the heuristic solutions need to be close to the optimum for the computing of statistical bounds to be meaningful. Carling and Meng (2014) suggest the statistic SR , given by the ratio $1000\sigma(\tilde{x}_i)/\hat{\theta}_{JK}^{(1)}$, as a measure of concordance and

⁶ The complete outcome matrix may be requested from the authors by contacting xme@du.se.

as a check of the heuristic solutions being in the tail near to θ . Figure 1 shows how coverage decreases with SR (for the some 75 per cent of treatment combinations with SR below 15). For SR above 25 the coverage is practically zero (not shown). Carling and Meng (2014) found SR equal to 4 to be an operational threshold in the case of simulated annealing, but the threshold also seems to apply to vertex substitution in ensuring statistical bounds to correspond reasonably well to the stipulated confidence level.

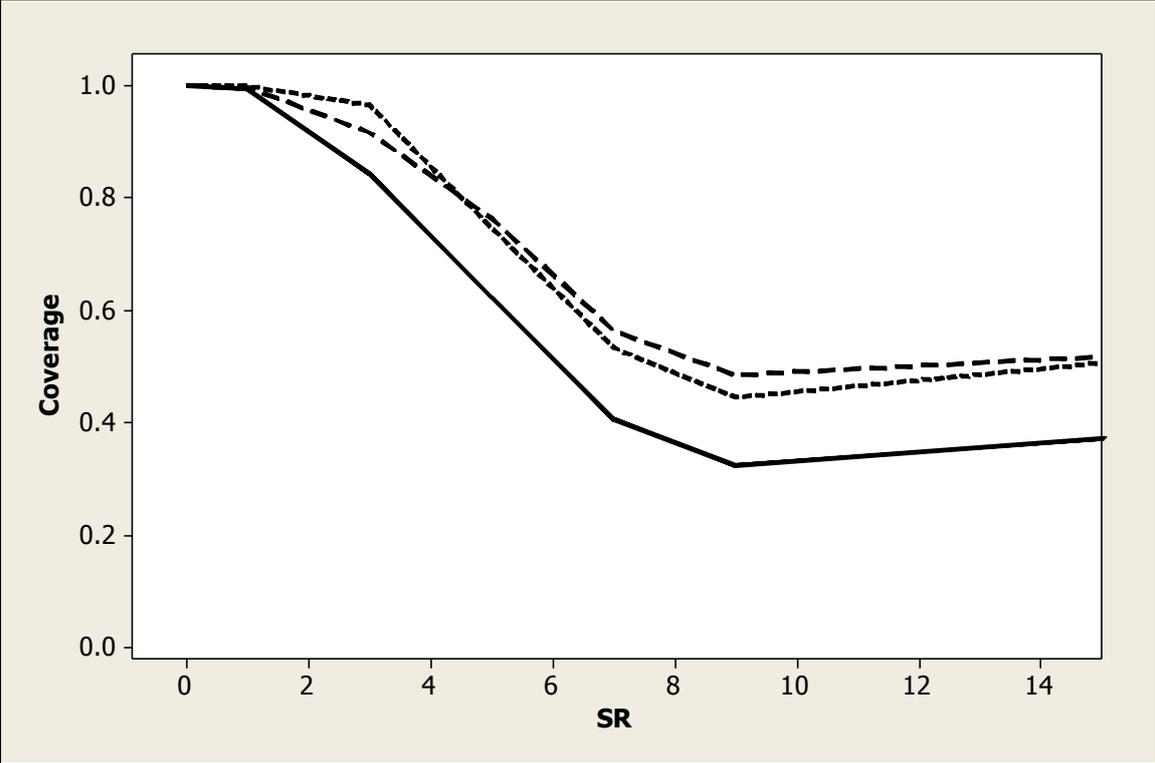


Figure 1: Coverage as a function of SR . Jackknife estimator is solid line (1st order) and dashed line (2nd order) and Weibull estimator is dotted line.

As a consequence of large SR implying improper statistical bounds, we will only examine the outcome of the treatment combinations for which $SR \leq 4$. There are 480 experimental combinations per estimator of which 247 are kept for analysis. Combinations, especially for vertex substitution, of short computing time and high complexity tend to fail the threshold check. Figure 2 illustrates the non-linear relationship with LOWESS curves between SR (in logarithms) and complexity for the three levels of computing time (Cleveland, 1979). Of the 247 combinations used in the analysis, 83 are vertex substitution (5, 38, and 40 with short, intermediate, and long computing time) and 164 are simulated annealing (35, 57, and 72 with short, intermediate, and long computing time) combinations. The partial failure of vertex

substitution to pass the threshold check is typically a consequence of the algorithm managing only a few iterations in the allotted computing time. For instance, the average number of iterations was 2.6 for the combinations of short computing time. We considered extending the allotted computing time in the experiment, but discarded this option as the allotted computing time generally was sufficient for both SA and LR.

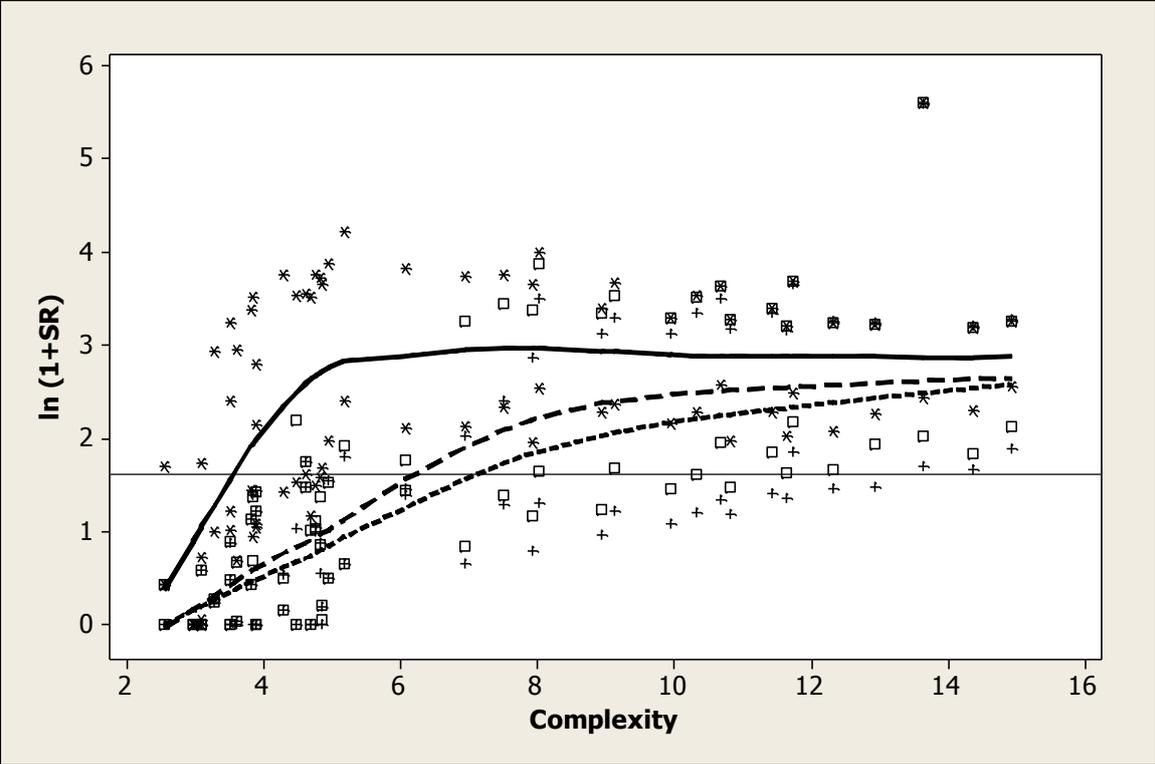


Figure 2: Experimental combinations removed from the analysis (SR above 4, solid reference line). Short computing time (solid line, asterisks), intermediate computing time (dashed line, squares), and long computing time (dotted line, pluses).

5.2. The bias of point-estimation

Before studying the properties of the statistical bounds, we first examine the estimators' ability in point-estimating the minimum. Table 1 gives the bias of the estimators focusing on the case $n = 25$. Theoretically the 2nd order Jackknife estimator has the smallest bias followed by the 1st order Jackknife estimator, while the Weibull point-estimate is simply the smallest heuristic solution and thereby always larger than (or possibly equal to) the optimum. In practice, the three estimators are comparable in terms of bias. For a given running time, solutions obtained by SA are much closer to the optimum compared with VS although solutions are consistently improving by time for both methods. LR, on the other hand, gets a

decent solution after a short running time which trivially improves with added running time. Looking specifically at the combinations for which $SR \leq 4$, the bias is petite as the heuristic solutions as clustered near to the optimum. There was no apparent difference in solutions obtained by SA and VS, and the outcomes of the methods are therefore pooled. The bias of LR is somewhat positive, particularly for the used combinations of long running time which contains a greater proportion of the more complex p -median problems. Although not shown in the table, the general conclusions regarding bias also apply to the case $n = 10$.

Table 1: Average relative bias (%) in the point estimation of θ . $n = 25$.

Method	Time	E-C ^a	JK 1 st	JK 2 nd	Weibull	LR ^c
SA	Short	40	1.7	1.6	1.8	
	Interm.	40	0.4	0.4	0.4	
	Long	40	0.1	0.1	0.2	
VS	Short	40	22.2	21.8	23.4	
	Interm.	40	16.0	15.7	16.7	
	Long	40	13.0	12.8	13.5	
LR	Short	40				3.4
	Interm.	40				3.3
	Long	40				3.3
Used combinations ^b	Short	19	-0.00	-0.00	0.01	0.31
	Interm.	47	0.02	0.02	0.03	1.18
	Long	56	0.04	0.04	0.05	1.74

Note: ^{a)} Experimental combinations. ^{b)} SA and VS pooled due to similarities. ^{c)} Results based on one replicate due to the deterministic outcome of the algorithm.

5.3. Coverage and length of intervals

Ideally, the estimators should provide statistical bounds (or intervals) that almost certainly contains the unknown parameter of interest, θ . If the condition is fulfilled, then a short interval is desirable. We begin by examining the coverage of θ of the intervals. In formal testing by ANOVA, we found no indication of a difference between SA and VS and we therefore pooled data from the two methods.

Table 2 gives the coverage of the three estimators for the used experimental combinations. There are two things to note in the table in addition to the fact that the coverage is near to 100

per cent. The first is that the coverage is similar for $n = 10$ and $n = 25$. The second is that the Weibull estimator almost certainly covers the optimum while the Jackknife estimators fail occasionally. From these findings we conclude that the Weibull estimator is preferable and that we offer further evidence to the findings of Brandeau and Chiu (1993) indicating that a large number of replicates is redundant.

Table 2: Proportion (%) of intervals containing θ .

Replicates	Time	E-C ^a	JK 1 st	JK 2 nd	Weibull
$n = 10$	Short	21	98.8	99.6	100.0
	Interm.	48	98.1	99.2	99.8
	Long	56	96.4	98.7	99.7
$n = 25$	Short	19	99.3	99.8	100.0
	Interm.	47	98.3	99.4	100.0
	Long	56	95.6	97.9	100.0

Note: ^{a)} Experimental combinations. SA and VS pooled due to similarities.

The 1st order Jackknife estimator gives intervals of a length of about 60 per cent of the Weibull estimator, whereas the 2nd order Jackknife estimator and Weibull estimator are comparable with regard to length of intervals. However, we require the estimator to give intervals with a factual coverage corresponding to the asserted confidence level, a condition the 1st order Jackknife estimator fails to meet.

Table 3: A comparison of statistical (Weibull estimator) and deterministic bounds (LR).
Counts of experimental combinations.

	$LB_W \geq LB_{LR}$	$LB_W < LB_{LR}$	
$UB_W \leq UB_{LR}$	231	9	240
$UB_W > UB_{LR}$	1	6	7
	232	15	247

To assess the precision offered by the Weibull intervals, it is useful to compare with the deterministic bounds given by LR. Table 3 gives a comparison of the upper bounds (UB) and lower bounds (LB) as obtained by the Weibull estimator and LR for the 247 used experimental combinations. In 231 of the combinations (93.5 %) is the Weibull interval, on

average, contained in the deterministic intervals. This happens as a consequence of the Weibull point-estimator being a better predictor of the optimum than the upper bound of LR, but more importantly because the deterministic lower bound almost always is below the statistical lower bound. The latter finding conforms to what Brandeau and Chiu (1993) reported.

The fact that the statistical intervals are contained in the deterministic intervals does not imply that the difference in length is of practical significance. We therefore compute the relative length of the statistical interval (average) to the deterministic interval.

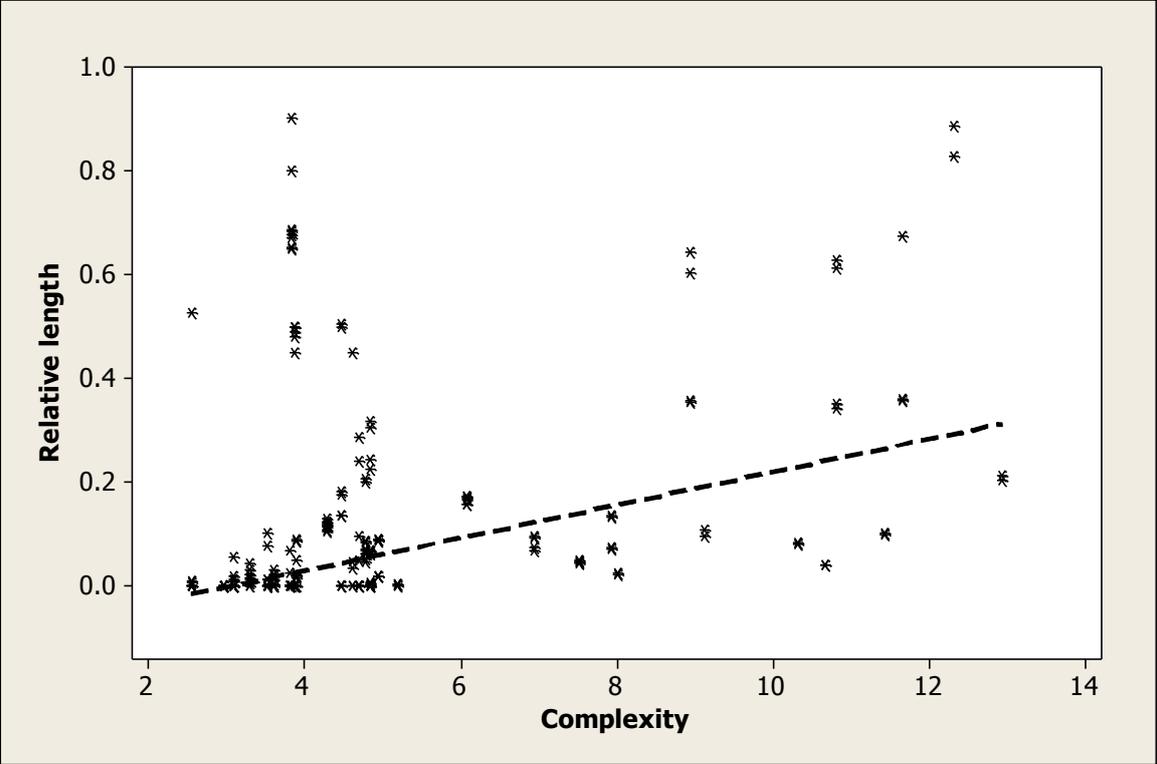


Figure 3: Relative length of the statistical intervals to the deterministic intervals as a function of the complexity. Resistant line imposed due to a few extreme outliers.

Figure 3 shows the relative length as a function of complexity as the other factors were found to have no impact. Imposed is a resistant line, in order to accommodate some extreme outliers, depicting the relationship between relative length and complexity (Velleman, 1980). The median relative length is 0.02 while the average is 0.25 being strongly affected by four extreme outliers coming from combinations with short running time.

To conclude the result section, reliable statistical bounds require heuristic solutions of good

quality (i.e. $SR \leq 4$). Provided that this is the case, the Weibull estimator gives proper intervals covering the optimum almost certainly with an interval substantially tighter than a deterministic interval. The required number of replicates is a modest of 10. Hence, statistical bounds are reliable, efficient and computationally affordable.

6. An illustrating problem

In this section, we illustrate statistical bounds applied to a practical location problem concerning allocating several distribution centers of the Swedish Post in one region in Sweden. In this real location problem the minimum is unknown. The problem is to allocate 71 distribution centers of the Swedish Post on some of the 6,735 candidate nodes in the network of Dalarna in mid-Sweden. The landscape of the region and its population of 277,725 inhabitants, distributed in 15729 demand points, is described by Carling, Han, and Håkansson (2012). Han, Håkansson, and Rebreyend (2013) provides a detailed description of the road network and urge that network distance, rather than the Euclidian, is used as distance measure. So we take the objective function as the distance in meters on the road network to the nearest postal center for the population.

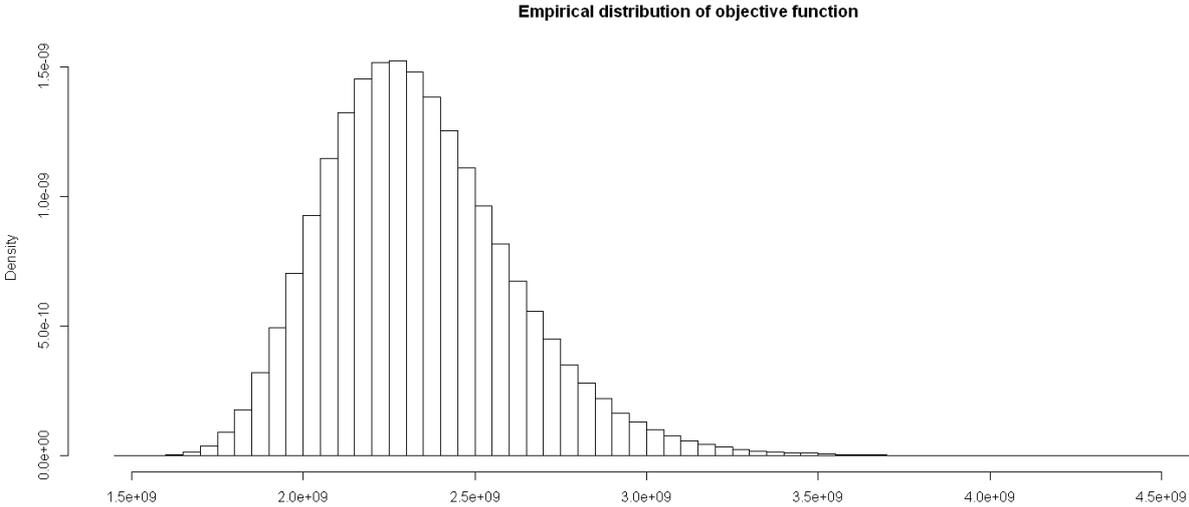


Figure 4: Empirical distribution of objective function for the Swedish Post problem.

To appreciate the complexity of this illustrative problem, we provide the distribution of the objective function. As Carling and Meng (2014) did for the OR-lib problems, we draw a

random sample of 1 million and show the empirical distribution of $g(z_p)$ in Figure 4. The distribution is slightly skewed to the right, but still approximately Normal. To evaluate the complexity of the problem, we use $\hat{\theta}_{JK}^{(2)}$ as the estimate of the minimum θ , and the mean and variance of $g(z_p)$ are derived by the 1 million random sample. The complexity is 5.47 and such problem is comparable to the median of the 40 OR-lib problems.

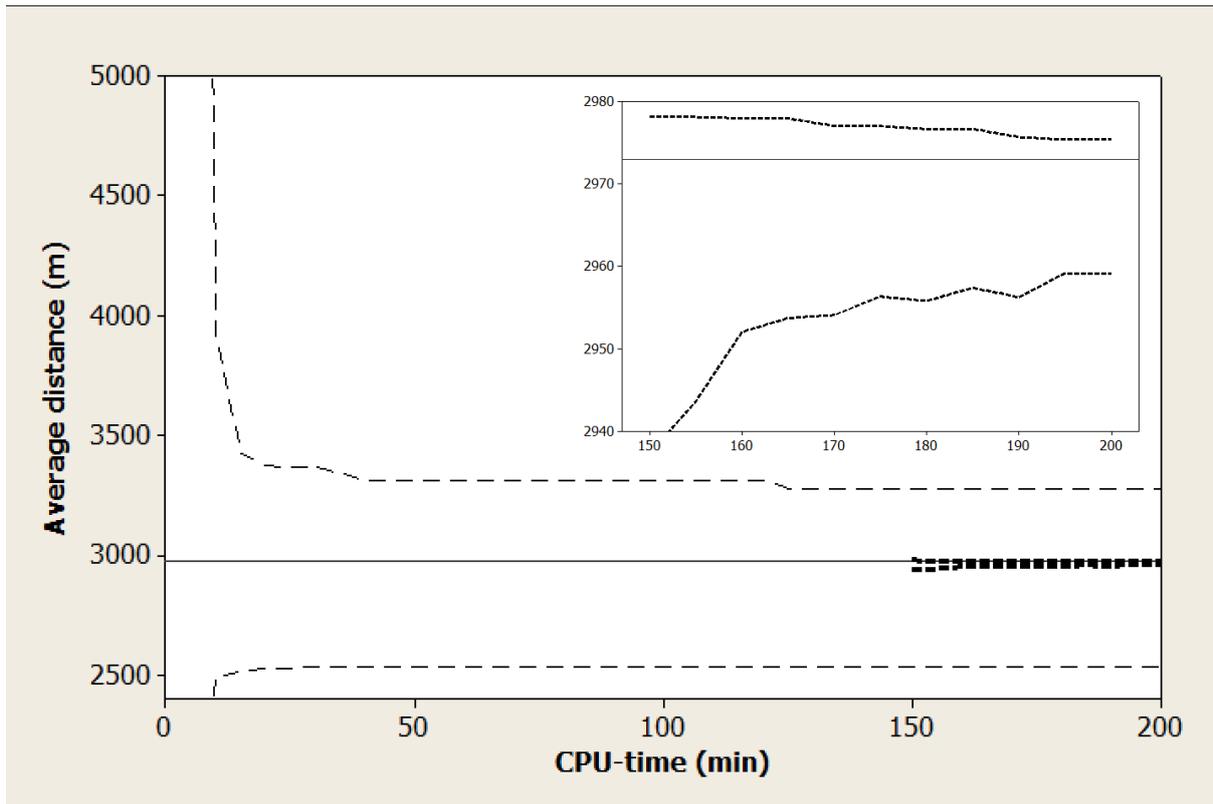


Figure 5: The solutions to the Swedish Post problem by running time. Upper and lower bounds in dashed lines (LR) and dotted lines (SA). Jackknife (1st order) point-estimate as solid reference line.

Drawing on the experimental results above, we set $n = 10$ and run the SA heuristic algorithm until SR reaches 4 or less. Furthermore, we focus on the statistical bounds of the Weibull estimator. We checked SR in steps of 5 minutes both for SA and LR. Figure 5 shows the evolution of LR's upper and lower bounds as a function of time. Within 90 minutes, the lower bound of LR has stabilized at 2,534 whereas the upper bound reaches 3,275 after 125 minutes. After our decision to stop running the algorithms at 200 minutes, LR's solution is 3,275 meters as the populations average distance to the distribution centers of the Swedish Post. However, the LR-solution is imprecise as the interval ranges from 2,534 to 3,275 meters

with a gap of 741 meters. The Jackknife 1st order point-estimate of the optimum is 2,973 as evaluated after 200 minutes, and this estimate is imposed in Figure 5 as a reference line to the unknown optimum.

The *SR* approaches 4 gradually, but slowly. It took 150 minutes of running time of SA for the ratio to come down to 4, and the statistical upper and lower bounds are thereafter computed and depicted in Figure 5. To appreciate the evolution of the statistical bounds, a graph is embedded in Figure 5 showing the evolution at 150 minutes of running time and onwards. The gap between the upper and lower statistical bounds is consistently closing and the gap amounts to 16 meters up on termination of the algorithm. Since 16 meters is a tolerable error for this application, there is no need to continue the search for a better solution and we content ourselves with a solution of 2,975 with a lower bound of 2,959 meters. We have also computed the statistical bounds using the other two estimators obtaining an almost identical result.

7. Concluding discussion

We have considered the problem of knowing when a solution provided by a heuristic is close to optimal. Deterministic bounds may sometimes be applicable and enough tight to shed knowledge on the problem. We have, however, studied statistical bounds potentially being of a more general applicability. We have examined the Weibull estimator as well as two variants on the Jackknife estimator. Furthermore, we have varied the number of replicates, running time of the heuristic algorithms, the complexity of the combinatorial problem as well as the heuristic algorithms. We find statistical bounds to be reliable and much more efficient than deterministic bounds provided that the heuristic solutions are sampled close to the optimum. Furthermore, statistical bounds may be computed based on a small number of replicates ($n = 10$) implying a modest computational cost up on exploiting parallel computing.

We have, however, restricted the experiment to one type of combinatorial optimization problem namely the p -median problem being the most common location problem in the OR-literature. Derigs (1985) made a number of concluding observations upon studying

statistical bounds with regard to the Travelling Salesman Problem (TSP) and Quadratic Assignment Problem (QAP). It appears that most of his conclusions are valid, except one. Derigs stated “The Weibull approach leads to a proper approach,”. We have however demonstrated that none of the estimators, including Weibull, are reliable unless the quality of the sample of heuristic solutions used for deriving the bounds is of high quality. To assess the quality we suggest using *SR* which is the standard deviation of the *n* solutions divided by the Jackknife point-estimator. The experiments suggest that *SR* exceeding 4 causes unreliable bounds. Nevertheless, a systematic study of statistical bounds on various classes of combinatorial optimization problems is warranted before advocating a general usage of statistical bounds in the estimation of optimums.

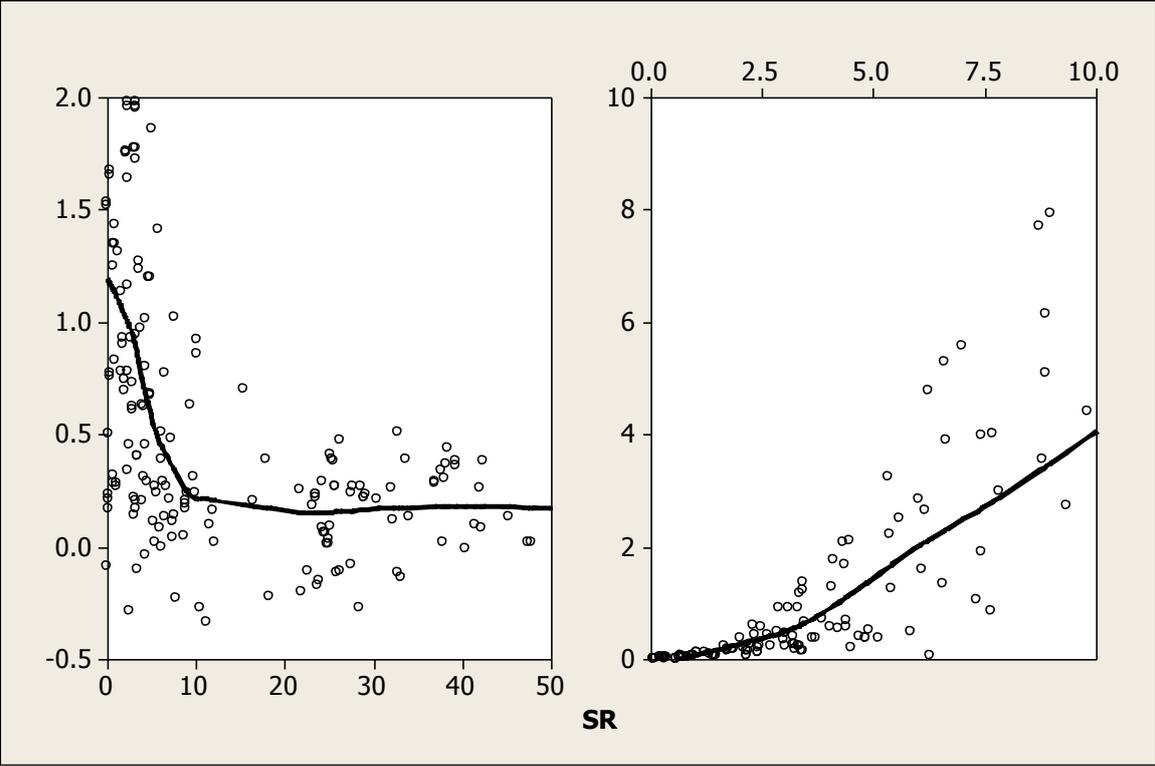


Figure 6: Skewness (left panel) and Relative (%) bias of average solution (right panel) as a function of *SR*. Combinations with *SR* equal to zero are removed and only the combinations for which $n = 25$ are shown.

The empirically discovered threshold of *SR* less than four may seem mystical. We have noted however that the distribution of the sample of solutions tend to be normal until the solutions are coming close to the optimum. The left panel of figure 6 shows skewness of the sample of solutions as a function of *SR*. For high values on *SR*, skewness is typically about zero and

kurtosis is about three. Once the sample of solutions is coming close to the optimum, the skewness (and kurtosis) increases (i.e. the sample of solutions is becoming right-skewed) while the variation in solution decreases (i.e. *SR* becomes smaller). The right panel of figure 6 shows the relative bias of the average solution of the sample. For *SR* large the bias is also large. However, at the threshold the bias is only about one per cent. Hence, reliable statistical bounds seem to require a solution method yielding solutions within about one per cent deviance from the optimum. The bias might be a better statistic than *SR* in deciding the reliability of the statistical bounds, but it requires of course knowing the (unknown) optimum.

An inherent difficulty in executing unbiased computer experiments on heuristic algorithms is the issue of their implementation. We have tried to render the comparison fair by running the algorithms on the same computers for the same CPU-time as well as closely following the original implementation of the algorithms in the same *R* environment. We think our results are robust to the implementation. The wide gap between the upper and lower bounds of LR did not seem to tighten by running time, on the contrary the bounds of the algorithm stabilized quickly in most experimental combinations. Vertex substitution was less successful than SA in fulfilling the requirement of *SR* below 4, but in the experimental combinations when both methods fulfilled the requirement the bounds were similar.

Acknowledgement

We are grateful to Mengjie Han, Johan Håkansson and Daniel Wikström for comments on an earlier draft of the paper. Financial support from the Swedish Retail and Wholesale Development Council is gratefully acknowledged.

References

- Akyüz, M.H., Öncan, T., Altınel, I.K., (2012). Efficient approximate solution methods for the multi-commodity capacitated multi-facility Weber problem. *Computers & Operations Research* 39.2, 225-237.
- Al-Khedhairi, A., (2008). Simulated annealing metaheuristic for solving p-median problem. *International Journal of Contemporary Mathematical Sciences*, 3:28, 1357-1365.

Beasley, J.E., (1990). OR library: Distributing test problems by electronic mail, *Journal of Operational Research Society*, 41:11, 1067-1072.

Beasley, J.E., (1993). Lagrangian heuristics for location problems. *European Journal of Operational Research*, 65, 383-399.

Brandeau, M.L., Chiu, S.S., (1993). Sequential location and allocation: worst case performance and statistical estimation. *Location Science*, 1:4, 289-298.

Carling, K., Han, M., Håkansson, J., (2012). Does Euclidean distance work well when the p -median model is applied in rural areas? *Annals of Operations Research*, 201:1, 83-97.

Carling, K., Meng, X., (2014). On statistical bounds of heuristic solutions to location problems. *Working papers in transport, tourism, information technology and microdata analysis*, 2014:10.

Cleveland, W.S., (1979). Robust Locally Weighted Regression and Smoothing Scatterplots, *Journal of the American Statistical Association*, 74, 829–836.

Dannenbring, D.G., (1977). Procedures for estimating optimal solution values for large combinatorial problems, *Management science*, 23:12, 1273-1283.

Daskin, M.S., (1995). *Network and discrete location: models, algorithms, and applications*. New York: Wiley.

Densham, P.J., Rushton, G., (1992). A more efficient heuristic for solving large p -median problems. *Papers in Regional Science*, 71, 307-329.

Derigs, U., (1985). Using confidence limits for the global optimum in combinatorial optimization. *Operations research*, 33:5, 1024-1049.

Efron, B., (1979). Bootstrap methods: Another look at the Jackknife. *Annals of statistics*, 7:1, 1-26.

Giddings, A.P., Rardin, R.L., Uzsoy, R., (2014). Statistical optimum estimation techniques for combinatorial problems: a review and critique. *Journal of Heuristics*, 20, 329-358.

Golden, B.L., Alt, F.B., (1979). Interval estimation of a global optimum for large combinatorial optimization. *Operations Research*, 33:5, 1024-1049.

Hakimi, S.L., (1964). Optimum locations of switching centers and the absolute centers and medians of a graph. *Operations Research*, 12:3, 450-459.

Hakimi, S.L., (1965). Optimum Distribution of Switching Centers in a Communication Network and Some Related Graph Theoretic Problems. *Operations Research*, 13:3, 462-475.

Han, M., (2013). *Heuristic Optimization of the p -median Problem and Population Re-distribution*. Dalarna Doctoral Dissertations, 2013:01.

- Han, M., Håkansson, J., Rebreyend, P. (2013). How do different densities in a network affect the optimal location of service centers?. Working papers in transport, tourism, information technology and microdata analysis, 2013:15.
- Handler, G.Y., Mirchandani, P.B., (1979). Location on networks: Theorem and algorithms. MIT Press, Cambridge, MA.
- Luis, M., Sahli, S., Nagy, G., (2009). Region-rejection based heuristics for the capacitated multi-source Weber problem. Computers & Operations Research, 36, 2007-2017.
- McRobert, K.L., (1971). A search model for evaluating combinatorially explosive problems. Operations Research, 19, 1331-1349.
- Meng, X., Carling, K., (2014). How to Decide Upon Stopping a Heuristic Algorithm in Facility-Location Problems?. In Web Information Systems Engineering–WISE 2013 Workshops, Lecture Notes in Computer Science, 8182, 280-283, Springer, Berlin/Heidelberg.
- Michalewicz, Z., Janikow, C.Z. (1991). Genetic algorithms for numerical optimization. Statistics and Computing, 1, 75-91.
- Monroe, H.M., Sielken, R.L., (1984). Confidence limits for global optima based on heuristic solutions to difficult optimization problems: a simulation study. American Journal of Mathematical and Management Sciences, 4, 139-167.
- Nydick, R.L., Weiss, H.J., (1988). A computational evaluation of optimal solution value estimation procedures. Computers & Operations Research, 5, 427-440.
- Quenouille, M.H., (1956). Notes on bias in estimation. Biometrika, 43, 353-360.
- Reese, J., (2006). Solution methods for the p -median problem: An annotated bibliography. Networks, 48:3, 125-142.
- Robson, D.S., Whitlock, J.H., (1964). Estimation of a truncation point. Biometrika, 51, 33-39.
- Velleman, P.F., (1980). Definition and Comparison of Robust Nonlinear Data Smoothing Algorithms. Journal of the American Statistical Association, 75, 609-615.
- Wilson, A.D., King, R.E., Wilson, J.R., (2004). Case study on statistically estimating minimum makespan for flow line scheduling problems. European Journal of Operational Research, 155, 439-454.

Appendix: Table A1

Table A1: Description of the 40 problems of the OR-library.

Problem	θ	$\mu_{g(z_p)}$	$\sigma_{g(z_p)}$	Complexity
P11	7696	10760	1195	2.56
P1	5819	8426	877	2.97
P16	8162	11353	1033	3.09
P6	7824	10522	869	3.10
P26	9917	13644	1133	3.29
P21	9138	12906	1070	3.52
P38	11060	15078	1143	3.52
P31	10086	13960	1077	3.60
P35	10400	14179	1085	3.81
P7	5631	7930	598	3.84
P2	4093	6054	506	3.88
P3	4250	6194	500	3.89
P27	8307	11428	727	4.29
P17	6999	9819	631	4.47
P22	8579	11699	676	4.62
P12	6634	9387	586	4.70
P36	9934	13436	735	4.77
P39	9423	12988	736	4.84
P32	9297	12687	699	4.85
P4	3034	4618	320	4.95
P5	1355	2376	197	5.18
P8	4445	6604	356	6.07
P13	4374	6293	276	6.95
P9	2734	4250	202	7.51
P18	4809	6769	248	7.92
P10	1255	2278	127	8.02
P23	4619	6586	220	8.94
P14	2968	4501	168	9.12
P28	4498	6369	188	9.95
P19	2845	4327	144	10.32
P15	1729	2896	109	10.67
P33	4700	6711	186	10.81
P24	2961	4486	134	11.42
P37	5057	7246	188	11.65
P20	1789	3108	112	11.73
P40	5128	7329	179	12.32
P29	3033	4559	118	12.93
P25	1828	3131	95	13.64
P34	3013	4617	112	14.36
P30	1989	3335	90	14.93

Appendix: R-code

Simulated Annealing:

```
V=100 # Number of candidates (dependent on test problem)
p=5 # Number of facilities (dependent on test problem)
n<-100 # Number of replicates (dependent on experimental combination)
ni<-10000 # Number of replicates (dependent on experimental combination)
heuristic.solution<-matrix(0, nrow=ni, ncol=n)
heuristic.location<-matrix(0, nrow=p, ncol=n)
Store.solution<-numeric()
Best.solution<-numeric()
Store.location<-matrix(0, nrow=ni, ncol=p)
Best.location<-numeric()
for (i in 1:n){
  subset<-numeric(0)
  select.location<-sample(1:V,p,replace=F)
  objective.function<-sum(apply(Distance.matrix[,select.location],1,min))
  iteration<-0;Tempearture<-400;beta<-0.5 # initial parameter setting.
  while (iteration<ni){
    sam<-sample(1:p,1)
    substitution<-sample((1:V)[-select.location[sam]],1)
    store.selection<-select.location
    select.location[sam]<-substitution
    updated.objective.function<-sum(apply(Distance.matrix[,select.location],1,min))
    if (updated.objective.function<=objective.function)
  {objective.function<-updated.objective.function;beta<-0.5;count<-0}
    if (updated.objective.function>objective.function){
      delta<-updated.objective.function-objective.function
      unif.number<-runif(1,0,1)
      if (unif.number<exp(-delta/Tempearture))
  {objective.function<-updated.objective.function;beta<-0.5;count<-0}
      if (unif.number>=exp(-delta/Tempearture)) {count<-count+1;select.location<-
store.selection } }
    iteration<-iteration+1
    Tempearture<-Tempearture*0.95
    Store.solution[iteration]<-objective.function
    Best.solution[iteration]<-min(Store.solution[1:iteration])
    Store.location[iteration,]<-select.location
    Best.location<-Store.location[min(which(Store.solution==Best.solution[iteration]))],]
  }
  heuristic.solution[,i]<-Best.solution
  heuristic.location[,i]<-Best.location
}
```

Vertex Substitution:

```
iteration=0
current.P.node<-sample(1:V,p,replace=F)
minimum.P.list<-current.P.node
candidate.node<-c(rep(0, (V-p)))
temp<-numeric()
min.obj.funciton=sum(apply(Distance.matrix[,current.P.node],1,min))
i=0; j=0; k=0; f=0
  for(i in 1:V){ f=0;
    for(j in 1:p){
      if(current.P.node[j]==i){ f=1;
        break } }
    if(f==0){
      candidate.node[k]=i;
      k=k+1;
      if(k==(V-p)) # if duplicate in plist stop earlier otherwise may have error break;
    } }
while(iteration<ni){
  for(i in 1:p){
    for(j in 1:(V-p)){
      for(k in 1:p){
        if(k==i){
          temp[k]=candidate.node[j];
        }else{
          temp[k]=current.P.node[k]}
      curOFV=sum(apply(Distance.matrix[,temp],1,min))
      if(curOFV<min.obj.funciton){
        min.obj.funciton=curOFV;
        minimum.P.list<-temp
        cont=1
      } } }
  current.P.node=minimum.P.list
  i=0; j=0; k=0; f=0
  for(i in 1:V){ f=0;
    for(j in 1:p){
      if(current.P.node[j]==i){ f=1;
        break } }
    if(f==0){
      candidate.node[k]=i;
      k=k+1;
      if(k==(V-p)) # if duplicate in plist stop earlier otherwise may have error break;
    } }
  iteration=iteration+1
  tbq1[iteration]<-min.obj.funciton }
```

Lagranian Relexation:

```
objective.f.matrix<-indicator.matrix<-matrix(0,N,N)
Distance.matrix<-ma1
lambda<-numeric()
V<-numeric()
alpha<-2
lb.cu<-lb.pr<-ub.cu<-ub.pr<-NULL
Lower.bound<-1e13;Upper.bound<-1e13
counter<-0
counter.up<-0
iter<-1

location.m<-sample(1:V,p,replace=F)
lambda<-apply(Distance.matrix,1,mean)/8
run.time=0
start.time<-as.numeric(Sys.time())
while(run.time<40){
objective.f.matrix<-Distance.matrix-lambda # Distance.matrix is the data
indicator.matrix<-(objective.f.matrix<0)
for (j in 1:N){V[j]<-sum(objective.f.matrix[,j][indicator.matrix[,j])}
location<-order(V)[1:p]
Lower.cu<-sum(lambda)+sum(V[location])
if(Lower.cu>Lower.bound){Lower.bound<-Lower.cu}
if(Lower.cu<Lower.bound){counter<-counter+1} # made when no impro. in lower bound
lb.cu<-c(lb.cu,Lower.cu);lb.pr<-c(lb.pr,Lower.bound)
Upper.cu<-sum(apply(Distance.matrix[,location],1,min))
if(Upper.cu<Upper.bound){Upper.bound<-Upper.cu;counter.up<-0}
if(Upper.cu>=Upper.bound){counter.up<-counter.up+1}
ub.cu<-c(ub.cu,Upper.cu);ub.pr<-c(ub.pr,Upper.bound)
location.m<-cbind(location.m,location)

violation<-sum((apply(indicator.matrix[,location],1,sum)-1)^2)
if(counter==5){alpha<-alpha/2
counter<-0}
update<-alpha*(Upper.bound-Lower.cu)/violation
multi<-(apply(indicator.matrix[,location],1,sum)-1)*update
lambda<-pmax(0,lambda-multi)
iteration.time<-as.numeric(Sys.time())
run.time<-iteration.time-start.time
iter=iter+1
}
node<-which(ub.pr==Upper.bound)[1]+1
optimal<-sum(apply(Distance.matrix[,location.m[,node]],1,min))
c(Upper.bound,Lower.bound)
```