A quantitative analysis of iterated local research

A Quantitative Analysis of Iterated Local Search

T.W.M. Vossen [1] M.G.A. Verhoeven [1] H.M.M. ten Eikelder [1] E.H.L. Aarts [1,2]

[1]Eindhoven University of Technology, Department of Mathematics and Computing Science, P.O. Box 513, 5600 MB Eindhoven, Netherlands

[2]Philips Research Laboratories, P.O. Box 80000, 5600 JA Eindhoven, Netherlands

Abstract

We present a probabilistic analysis of the average case-behavior of the 2-opt algorithm for the Traveling Salesman Problem. We derive an expression for the distribution of local minima of the 2-opt neighborhood. This distribution is numerically computed and empirically validated for a restricted version of the 2-opt neighborhood. Furthermore, we present a semi-empirical analysis of the behavior of an iterated 2-opt algorithm based on extensive numerical results. Finally, we discuss the expected time to find a solution within a given range from optimality.

1 Introduction

Local search algorithms are a class of approximation algorithms for combinatorial optimization problems, i.e., problems in which an optimal solution has to be found among a finite number of alternatives. The performance of a local search algorithm can be quantified by the relative error of the obtained solutions and its running time. Extensive empirical results have shown that there is a considerable difference between the worst-case and average-case behavior of local search algorithms, and that occurrences of the worst case are rare. Hence, average-case analyzes could be useful, and provide a better understanding of local search algorithms.

In this paper we present a probabilistic analysis of the average-case performance of a simple local search algorithm. We concentrate on the well-known 2-opt algorithm for the Traveling Salesman Problem (TSP) of Lin [1965]. We formulate a probabilistic model, which we use to predict the average quality of the final solutions. Furthermore, we analyze the performance of iterated local search methods.

2 Local search

Local search algorithms constitute a class of approximation algorithms for combinatorial optimization that are based on repeatedly replacing a solution by a neighboring solution. An essential concept in local search algorithms is the notion of a neighborhood structure.

Definition 2.1. An instance of a combinatorial optimization problem is given by a compact representation of a pair $(\mathcal{S}, f)$, where $\mathcal{S}$ is the set of solutions, and $f : \mathcal{S} \rightarrow \mathbb{R}$ is a function that gives the cost of a solution. The objective is to find a solution with minimal cost. A *neighborhood structure* $\mathcal{N} : \mathcal{S} \rightarrow \mathcal{P}(\mathcal{S})$ assigns to each solution a set of solutions, called a *neighborhood*, that

can be directly obtained from it. A solution $s \in S$ is a *local optimum* w.r.t. $\mathcal{N}$, if $f(s') \geq f(s)$ for each $s' \in \mathcal{N}(s)$. □

The basic local search algorithm is the iterative improvement algorithm. An iterative improvement algorithm starts off with a randomly chosen solution or a solution constructed by some heuristic. Next, the algorithm repeatedly tries to improve the current solution by replacing it with a neighboring solution with lower cost. If a solution has been reached that has no neighbors with lower cost, a locally optimal solution has been found. A pivoting rule determines which neighboring solution becomes the new current solution. Well-known pivoting rules are *first improvement*, replacing the current solution by the first neighbor found with lower cost, and *best improvement* in which the lowest cost neighbor replaces the current solution. An iterative improvement algorithm with best-improvement is called a *steepest descent* algorithm.

Local search algorithms have the advantage of being generally applicable and flexible, since they only require a specification of the solution space, a cost function and a neighborhood structure. A disadvantage is that local search algorithms may get trapped in local minima of poor quality. To overcome this disadvantage, many variants of the basic local search algorithm have been proposed in literature. Well-known examples are simulated annealing, tabu search, and genetic local search. For an overview of local search we refer to [Aarts & Lenstra, 1995].

In this paper we concentrate on iterative improvement and iterated local search [Johnson, 1990], which can be considered as the starting point for the analysis of more advanced local search algorithms. Iterated local search algorithms repeatedly execute an iterative improvement algorithm. Each time the iterative improvement algorithm terminates, the obtained local minimum is modified and the iterative improvement algorithm is restarted with the modified local minimum. The choice of a modification mechanism is usually guided by two principles: *intensification* and *diversification*. Intensification is based on the observation that good solutions seem to have common properties. Therefore, the modification mechanism should try to preserve these properties, thus intensifying the search into a special region of the solution space. On the other hand, diversification of the search is needed to ensure that no region of the solution space is neglected. These two principles can be found in many advanced algorithms, e.g., the use of tabu lists in tabu search and the cross-over step used in genetic local search.

2.1 Local search for the traveling salesman problem

Probably the best-known combinatorial optimization problem is the TSP. In the TSP a salesman wishes to visit a number of cities and return to the starting point, in such a way that each city is visited exactly once, and the total distance covered is as short as possible. The TSP belongs to the class of NP-hard problems and therefore considerable effort has gone into designing efficient approximation algorithms. Formally, the TSP can be defined as follows.

Definition 2.2. Let V be a set of N cities and $d_{ij} \in \mathbb{R}$ a distance for each $i, j \in V$. A tour t is a set of N edges $\{e_0, \ldots, e_{N-1}\}$ that constitutes a Hamiltonian cycle in the complete graph $(V, V \times V)$. A tour t is represented by a bijection $\pi : \{0, \ldots, N-1\} \rightarrow V$, where π_i gives the city at the i^{th} position in t, such that $e_i = (\pi_i, \pi_{(i+1) \bmod N})$. The solution space S of a TSP

instance is the set of all tours. The cost function f is given by

$$f(t) = \sum_{(i,j)\in t} d_{ij}.$$

The problem is to find $t \in \mathcal{S}$ for which $f(t)$ is minimal. □

We consider only *symmetric* instances of the TSP, in which the distances satisfy $d_{ij} = d_{ji}$ for each $i, j \in V$. From now on, we consider all arguments of bijection π to be modulo N, that is, π_i denotes $\pi_{i \bmod N}$.

The choice of the neighborhood structure has great influence on the average cost of the solutions found by local search algorithms. Therefore various neighborhood structures have been introduced for the TSP, most of which are based on edge exchanges. In the 2-opt neighborhood of Lin [1965] a tour t' is a neighbor of tour t, if t' can be obtained from t by removing two edges and inserting two edges such that t' is obtained. Another well-known neighborhood is the variable-depth neighborhood of Lin & Kernighan [1973].

For an overview of the worst-case complexity and empirical behavior of local search for the TSP, we refer to [Johnson, 1990]. The main conclusion from the empirical results in this paper is that local search algorithms can find good-quality solutions within low-order polynomial empirical running time, but it is conjectured that worst-case running times can not be bounded polynomially. Furthermore, it is not possible to give upper bounds on the relative error of local minima.

Other investigations have dealt with the theoretical average-case behavior of local search. Kern [1989] showed with a probabilistic analysis that the 2-exchange algorithm for Euclidian instances of the TSP has an average-case running time that is polynomially bounded. Other probabilistic models for local search have been studied by Tovey [1985]. In these studies artificial problems are considered with special classes of neighborhood graphs with regular structures, e.g. the hypercube. The cost function for these problems is chosen to induce an orientation on this regular structure. Different probability distributions on the orientations are considered and for some cases low order polynomial average case running times are proved. More recently, Chandra, Karloff & Tovey [1994] showed similar results for 2-opt and 3-opt algorithms for the TSP, and derived weak upper bounds, both worst case and probabilistic, on the cost of the local minima obtained by these algorithms.

Another approach is studied by Nakano & Nakanishi [1983], who model local search algorithms by Markov chains. Using this method, they obtain distributions for the cost of locally optimal solutions for 2-opt and 3-opt algorithms, based on assumptions for the transition probabilities and the probability that a solution is a local optimum.

3 A probabilistic analysis of the 2-opt neighborhood

In this section we analyze the average-case performance of local search algorithms that use the 2-opt neighborhood structure. The 2-opt neighborhood $\mathcal{N}_2$ is specified as follows.

Definition 3.1. Let $t \in \mathcal{S}$ be represented by π. Define the function 2-*change*$(t, r, s) : \mathcal{S} \times \mathbb{N} \times \mathbb{N} \to \mathcal{S}$ which gives for each tour the neighbor that is obtained by removing the two edges at the r^{th} and s^{th} position in the tour and inserting two different edges, that is

$$2\text{-}change(t, r, s) = t \setminus \{(\pi_r, \pi_{r+1}), (\pi_s, \pi_{s+1})\} \cup \{(\pi_r, \pi_s), (\pi_{r+1}, \pi_{s+1})\}.$$

Then, $\mathcal{N}_2$ is defined by $\mathcal{N}_2(t) = \{2\text{-}change(t, r, s) \mid 0 \leq r, s < N \wedge 1 < r - s < N - 1\}$. □

3.1 Preliminaries

An instance of the TSP is completely specified by its distance matrix. As the first step in our approach we assume that the distances d_{ij} are independently drawn from some distribution. This is a common approach in probabilistic analyzes; examples can be found in [Kirkpatrick & Toulouse, 1985; Weinberger, 1991; Stadler & Schnabl, 1992]. However, it should be noted that the assumption of independence is in fact a restriction that excludes Euclidean instances, because then the triangle inequality holds. Still, Euclidian distances are sometimes considered to be effectively independent [Bonomi & Lutton, 1984]. In view of the above assumption, we can easily define a *class* of instances by letting the edge lengths $\underline{d}_{ij}$ be independent, identically distributed random variables. Consequently, we can also view the costs of solutions as random variables, i.e., the sum of N independent, identically distributed edge lengths. Hence the cost approximates the normal distribution, according to the central limit theorem. Formally, this is stated as follows.

Theorem 3.1. *Let the distances $\underline{d}_{ij}$ be independent, identically distributed random variables, with mean μ_l and variance σ_l^2. Define $\mu = N\mu_l$ and $\sigma^2 = N\sigma_l^2$.*
Then, the cost $\underline{f}(t)$ of a solution t has a normal distribution with density

$$\omega(\underline{f}(t)) = \frac{1}{\sqrt{2\pi}\sigma}\exp[-(\underline{f}(t) - \mu)^2/2\sigma^2].$$

□

Thus, we can represent a class of instances as a tuple $(S, \underline{f})$, where $\underline{f}$ is a function that assigns a random variable to each solution.

The next step is to express the relationships between different solutions in this class. Usually, the dependence between two random variables is measured by their covariance, so it is natural to express the relation between solution by the covariance of their cost. The covariance $r_{st} = \mathbb{E}[(\underline{f}(s) - \mu)(\underline{f}(t) - \mu)]$ between the costs of solutions s and t is calculated as follows:

Theorem 3.2. *Suppose the cost of two tours $s, t \in S$ is given by $\underline{f}(s) = \sum_{(i,j)\in s} \underline{d}_{ij}$ and $\underline{f}(t) = \sum_{(i,j)\in t} \underline{d}_{ij}$. Define $K = |s \cap t|$, that is, the number of edges s and t have in common. Then, $r_{st} = K\sigma_l^2$.*

Proof. Since the covariance distributes over addition we find

$$r_{st} = \sum_{(i,j)\in s}\sum_{(i',j')\in t} \mathrm{cov}(\underline{d}_{ij}, \underline{d}_{i'j'}).$$

Due to the independence of the edge lengths the covariance $\mathrm{cov}(\underline{d}_{ij}, \underline{d}_{i'j'})$ will be either 0 if $\underline{d}_{ij} \neq \underline{d}_{i'j'}$, or σ_l^2 if $\underline{d}_{ij} = \underline{d}_{i'j'}$. Then, it immediately follows that $r_{st} = K\sigma_l^2$. □

In order to express the influence of the covariance, an important aspect of our analysis deals with conditional probabilities. The following theorem given by Papoulis [1965] shows how, given the cost of a solution, the covariance affects the distribution of the costs of other solutions.

Theorem 3.3. *Let $s, t, u \in S$ be tours, whose costs are normally distributed with mean μ and variance σ. Let $\rho(s, t)$ be the correlation coefficient r_{st}/r_{tt}. Then the cost of tour s, given the*

cost of tour t, is normally distributed with mean

$$\mathbb{E}[\underline{f}(s) \mid \underline{f}(t) = c] = \mu + \rho(s,t)(c - \mu).$$

Given the cost of tour t, the covariance between tours s and u can be expressed as

$$\mathbb{E}[(\underline{f}(s) - \mu)(\underline{f}(u) - \mu) \mid \underline{f}(t) = c] = \sigma^2(\rho(s,u) - \rho(s,t)\rho(u,t)).$$

□

So far, we calculated properties of one tour, averaging over all instances in a class. However, the properties we are interested in are instance properties, i.e., averages taken over all tours in one instance. The mean, variance, and covariances of an instance are defined as follows:

Definition 3.2. Let $(\mathcal{S}, f)$ be an instance of the TSP. Then the instance mean $\hat{\mu}$ is given by

$$\hat{\mu} = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} f(s).$$

The instance variance $\hat{\sigma}^2$ is given by

$$\hat{\sigma}^2 = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} f^2(s) - \hat{\mu}^2.$$

Let $\mathcal{D}_k(s)$ be the set of all tours that have $N - k$ edges in common with a tour $s \in \mathcal{S}$. Then the instance covariance $\hat{r}_k$ between all pairs of tours that have $N - k$ edges in common is given by

$$\hat{r}_k = \frac{1}{|\mathcal{S}| \cdot |\mathcal{D}_k|} \sum_{s \in \mathcal{S}} \sum_{s' \in \mathcal{D}_k(s)} f(s) f(s') - \hat{\mu}^2.$$

□

Since the number of tours is extremely large, it is impossible to calculate the instance averages exactly. The question is now how to obtain expressions for these averages. To answer this question we prove first that, on the average, the instance mean, variance and covariance are approximately the same as their class equivalents.

Theorem 3.4. *Let* $\epsilon = 2\frac{N}{N-1}\sigma_f^2$. *Then we have*

$$\begin{aligned} \mathbb{E}[\hat{\mu}] &= \mu, \\ \mathbb{E}[\hat{\sigma}^2] &= \sigma^2 - \epsilon, \\ \mathbb{E}[\hat{r}_k] &= (N-k)\sigma_f^2 - \epsilon. \end{aligned}$$

Proof. See Appendix. □

Still, one might argue that for one specific instance the quantities can be substantially different from the average. In order to show that these differences effectively disappear, we introduce the notion of *self-averaging* properties, cf. [Stadler & Schnabl, 1992].

Definition 3.3. A quantity $X(N)$ of an instance of the TSP with N cities is called self–averaging if

$$\lim_{N \to \infty} \left(\frac{(\mathbb{E}[X^2(N)] - \mathbb{E}^2[X(N)])^{\frac{1}{2}}}{\mathbb{E}[X(N)]} \right) = 0.$$

□

Informally seen, this means that as the number of cities increases, the quantity $X(N)$ approaches the mean for all instances in the class. For the quantities defined in Definition 3.2, the following theorem holds

Theorem 3.5. *The instance quantities $\hat{\mu}$, $\hat{\sigma}^2$ and $\hat{r}_k$ are self–averaging.*
Proof. See Appendix. □

Consequently, we conclude from Theorems 3.4 and 3.5 that class properties can be good estimates for the properties of an instance.

3.2 The distribution of local minima

Next we express analytically the distribution of the costs of local minima, based on the approach outlined in the previous section. However, the expression we obtain proves to be very hard to evaluate or even approximate, because of the complex dependencies induced by the neighborhood structure $\mathcal{N}_2$. Therefore, we introduce a restricted version of the 2-opt neighborhood, for which we give an expression for the distribution of local minima that can be evaluated numerically.

In order to express the distribution of local minima, we consider first an instance class $(S, \underline{f})$ of the TSP. In the previous section we have established that the costs of the tours in this class are normally distributed, and we have expressed the mean and variance of this distribution, and the covariance between tours. We use these quantities to express the *probability* that a solution is a local minimum, i.e., the probability that all its neighbors have higher cost.

Our goal is to find the density of the costs of local minima, which can easily be expressed as the density of the cost c of a tour $t \in S$, given that t is a local minimum.

$$\omega(c \mid t \text{ is a local minimum}) = \frac{\mathbb{P}\{t \text{ is a local minimum} \mid c\}\omega(c)}{\mathbb{P}\{t \text{ is a local minimum}\}}. \tag{1}$$

Of course, these expressions are properties of one tour in a class of instances. Again, we have to show how these properties are related to their instance equivalents, that is, the local minima and the density of the local minima in an instance. This is done in the following theorem and conjecture.

Theorem 3.6. *Let (S, f) be an instance of the TSP. Define $LM(f)$ to be the number of local minima in S and $LM(f, c)$ to be the number of local minima in S with cost c. Let t be a randomly chosen tour. Then,*

$$\begin{aligned} \mathbb{E}[LM(f)] &= |S| \cdot \mathbb{P}\{t \textit{ is a local minimum}\}, \\ \mathbb{E}[LM(f,c)] &= |S| \cdot \mathbb{P}\{t \textit{ is a local minimum} \mid c\}\omega(c). \end{aligned}$$

Proof. See Appendix. □

Conjecture 3.1. *Let (S, f) be an instance of the TSP, and let $LM(f)$ and $LM(f, c)$ be as defined in Theorem 3.6. Let t be a randomly chosen tour. Then, we conjecture that*

$$\mathbb{E}\left[\frac{LM(f,c)}{LM(f)}\right] = \omega(c \mid t \textit{ is a local minimum}).$$

□

An intuitive foundation for Conjecture 3.1 can be found in the Appendix. Hence, we consider expression (1) to be a good estimate for the density of the costs of local minima in an instance.

Consider now a randomly chosen tour t_0 with cost c_0, and let its neighbors have cost $c_1, \ldots, c_b$, where b is the number of neighbors. We denote the *joint density* of these costs as $\omega(c_0, \ldots, c_b)$. Now, we express the probability that t_0 is a local minimum given its cost as

$$\mathbb{P}\{t_0 \text{ is a local minimum} \mid c_0\} = \int_{c_0}^{\infty} dc_1 \cdots \int_{c_0}^{\infty} dc_b \omega(c_1, \ldots, c_b \mid c_0), \tag{2}$$

and the probability that t_0 is a local minimum as

$$\mathbb{P}\{t_0 \text{ is a local minimum}\} = \int_{-\infty}^{\infty} \mathbb{P}\{t_0 \text{ is a local minimum}|c_0\}\omega(c_0)dc_0.$$

Consequently, the problem reduces to expressing the probability that a tour is a local minimum, given its cost. In order to evaluate conditional probability (2) we assume that the costs of a tour and its neighbors have a multivariate normal distribution, which is defined as follows.

Definition 3.4. Let the multidimensional random variable $\mathbf{c}$ denote the cost vector $(c_0, \ldots, c_b)$ and μ the mean vector. Let $\mathbf{R} = [r_{ij}]$ be the covariance matrix, with $\|\mathbf{R}\|$ its determinant and $\mathbf{R}^{-1}$ its inverse. Then $\mathbf{c}$ has a *multivariate normal distribution*, if $\mathbf{R}$ is positive definite, i.e., $\mathbf{x}\mathbf{R}\mathbf{x}^{\mathbf{T}} > 0$ for all non-zero vectors $\mathbf{x}$, and if the joint density is given by

$$\omega(\mathbf{c}) = \frac{1}{(2\pi)^{(b+1)/2}\|\mathbf{R}\|}\exp[(\mathbf{c}-\mu)\mathbf{R}^{-1}(\mathbf{c}-\mu)^{\mathbf{T}}/2].$$

□

Tong [1990] states that a multidimensional normal random variable $\mathbf{c}$ has this density if all linear combinations of the costs $c_0, \ldots, c_b$ are univariate normally distributed. This means that for all non–zero vectors $\mathbf{a}$, $\mathbf{a}\mathbf{c}^{\mathbf{T}}$ has a normal distribution with mean $\mathbf{a}\mu\mathbf{a}^{\mathbf{T}}$ and variance $\mathbf{a}\mathbf{R}\mathbf{a}^{\mathbf{T}}$. Since any linear combination of the costs can be rewritten to a linear combination of the edge lengths, this holds if the edge lengths are drawn from a normal distribution, because any sum of normally distributed random variables is again normally distributed. In case edge lengths are not drawn from a normal distribution, we assume that the linear combination of the edge lengths approximates a normal distribution, due to the central limit theorem.

In order to specify the multidimensional normal density completely, we have to calculate the elements in the covariance matrix $\mathbf{R}$. The elements r_{ij} in the covariance matrix can be obtained by Theorem 3.2, in which we showed that the covariance between two tours is determined by the number of edges they have in common. Since a tour and its neighbor always have $N-2$ edges in common, it is easy to see that

$$r_{0i} = r_{i0} = (N-2)\sigma_l^2 \text{ for } 1 \leq i \leq b.$$

The covariance between two neighbors of a tour is determined in the following theorem, which states how many edges they have in common.

Theorem 3.7. *Let t' = 2-change(t, r, s) and t'' = 2-change(t, u, v) be different neighbors of t. Denote the successor of a two-change as succ(2-change(t, r, s)) = 2-change$(t, r+1, s+1)$. Then, the number of edges t' and t'' have in common is equal to $N-3$, if $\{r, s\} \cap \{u, v\} \neq \emptyset$ or*

succ(2-*change*(t, r, s)) = 2-*change*(t, u, v), *or succ*(2-*change*(t, u, v)) = 2-*change*(t, r, s). *In the remaining cases t' and t'' have $N-4$ edges in common.* □

Consequently, the covariance between two neighbors of t is either $(N-3)\sigma_l^2$, if the two-changes remove or introduce a common edge, or $(N-4)\sigma_l^2$ otherwise. For each neighbor 2-*change*(t, r, s) there are exactly $2(N-3)-2$ other neighbors of t resulting from the removal of either edge e_r or edge e_s. Adding the succeeding and preceeding 2-change we find that each neighbor has $N-3$ edges in common with $2N-6$ other neighbors of t.

We have now obtained an expression for the joint density $\omega(c_0, \ldots, c_b)$. However, to express the probability integral (2) we need the conditional density $\omega(c_1, \ldots, c_b \mid c_0)$. Papoulis [1965] states that the conditional density is again multivariate normal, and can be expressed as

$$\omega(c_1, \ldots, c_b \mid c_0) = \frac{1}{(2\pi)^{b/2}\|\mathbf{R}'\|}\exp[-\frac{1}{2}(\mathbf{c}-\boldsymbol{\mu}')\mathbf{R}'^{-1}(\mathbf{c}-\boldsymbol{\mu}')^{\mathrm{T}}].$$

The conditional mean vector $\boldsymbol{\mu}'$ and the elements in the conditional covariance matrix $\mathbf{R}'$ are expressed in terms of the mean vector $\boldsymbol{\mu}$ and covariance matrix $\mathbf{R}$. The vector $\boldsymbol{\mu}'$ contains the conditional means, and is calculated by applying Theorem 3.3, that is,

$$\mu_i' = \mathbb{E}[c_i \mid c_0] = (1-2/N)c_0 + (2/N)\mu \quad \text{for } 1 \le i \le b.$$

The matrix $\mathbf{R}' = [r_{ij}']$ is the conditional covariance matrix. Its elements can be calculated by applying Theorem 3.3 and 3.7. This leads to

$$r_{ij}' = \begin{cases} (4-4/N)\sigma_l^2 & : \ i = j, \text{ the conditional variance.} \\ (-4/N)\sigma_l^2 & : \ i \neq j \text{ and tours } i \text{ and } j \text{ have } N-4 \text{ edges in common.} \\ (1-4/N)\sigma_l^2 & : \ i \neq j \text{ and tours } i \text{ and } j \text{ have } N-3 \text{ edges in common.} \end{cases}$$

We have now obtained all ingredients to express the probability that a tour is a local minimum, given its cost. However, the problem that remains is how to evaluate or approximate the probability integral (2).

A general method to evaluate a multidimensional normal probability is by a linear transformation of the random variables, such that the resulting variables are independent. However, after such a transformation the dependencies typically return in the region of integration.

Several other methods are known in literature to evaluate a multidimensional probability integral[Tong, 1990]. These methods are based on special properties of the covariance matrix, which do not apply to the covariance matrix induced by the neighborhood structure $\mathcal{N}_2$.

3.3 The restricted neighborhood $\mathcal{N}_2'$

In the previous section we derived an expression for the density of the costs of local minima, which, however, we can not be evaluated numerically. To validate our model, we introduce therefore the restricted neighborhood structure $\mathcal{N}_2'$. In this neighborhood structure, the neighbors of a tour are a subset of its neighbors in $\mathcal{N}_2$. This subset is chosen in such a way that the density of the cost of local minima (of $\mathcal{N}_2'$) can be computed numerically. Formally, the neighborhood structure $\mathcal{N}_2'$ is defined as follows.

Definition 3.5. Let $t \in S$. Then, the neighborhood $\mathcal{N}_2'(t)$ is given by $\{2\text{-}change(t, 0, s) \mid 1 < s < N-1\}$. □

It follows that each tour has $N-3$ neighbors. The key property of this neighborhood structure is that all neighbors of a tour have $N-3$ edges in common, which allows us to express the density of the costs of local minima of $\mathcal{N}_2'$.

Just as in the previous section, this problem reduces to calculating the probability that a tour is a local minimum, given its cost. Let t_0 again be a randomly chosen tour with cost c_0, and let its neighbors have cost $c_1, \ldots, c_{N-3}$. Applying the same strategy as in the previous section we find that the conditional density $\omega(c_1, \ldots, c_{N-3} \mid c_0)$ is again jointly normal, and is given by

$$\omega(c_1, \ldots, c_{N-3} \mid c_0) = \frac{1}{(2\pi)^{(N-3)/2}\|\mathbf{R}'\|}\exp[-\frac{1}{2}(\mathbf{c}-\boldsymbol{\mu}')\mathbf{R}'^{-1}(\mathbf{c}-\boldsymbol{\mu}')^{\mathbf{T}}]. \tag{3}$$

The conditional means μ_i' are the same as in the previous section, that is

$$\mu_i' = (1-2/N)c_0 + (2/N)\mu.$$

Since all pairs of neighbors have $N-3$ edges in common, the elements in the conditional covariance matrix $\mathbf{R}' = [r_{ij}']$ are

$$r_{ij}' = \begin{cases} (4-4/N)\sigma_l^2 & : \quad i = j, \text{ the conditional variance.} \\ (1-4/N)\sigma_l^2 & : \quad i \neq j, \text{ the conditional covariance.} \end{cases}$$

The essential observation is that all covariances are equal and positive. In this case, the joint density (3) can be simplified, as is shown in [Ihm, 1959; Marsaglia, 1963]. In the remainder of this section we show how this is done. First, we introduce

$$y_i = c_i - \mu_i' \quad \text{for } 1 \leq i \leq N-3.$$

Then, the means become zero and, trivially, density (3) can be rewritten to

$$\omega(y_1, \ldots, y_{N-3} \mid c_0) = \frac{1}{(2\pi)^{(N-3)/2}\|\mathbf{R}'\|}\exp[-\frac{1}{2}\mathbf{y}\mathbf{R}'^{-1}\mathbf{y}^{\mathbf{T}}]. \tag{4}$$

The central observation in this method is that each random variable y_i can be represented as the sum of two random variables, that is,

$$y_i = \sigma_l\sqrt{1-4/N}x_0 + \sigma_l\sqrt{3}x_i \quad \text{for } 1 \leq i \leq N-3\,.$$

Here, $x_0, \ldots, x_{N-3}$ are independent, standardized, normally distributed random variables. The random variables y_i again have a multidimensional normal distribution, with zero means, variances $(4-4/N)\sigma_l^2$, and covariances $(1-4/N)\sigma_l^2$.

Because the random variables x_i are independent, we can, by a generalization of the convolution theorem, express density (4) as

$$\int_{-\infty}^{\infty} \omega(x_0)\omega\left(\frac{y_1 - \sigma_l\sqrt{1-4/N}x_0}{\sigma_l\sqrt{3}} \mid x_0\right)\cdots\omega\left(\frac{y_{N-3} - \sigma_l\sqrt{1-4/N}x_0}{\sigma_l\sqrt{3}} \mid x_0\right)dx_0.$$

Now, we are able to express the probability that t_0 is a local minimum, given that its cost is c_0. Since the random variables x_i are identically distributed, we have

$$\mathbb{P}\{t_0 \text{ is a local minimum} \mid c_0\} = \int_{-\infty}^{\infty} \omega(x_0)\left[\int_{c_0-\mu'}^{\infty} \omega(\frac{y_1 - \sigma_l\sqrt{1-4/N}x_0}{\sigma_l\sqrt{3}} \mid x_0)dy_1\right]^{N-3} dx_0. \tag{5}$$

The inner integral in (5) can be rewritten to

$$\frac{1}{2}\text{erfc}\left(\frac{\sqrt{2}(c_0-\mu)}{\sqrt{3N}\sigma} - \frac{x_0\sqrt{1-4/N}}{\sqrt{6}}\right).$$

So, the probability that t_0 is a local minimum, given that its cost is c_0, is equal to

$$(6) \qquad \frac{1}{2^{N-3}\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp(-s^2/2) \left[\operatorname{erfc}\left(\frac{\sqrt{2}(c_0 - \mu)}{\sqrt{3N}\sigma} - \frac{s\sqrt{1-4/N}}{\sqrt{6}} \right) \right]^{N-3} ds.$$

Expression (6) can be evaluated numerically.

3.4 An analysis of first improvement algorithms

In this section we outline a simple method to analyze the average-case performance of first improvement algorithms, based on ideas described by Weinberger [1991] and Nakano & Nakanishi [1983]. A first improvement algorithm repeatedly moves to the firstly found neighboring solution with lower cost, until no more neighbors with lower cost are found. As such, an execution of a first improvement algorithm can be represented as a path in the neighborhood graph.

In this section we describe how these paths can be modeled statistically, and how this model can be used to analyze the running time and the distribution of final solutions of first improvement algorithms that use the neighborhood structure $\mathcal{N}_2'$.

First, we describe how a single transition can be modeled. For this, we define the transition probability $P(c_i, c_{i+1})$, that is, the probability that a solution t_i with cost c_i found after i iterations is improved to a solution t_{i+1} with cost c_{i+1}. Hence,

$$P(c_i, c_{i+1}) = \mathbb{P}\{\underline{f}(t_{i+1}) = c_{i+1} \mid \underline{f}(t_i) = c_i \}.$$

In order to express the transition probability $P(c_i, c_{i+1})$, we consider first that if tour t_i has cost c_i, we have two possibilities. The first possibility is that tour t_i is a local minimum, in which case no improvement is made. The second possibility is that tour t_i is not a local minimum. Then, we still have to consider the probability that solution t_i with cost c_i moves to a neighboring solution t_{i+1} with cost c_{i+1}, given that t_i is not a local minimum. If we make the simple assumption that the neighbors are conditionally independent and that the transition is made to a randomly chosen neighbor with lower cost, we can approximate this probability as

$$\mathbb{P}\{\underline{f}(t_{i+1}) = c_{i+1} \mid \underline{f}(t_i) = c_i \wedge c_{i+1} < c_i\} = \frac{\mathbb{P}\{\underline{f}(t_{i+1}) = c_{i+1} \wedge c_{i+1} < c_i \mid \underline{f}(t_i) = c_i \}}{\mathbb{P}\{\underline{f}(t_{i+1}) < \underline{f}(t_i) \mid \underline{f}(t_i) = c_i \}}.$$

It follows from Theorem 3.3 and the replacement of 2 edges by a 2-change that the cost c_{i+1} of t_{i+1} is drawn from a normal distribution, with mean and variance

$$\begin{aligned} \mu_i &= \mu + (1 - 2/N)(c_i - \mu), \\ \sigma_i^2 &= (4 - 4/N)\sigma_i^2 \end{aligned}$$

and density

$$\omega(c_{i+1} \mid c_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp[-(c_{i+1} - \mu_i)^2/2\sigma_i^2].$$

Then, we can easily express

$$\begin{aligned} \mathbb{P}\{\underline{f}(t_{i+1}) < \underline{f}(t_i) \mid \underline{f}(t_i) = c_i \} &= \int_{-\infty}^{c_i} \omega(c' \mid c_i) dc' \\ &= \frac{1}{2} \operatorname{erf}[(c_i - \mu_i)/\sqrt{2}\sigma_i], \end{aligned}$$

and

$$\mathbb{P}\{\underline{f}(t_{i+1}) = c_{i+1} \wedge c_{i+1} < c_i \mid \underline{f}(t_i) = c_i\} = \begin{cases} \omega(c_{i+1} \mid c_i) & : \text{ if } c_{i+1} < c_i. \\ 0 & : \text{ otherwise.} \end{cases}$$

Consequently, we can express the transition probability $P(c_i, c_{i+1})$ as

(7)

$$P(c_i, c_{i+1}) = \begin{cases} \dfrac{2\omega(c_{i+1} \mid c_i)}{\operatorname{erf}[(c_i - \mu_i)/\sqrt{2}\sigma_i]}(1 - \mathbb{P}\{t_i \text{ is loc. min.} \mid f(t_i) = c_i\}) & : \text{ if } c_{i+1} \leq c_i. \\ 0 & : \text{ otherwise.} \end{cases}$$

The distribution of final solutions. Next, we show how transition probability (7) can be used to express the distribution of the cost of local minima obtained by a first improvement algorithm.

For this, we need an additional quantity $l(c)$, which represents the probability that a solution with cost c is reached during an execution of a first improvement algorithm. Since a first improvement algorithm repeatedly moves to a neighbor with lower cost, a solution with cost c can be reached either initially, or by a transition from a solution with higher cost. Hence, if we let $\omega(c)$ be the density of the tour lengths, $l(c)$ can be recursively expressed as

$$l(c) = \omega(c) + \int_c^\infty l(c')P(c', c)dc'. \tag{8}$$

Then, the probability that the algorithm finds a final solution with cost c equals the probability that a local minimum has been found once a solution with cost c is reached. Thus, the density $\omega_{fi}(c)$ of the costs of the final solutions obtained by a first improvement algorithm can be expressed as

$$\omega_{fi}(c) = l(c) \cdot \mathbb{P}\{t \text{ is a local minimum} \mid \underline{f}(t) = c\}. \tag{9}$$

The running time of first improvement algorithms. Next, we propose a simple approach to approximate the average running time of first improvement algorithms. As a measure for the running time we use the number of *accepted* neighbors, i.e., the number of 2-changes effectuated, even though, strictly seen, the number of *proposed* neighbors, should be used to represent the running time. However, in order to model the intermediate states of the algorithm the number of accepted neighbors is more useful.

The state of a first improvement algorithm can be characterized by the probability $p_i(c)$, i.e., the probability that the solution has cost c after i improvements. Similar to the approach outlined above, we can recursively express the probability $p_i(c)$ as

$$\begin{aligned} p_0(c) &= \frac{1}{\sqrt{2\pi}\sigma}\exp[(c - \mu)^2/2\sigma^2], \\ p_{i+1}(c) &= \int_c^\infty p_i(c_i)P(c_i, c)dc_i. \end{aligned}$$

Here, we assume that the first improvement algorithm starts from randomly chosen initial solution. Since a first improvement algorithm stops if a local minimum is found, we can express the

probability that the algorithm terminates after i improvements as

$$\mathbb{P}\{t_i \text{ is a local minimum}\} = \int_{-\infty}^{\infty} \mathbb{P}\{t_i \text{ is a local minimum} \mid \underline{f}(t_i) = c\} p_i(c) dc.$$

Hence, the average number of of iterations needed by a first improvement algorithm can be expressed as

$$\mathbb{E}[i] = \sum_{i=0}^{\infty} i \int_{-\infty}^{\infty} \mathbb{P}\{t_i \text{ is a local minimum} \mid \underline{f}(t_i) = c\} p_i(c) dc. \tag{10}$$

3.5 Empirical results

We aim to validate the theoretically obtained results from the previous sections. A problem, however, is to empirically obtain the distribution of the cost of local minima. The correct method is to take a random sample of the solutions, and consider the local minima in this sample. Unfortunately, the sample size would have to be extremely large, because the probability that a solution is locally minimal is very small. So, this method is not feasible. Therefore, we only compare empirical distributions of final solutions with their theoretical predictions, discussed in Section 3.4. We consider instances in which edge lengths are distributed according to a normal distribution, which agree best with our assumptions.

The results are shown in Figure 1. The cost is again normalized w.r.t. the mean and standard

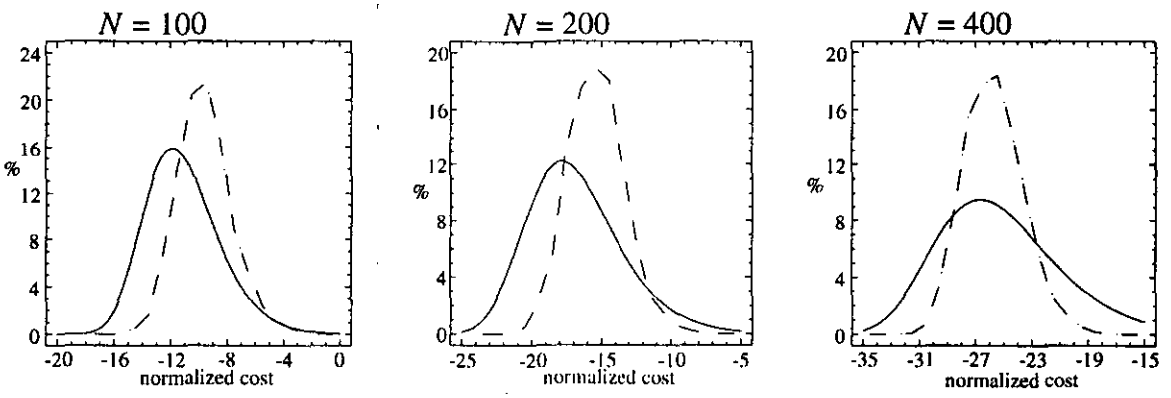

Figure 1: The distribution of final solutions.

deviation of the tour lengths. The solid curves represent the theoretically predicted distributions, the dashed curves the empirically obtained distributions.

We observe a reasonable agreement between the theoretically predicted and the empirically obtained curves, both in position and shape of the distributions. However, we see an increasing overestimation of the standard deviation as the size of the instance grows. This might be caused by our relatively simple approximation of the transition probability, for which we used the rather strong assumption that neighbors of a tour are conditionally independent.

Further evidence that the assumption of conditional independence is too strong can be found in Figure 2, which shows our results for the average running time. For this, we have performed 500 executions of a first improvement algorithm that employs the restricted neighborhood structure

$\mathcal{N}_2'$. We have used a random normal instance with 100 cities. During the executions, we kept track of the cost obtained at each improvement and of the number of improvements that were needed to reach a local minimum.

In the left figure the mean cost of the current solution is given as a function of the improvements that were made during the execution of the first improvement algorithm. The right figure shows the distribution of the number of iterations made by the algorithm. The solid curves represent the theoretically predicted results, the dashed curves the empirically obtained results. We observe

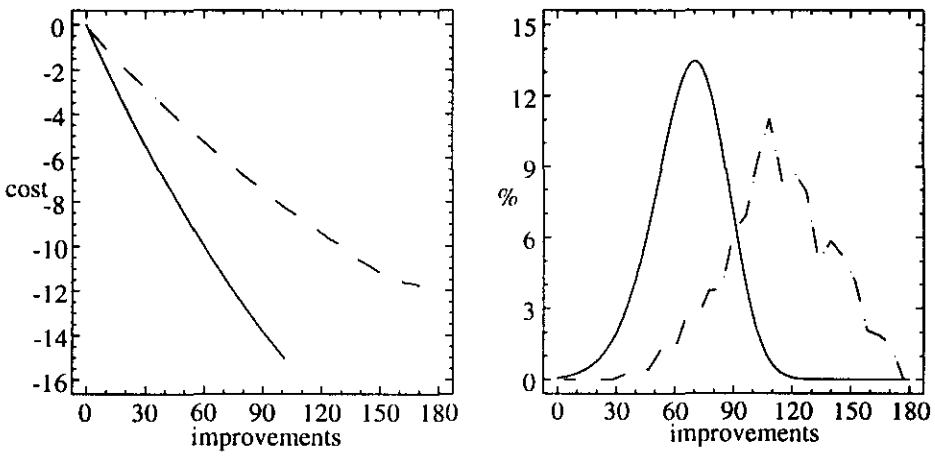

Figure 2: The average running time of first improvement algorithms.

in Figure 2 that already early during the execution, the mean cost of the current solution is underestimated. This could be further evidence that our assumption of conditional independence is too strong, since at that time the probability that a tour is a local minimum given its cost has little influence on the transition probability. From this result, it follows that we underestimate the average number of improvements made, as can be seen in the right figure.

Empirical results for first improvement 2-opt and Lin–Kernighan. Finally, we consider the distribution of the final solutions obtained by a first improvement 2-opt algorithm and a Lin–Kernighan algorithm. Figure 3 shows the empirical results for a number of real world instances, that is, instances derived from practical problems. The frequency is plotted against the relative deviation w.r.t. the optimal solution value.

For the 2-opt algorithm, the solid line represents the results for the kroB200 instance, the short-dashed line the results for the lin318 instance and the long-dashed line the results for the pcb442 instance. For the Lin–Kernighan algorithm, the solid line represents the results for the pcb442 instance, the short-dashed line the results for the u574 instance and the long-dashed line the results for the pr1002 instance.

The distributions of the final solutions obtained by these iterative improvement algorithms appear to have two interesting properties. First, we observe that the final solutions seem to be distributed according to a gamma distribution. We checked this by performing a chi-square test, which was accepted for all instances except the kroB200 instance. Other instances, not shown in Figure 3 also fitted to a gamma distribution.

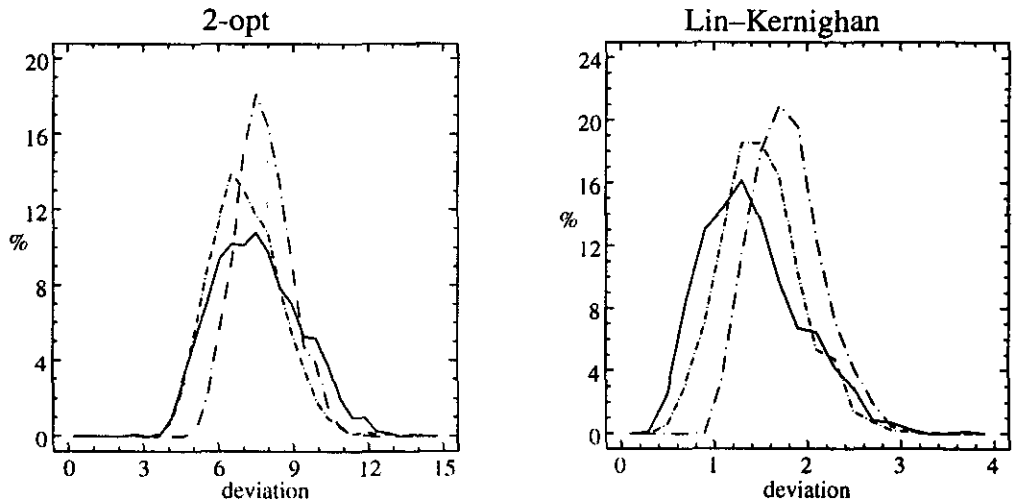

Figure 3: The distribution of final solutions obtained by a 2-opt and a Lin–Kernighan algorithm.

Secondly, we see that for both the 2-opt and the Lin–Kernighan algorithm, the standard deviation of the distribution decreases as the size of the instance grows. Again, the results obtained for other instances, not shown in Figure 3 agree with this property.

4 A semi-empirical analysis of iterated local search

The best known heuristic to handle the TSP is the iterated Lin–Kernighan algorithm of [Johnson, 1990]. In Johnson's iterated Lin–Kernighan algorithm a local minimum is modified by a 4-change, i.e., replacing four edges by four new edges, and used as starting solution for the next run of the Lin–Kernighan heuristic.

In this section we approximate the average computation time an iterated local search algorithm needs to find a solution within a given range from the optimal solution. Time is measured by the number of evaluations, i.e., the number of solutions examined by the algorithm. Our approximation is of a semi-empirical nature, which means that the average number of evaluations is expressed as a function of empirically obtained parameters, viz., the distribution of the final solutions and the average running time of the underlying iterative improvement algorithm.

To approximate the average running time we differentiate between the average number of iterations needed and the number of evaluations needed per iteration. Hence, we assume an execution of an iterated local search algorithm to be a sample of the local minima, which means that we neglect the intermediate solutions generated by the iterative improvement algorithm. As a consequence of this assumption, we can express the probability that a solution whose cost is lower than the predefined value ϵ has been found after i iterations, as

$$\mathbb{P}(i,\epsilon) = 1 - \prod_{k=1}^{i}(1 - \mathbb{P}\{f(s_k) \leq (1+\epsilon)f_{\text{opt}}\}), \tag{11}$$

with s_k be the final tour obtained at iteration k, $(1 \leq k \leq i)$.

In order to determine the average number of iterations needed, we assume that the modification mechanism is primarily a diversification mechanism. Hence, we consider subsequent local

minima to be independent, so that probability $\mathbb{P}\{f(s_k) \leq \epsilon\}$ is independent of k. It should be noted that this is rather a strong assumption, since the modification mechanism only changes four edges. Because we assume the density $\omega_{lm}(c)$ of the costs of local minima to be given, we then have

$$p(\epsilon) = \mathbb{P}\{f(s_k) \leq (1+\epsilon)f_{\text{opt}}\} = \int_{-\infty}^{(1+\epsilon)f_{\text{opt}}} \omega_{lm}(c)dc.$$

This allows us to express probability (11) as

$$\mathbb{P}(i, \epsilon) = 1 - (1 - p(\epsilon))^i.$$

This can be rewritten to

$$\mathbb{P}(i, \epsilon) = 1 - e^{-\lambda i}, \tag{12}$$

where

$$\lambda = -\ln(1 - p(\epsilon)).$$

Expression (12) then implies that the number of iterations is distributed according to an exponential distribution, so we can easily express the average number of iterations as

$$\mathbb{E}_\epsilon[i] = \frac{1}{\lambda}.$$

After the first iteration the iterative improvement algorithm starts with a tour that differs only four edges with a local optimum. Consequently, the average number of evaluations required per iteration decreases substantially from that required by the first iteration. In order to approximate the average number of evaluations needed per iteration, we therefore differentiate between two values:

- k_1, the average number of evaluations needed to reach a local minimum from a randomly chosen starting solution.
- k_r, the average number of evaluations needed to reach a local minimum that has been obtained by modifying a local minimum

The averages k_1 and k_r are obtained empirically. We have observed that the relative difference between k_1 and k_r increases as the size of the problem instance grows.

Using the above observations, the average number of evaluations needed to find a solution with a given relative error ϵ is approximated as

$$\mathbb{E}_\epsilon[k] = k_1 + (\frac{1}{\lambda} - 1)k_r.$$

4.1 Empirical results

We have analyzed iterated 2-opt and Lin–Kernighan algorithms with a 4-change as modification mechanism. Both algorithms have been tested on instances from Reinelt's TSPLIB. In order to acquire the average number of evaluations empirically, 100 executions of the iterated local search algorithm have been performed for each instance.

Figures 4 and 5 show the results of the iterated 2-opt algorithm for the instance kroB200, and the iterated Lin–Kernighan algorithm for the instance u574, respectively. Iterated local search for other instances, not shown in Figure 4 and 5, display a similar behavior. In these figures, the probability that a solution is found with the given relative error ϵ is depicted as function of the

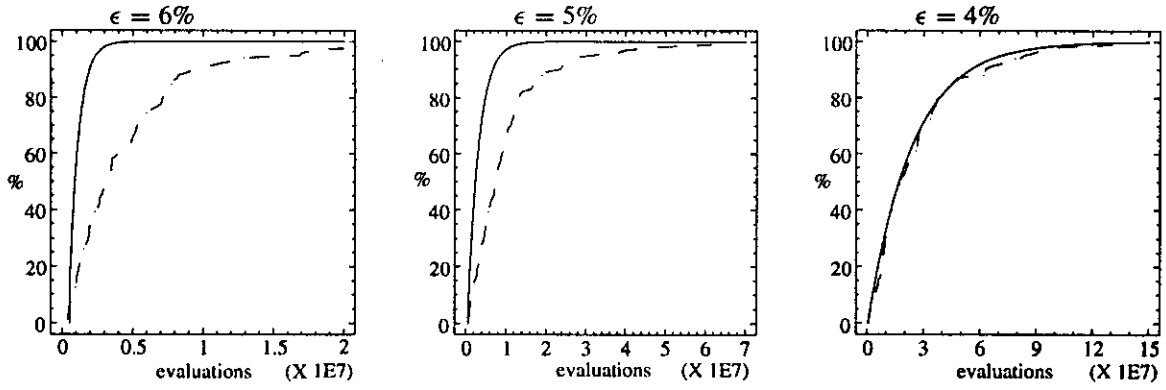

Figure 4: The average running time of an iterated 2-opt algorithm for KroB200.

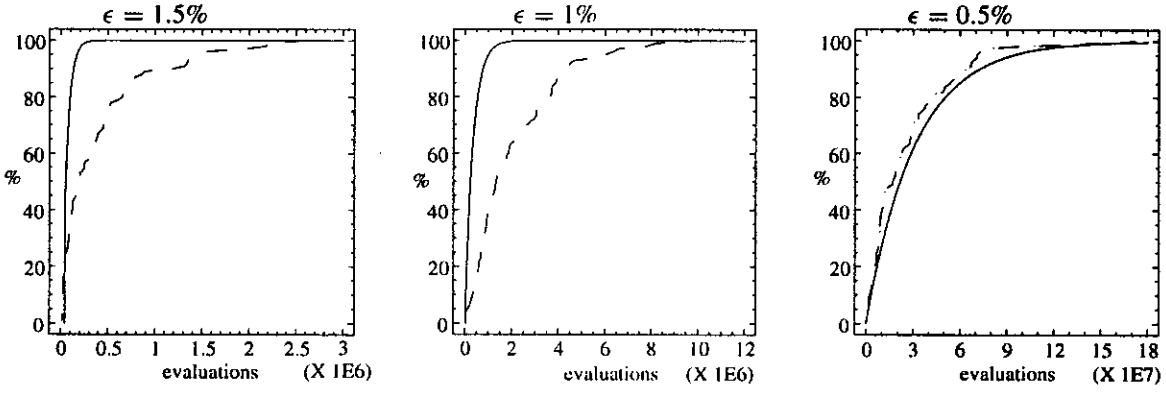

Figure 5: The average running time of an iterated Lin–Kernighan algorithm for u574.

number of evaluations. The solid curves represent the theoretically predicted distributions, the dashed curves the empirically obtained distributions.

We observe that the cumulative density fits well with an exponential distribution. A good agreement between our theoretical prediction and the empirical results is only obtained for low deviations ϵ. This is explained by the observation that even a small under- or overestimation of the probability that a tour is within the desired range ϵ, can have a large effect on the theoretical curves.

5 Conclusions

In this paper we have discussed the average-case performance, both with respect to quality of final solutions and running time, of the 2-opt algorithm for the Traveling Salesman Problem. We have used a probabilistic approach, in which we explicitly state the assumptions we made, to express the distribution of local minima for the restricted neighborhood structure $\mathcal{N}_2'$. Furthermore, we have outlined a simple method to approximate the distribution of the final solutions obtained by a first improvement algorithm that uses this neighborhood structure.

An empirical investigation of the distribution of final solutions obtained by a first improvement 2-opt and a Lin–Kernighan algorithm furthermore shows two interesting properties. First, we have observed that the local minima are distributed according to a gamma distribution and secondly, we have observed that the standard deviation of these distributions seems to decrease as the instance size grows. These characteristics are an important aspect of a neighborhood for iterated local search, or other more advanced local search algorithms, because they indicate the additional computational effort required by iterated local search to find lower-cost local minima.

References

AARTS, E.H.L., AND J.K. LENSTRA (eds.) [1995], *Local Search in Combinatorial Optimization*, Wiley.

BONOMI, E., AND J-L. LUTTON [1984], The N-city travelling salesman problem: statistical mechanics and the metropolis algorithm, *SIAM Review* **26**, 551–568.

CHANDRA, B., H. KARLOFF, AND C. TOVEY [1994], New Results on the old k-opt algorithm for the TSP, *Proc. of the 5th annual ACM SIAM Symposium on Discrete Algorithms*, 150–159.

IHM, P. [1959], Numerical evalutation of certain multivariate normal integrals, *Sankhya : The Indian Journal of Statistics* **21**, 363–366.

JOHNSON, D.S. [1990], Local optimization and the traveling salesman problem, *Proc. 17th Colloquium on Automata, Languages, and Programming*, LNCS 447, 446–461.

KERN, W. [1989], A probabilistic analysis of the switching algorithm for the Euclidian TSP, *Mathematical Programming* **44**, 213–219.

KIRKPATRICK, S., AND G. TOULOUSE [1985], Configuration space analysis of Travelling Salesman Problems, *Journal Physique* **46**, 1277–1292.

LIN, S. [1965], Computer solutions of the traveling salesman problem, *Bell System Technical Journal* **44**, 2245–2269.

LIN, S., AND B.W. KERNIGHAN [1973], An effective heuristic algorithm for the traveling salesman problem, *Operations Research* **21**, 498–516.

MARSAGLIA, G. [1963], Expressing the normal distribution with covariance matrix $a+b$ in terms of one with covariance matrix a, *Biometrika* **50**, 535–538.

NAKANO, H., AND Y. NAKANISHI [1983], An analysis of local neighborhood search method for combinatorial optimization problems, *Proc. of the IEEE Symposium on Circuits and Systems*, 1055–1058.

PAPOULIS, A. [1965], *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill.

STADLER, P.F., AND W. SCHNABL [1992], The landscape of the traveling salesman problem, *Physics Letters A* **161**, 337–344.

TONG, Y.L. [1990], *The Multivariate Normal Distribution*, Springer-Verlag.

TOVEY, C.A. [1985], Hill climbing with multiple local optima, *SIAM Journal of Discrete Mathematics* **6**, 383–393.

WEINBERGER, E.D. [1991], Local properties of Kauffman's N-k model: A tunably rugged energy landscape, *Physical Review A* **44**, 6399–6413.

Appendix

This appendix contains proofs of the theorems. First, we prove Theorem 3.4.

Theorem 3.4. *Let* $\epsilon = 2\frac{N}{N-1}\sigma_l^2$. *Then we have*

$$\begin{aligned}\mathbb{E}[\hat{\mu}] &= N\mu_l,\\ \mathbb{E}[\hat{\sigma}^2] &= N\sigma_l^2 - \epsilon,\\ \mathbb{E}[\hat{r}_k] &= (N-k)\sigma_l^2 - \epsilon.\end{aligned}$$

Proof. Define the set of edges $\mathcal{D}$ and the number of edges e as

$$\begin{aligned}\mathcal{D} &= \{(i,j)|0 \le i < j < N\}\\ e &= |\mathcal{D}|\\ &= N(N-1)/2\end{aligned}$$

Recall that for any $(i, j) \in \mathcal{D}$

$$\mu_l = \mathbb{E}[d_{ij}] \quad \text{and} \quad \sigma_l^2 = \mathbb{E}[d_{ij}^2] - \mathbb{E}^2[d_{ij}]$$

The proofs we present below are based on rewriting summations over solutions as summations over edges. For instance, it is not hard to see that

$$\frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} f(s) = \frac{N}{e}\sum_{(i,j)\in\mathcal{D}} d_{ij}. \tag{13}$$

Furthermore, we use our assumption of independence between edge lengths, that is,

$$(i,j) \neq (i',j') \Rightarrow \mathbb{E}[d_{ij}d_{i'j'}] = \mathbb{E}[d_{ij}]\mathbb{E}[d_{i'j'}]$$

Then we have

1. $\mathbb{E}[\hat{\mu}] = N\mu_l$.

$$\begin{aligned}\mathbb{E}[\hat{\mu}] &= \mathbb{E}\left[\frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} f(s)\right]\\ &= \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}}\mathbb{E}[f(s)]\\ &= \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}}\sum_{(i,j)\in s}\mathbb{E}[d_{ij}]\\ &= N\mu_l\end{aligned}$$

2. $\mathbb{E}[\hat{\sigma}^2] = N\sigma_l^2 - \epsilon$.

$$\begin{aligned}\mathbb{E}[\hat{\sigma}^2] &= \mathbb{E}\left[\frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} f^2(s) - \hat{\mu}^2\right]\\ &= \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}}\mathbb{E}[f^2(s)] - \mathbb{E}[\hat{\mu}^2] \qquad (14)\end{aligned}$$

We can rewrite the left term in (14) as follows.

$$\begin{aligned}\frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}}\mathbb{E}[f^2(s)] &= \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}}\left(\sum_{(i,j),(i',j')\in s}\mathbb{E}[d_{ij}d_{i'j'}]\right)\\ &= N\mathbb{E}[d^2]+(N^2-N)\mathbb{E}^2[d]\\ &= N\sigma_l^2+N^2\mu_l^2\end{aligned}$$

Using (13) we can rewrite the right term in (14) as

$$\begin{aligned}\mathbb{E}[\hat{\mu}^2] &= \mathbb{E}\left[\frac{N^2}{e^2}\left(\sum_{(i,j)\in D}d_{ij}\right)^2\right]\\ &= \frac{4}{(N-1)^2}\sum_{(i,j),(i',j')\in D}\mathbb{E}[d_{ij}d_{i'j'}]\\ &= \frac{4}{(N-1)^2}\left(e\mathbb{E}[d^2]+e(e-1)\mathbb{E}^2[d]\right)\\ &= 2\frac{N}{N-1}\mathbb{E}[d^2]+N^2\mathbb{E}^2[d]-2\frac{N}{N-1}\mathbb{E}^2[d]\\ &= N^2\mu_l^2+\epsilon \qquad (15)\end{aligned}$$

Thus,

$$\begin{aligned}\mathbb{E}[\hat{\sigma}^2] &= N\sigma_l^2+N^2\mu_l^2-(N^2\mu_l^2+\epsilon)\\ &= N\sigma_l^2-\epsilon\end{aligned}$$

3. $\mathbb{E}[\hat{r}_k]=(N-k)\sigma_l^2-\epsilon$.

$$\begin{aligned}\mathbb{E}[\hat{r}_k] &= \mathbb{E}\left[\frac{1}{|\mathcal{S}|\cdot|\mathcal{D}_k|}\sum_{s\in\mathcal{S}}\sum_{s'\in\mathcal{D}_k(s)}f(s)f(s')-\hat{\mu}^2\right]\\ &= \frac{1}{|\mathcal{S}|\cdot|\mathcal{D}_k|}\sum_{s\in\mathcal{S}}\sum_{s'\in\mathcal{D}_k(s)}\mathbb{E}[f(s)f(s')]-\mathbb{E}[\hat{\mu}^2] \qquad (16)\end{aligned}$$

We can rewrite the left term in (16) as follows.

$$\begin{aligned}\frac{1}{|\mathcal{S}|\cdot|\mathcal{D}_k|}\sum_{s\in\mathcal{S}}\sum_{s'\in\mathcal{D}_k(s)}\mathbb{E}[f(s)f(s')] &= \frac{1}{|\mathcal{S}|\cdot|\mathcal{D}_k|}\sum_{s\in\mathcal{S}}\sum_{s'\in\mathcal{D}_k(s)}\left(\sum_{(i,j)\in s,(i',j')\in s'}\mathbb{E}[d_{ij}d_{i'j'}]\right)\\ &= (N-k)\mathbb{E}[d^2]+(N^2-N+k)\mathbb{E}^2[d]\\ &= (N-k)\sigma_l^2+N^2\mu_l^2\end{aligned}$$

Hence, subtraction of (15) leads to,

$$\begin{aligned}\mathbb{E}[\hat{r}_k] &= (N-k)\sigma_l^2+N^2\mu_l^2-(N^2\mu_l^2+\epsilon)\\ &= (N-k)\sigma_l^2-\epsilon\end{aligned}$$

□

Next, we prove Theorem3.5.

Theorem 3.5. *The empirical quantities $\hat{\mu}$, $\hat{\sigma}^2$ and $\hat{r}_k$ are self-averaging.*

Proof.

1. $\hat{\mu}$ is self-averaging.
   We already derived in the previous theorem that $\mathbb{E}[\hat{\mu}^2] = N^2\mu_l^2 + \epsilon$. Hence,
   $$\begin{aligned}(\mathbb{E}[\hat{\mu}^2] - \mathbb{E}^2[\hat{\mu}])^{\frac{1}{2}} &= (N^2\mu_l^2 + \epsilon - N^2\mu_l^2)^{\frac{1}{2}} \\ &= \sqrt{\epsilon}\end{aligned}$$
   Since ϵ is nearly constant, it is not hard to see that
   $$\lim_{N\to\infty} \frac{(\mathbb{E}[\hat{\mu}^2] - \mathbb{E}^2[\hat{\mu}])^{\frac{1}{2}}}{\mathbb{E}[\hat{\mu}]} = \lim_{N\to\infty} \frac{\sqrt{\epsilon}}{N\mu_l} = 0$$
2. $\hat{\sigma}^2$ is self-averaging.
   We begin the proof with a calculation of $\mathbb{E}[(\hat{\sigma}^2)^2]$, for which we first rewrite $\hat{\sigma}^2$ as a summation over edges. We have
   $$\begin{aligned}\frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} f^2(s) &= \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}}\left(\sum_{(i,j)\in s} d_{ij}^2 + \sum_{\substack{(i,j)\neq(i',j')\\(i,j),(i',j')\in s}} d_{ij}d_{i'j'}\right) \\ &= \frac{N}{e}\sum_{(i,j)\in\mathcal{D}} d_{ij}^2 + \frac{N(N-1)}{e(e-1)}\sum_{\substack{(i,j)\neq(i',j')\\(i,j),(i',j')\in\mathcal{D}}} d_{ij}d_{i'j'}\end{aligned}$$
   and
   $$\hat{\mu}^2 = \frac{N^2}{e^2}\sum_{(i,j)\in\mathcal{D}} d_{ij}^2 + \frac{N^2}{e^2}\sum_{\substack{(i,j)\neq(i'j')\\(i,j),(i'j')\in\mathcal{D}}} d_{ij}d_{i'j'}$$
   Substitution of $\alpha = \frac{N}{e}$ and $\beta = \frac{N-1}{e-1}$ then leads to
   $$\hat{\sigma}^2 = \alpha(1-\alpha)\sum_{(i,j)\in\mathcal{D}} d_{ij}^2 + \alpha(\beta-\alpha)\sum_{\substack{(i,j)\neq(i'j')\\(i,j),(i'j')\in\mathcal{D}}} d_{ij}d_{i'j'}$$
   Now let
   $$A = \alpha(1-\alpha)\sum_{(i,j)\in\mathcal{D}} d_{ij}^2 \quad \text{and} \quad B = \alpha(\beta-\alpha)\sum_{\substack{(i,j)\neq(i'j')\\(i,j),(i'j')\in\mathcal{D}}} d_{ij}d_{i'j'}$$
   so that
   $$\hat{\sigma}^2 = A + B$$
   Then we have
   $$\mathbb{E}[(\hat{\sigma}^2)^2] = \mathbb{E}[A^2] + 2\mathbb{E}[AB] + \mathbb{E}[B^2]$$

Let us first calculate $\mathbb{E}[A^2]$

$$\begin{aligned}\mathbb{E}[A^2] &= \alpha^2(1-\alpha)^2\mathbb{E}\left[\left(\sum_{(i,j)\in\mathcal{D}} d_{ij}^2\right)^2\right] \\ &= \alpha^2(1-\alpha)^2\mathbb{E}\left[\sum_{(i,j)\in\mathcal{D}} d_{ij}^4\right] + \alpha^2(1-\alpha)^2\mathbb{E}\left[\sum_{\substack{(i,j)\neq(i'j') \\ (i,j),(i',j')\in\mathcal{D}}} d_{ij}^2 d_{i'j'}^2\right] \\ &= \alpha^2(1-\alpha)^2 e\mathbb{E}[d^4] + \alpha^2(1-\alpha)^2 e(e-1)\mathbb{E}^2[d^2]\end{aligned}$$

Since $\alpha^2(1-\alpha)^2 e = \mathcal{O}(1)$ and $\alpha^2(1-\alpha)^2 e(e-1) = N^2 + \mathcal{O}(N)$ we get

$$\mathbb{E}[A^2] = N^2\mathbb{E}^2[d^2] + \mathcal{O}(N)$$

Next, we calculate $\mathbb{E}[AB]$

$$\begin{aligned}\mathbb{E}[AB] &= \alpha^2(1-\alpha)(\beta-\alpha)\mathbb{E}\left[\left(\sum_{(i,j)\in\mathcal{D}} d_{ij}^2\right)\left(\sum_{\substack{(i,j)\neq(i',j') \\ (i,j),(i',j')\in\mathcal{D}}} d_{ij}d_{i'j'}\right)\right] \\ &= \alpha^2(1-\alpha)(\beta-\alpha)2e(e-1)\mathbb{E}[d^3]\mathbb{E}[d] \\ &\quad + \alpha^2(1-\alpha)(\beta-\alpha)(e-2)e(e-1)\mathbb{E}[d^2]\mathbb{E}^2[d]\end{aligned}$$

Using $\alpha^2(1-\alpha)(\beta-\alpha)2e(e-1) = \mathcal{O}(1)$
and $\alpha^2(1-\alpha)(\beta-\alpha)(e-2)e(e-1) = -N^2 + \mathcal{O}(N)$ we obtain

$$\mathbb{E}[AB] = -N^2\mathbb{E}^2[d]\mathbb{E}[d^2] + \mathcal{O}(N)$$

Finally, we calculate $\mathbb{E}[B^2]$

$$\begin{aligned}\mathbb{E}[B^2] &= \alpha^2(\beta-\alpha)^2\mathbb{E}\left[\left(\sum_{\substack{(i,j)\neq(i',j') \\ (i,j),(i',j')\in\mathcal{D}}} d_{ij}d_{i'j'}\right)^2\right] \\ &= \alpha^2(\beta-\alpha)^2 2e(e-1)\mathbb{E}^2[d^2] \\ &\quad + \alpha^2(\beta-\alpha)^2(4e-6)e(e-1)\mathbb{E}[d^2]\mathbb{E}^2[d] \\ &\quad + \alpha^2(\beta-\alpha)^2(e-4)(e-1)e(e-1)\mathbb{E}^4[d]\end{aligned}$$

Straightforward calculation leads to

$$\mathbb{E}[B^2] = N^2\mathbb{E}^4[d] + \mathcal{O}(N)$$

Consequently, we find that

$$\mathbb{E}[(\hat{\sigma}^2)^2] = N^2(\mathbb{E}[d^2] - \mathbb{E}^2[d])^2 + \mathcal{O}(N)$$

Subtraction of

$$\mathbb{E}^2[\hat{\sigma}^2] = N^2(\mathbb{E}[d^2] - \mathbb{E}^2[d])^2 + \mathcal{O}(N)$$

yields

$$\mathbb{E}[(\hat{\sigma}^2)^2] - \mathbb{E}^2[\hat{\sigma}^2] = \mathcal{O}(N)$$

Hence,

$$\lim_{N\to\infty} \frac{(\mathbb{E}[(\hat{\sigma}^2)^2] - \mathbb{E}^2[\hat{\sigma}^2])^{\frac{1}{2}}}{N\mathbb{E}[\hat{\sigma}^2]} = 0$$

3. $\hat{r}_k$ is self-averaging.
   This proof is analogous to the proof that the instance variance is self-averaging.

□

Theorem 3.6. *Let $(\mathcal{S}, f)$ be an instance of the TSP. Define $LM(f)$ to be the number of local minima in S and $LM(f, c)$ to be the number of local minima in S with cost c. Let t be a randomly chosen tour, and $\omega(c)$ the density of t in the instance class. Then,*

$$\begin{aligned} \mathbb{E}[LM(f)] &= |\mathcal{S}| \cdot \mathbb{P}\{t \text{ is a local minimum}\}, \\ \mathbb{E}[LM(f,c)] &= |\mathcal{S}| \cdot \mathbb{P}\{t \text{ is a local minimum} \mid c\}\omega(c). \end{aligned}$$

Proof. Let all solutions in $\mathcal{S}$ be consecutively numbered. Then, we can define

$$\chi_i(f) = \begin{cases} 1 & \text{if solution } i \text{ is a local minimum in instance } f \\ 0 & \text{else} \end{cases}$$

$$\chi_i(f,c) = \begin{cases} 1 & \text{if solution } i \text{ has cost } c \text{ and is a local minimum in instance } f \\ 0 & \text{else} \end{cases}$$

We use $\omega(f)$ to represent the joint density of all distances in the instance class. Consequently we have

$$\begin{aligned} LM(f) &= \chi_1(f) + \cdots + \chi_{|\mathcal{S}|}(f) \\ LM(f,c) &= \chi_1(f,c) + \cdots + \chi_{|\mathcal{S}|}(f,c) \end{aligned}$$

Furthermore, it is not difficult to see that

$$\int \omega(f)\chi_i(f)df = \mathbb{P}\{\text{tour } i \text{ is a local minimum}\}$$

holds for any i and is, in fact, independent of i. Then, we prove

1. $\mathbb{E}[LM(f)] = |\mathcal{S}| \cdot \mathbb{P}\{\text{t is a local minimum}\}$
   It is easy to see that

$$\begin{aligned} \mathbb{E}[LM(f)] &= \int (\chi_1(f) + \cdots + \chi_{|\mathcal{S}|}(f))\omega(f)df \\ &= \int \chi_1(f)\omega(f)df + \cdots + \int \chi_{|\mathcal{S}|}(f)\omega(f)df \\ &= |\mathcal{S}| \cdot \mathbb{P}\{\text{t is a local minimum}\} \end{aligned}$$

2. $\mathbb{E}[LM(f,c)] = |\mathcal{S}| \cdot \mathbb{P}\{\text{t is a local minimum} \mid c\}\omega(c)$
   Analogous to the proof of 1.

□

Next, we consider Conjecture 3.1.

Conjecture 3.1. *Let (S, f) be an instance of the TSP, and let $LM(f)$ and $LM(f, c)$ be as defined in Theorem 3.6. Let t be a randomly chosen tour. Then, we conjecture that*

$$\mathbf{E}\left[\frac{LM(f,c)}{LM(f)}\right] = \omega(c|t \text{ is a local minimum}).$$

By definition, we have for the left term

$$\mathbf{E}\left[\frac{LM(f,c)}{LM(f)}\right] = \int \frac{(\chi_1(f,c) + \cdots + \chi_{|S|}(f,c))}{(\chi_1(f) + \cdots + \chi_{|S|}(f))}\omega(f)df$$

However, for the right term we have

$$\omega(c|t \text{ is a local minimum}) = \frac{\int(\chi_1(f,c) + \cdots + \chi_{|S|}(f,c))\omega(f)df}{\int(\chi_1(f) + \cdots + \chi_{|S|}(f))\omega(f)df}$$

A proof that both terms are equal could consist of showing that $LM(f)$ is more or less a constant. Possibly, this might be done by proving that $LM(f)$ is self-averaging. □

In this series appeared:

93/01	R. van Geldrop	Deriving the Aho-Corasick algorithms: a case study into the synergy of programming methods, p. 36.
93/02	T. Verhoeff	A continuous version of the Prisoner's Dilemma, p. 17
93/03	T. Verhoeff	Quicksort for linked lists, p. 8.
93/04	E.H.L. Aarts J.H.M. Korst P.J. Zwietering	Deterministic and randomized local search, p. 78.
93/05	J.C.M. Baeten C. Verhoef	A congruence theorem for structured operational semantics with predicates, p. 18.
93/06	J.P. Veltkamp	On the unavoidability of metastable behaviour, p. 29
93/07	P.D. Moerland	Exercises in Multiprogramming, p. 97
93/08	J. Verhoosel	A Formal Deterministic Scheduling Model for Hard Real-Time Executions in DEDOS, p. 32.
93/09	K.M. van Hee	Systems Engineering: a Formal Approach Part I: System Concepts, p. 72.
93/10	K.M. van Hee	Systems Engineering: a Formal Approach Part II: Frameworks, p. 44.
93/11	K.M. van Hee	Systems Engineering: a Formal Approach Part III: Modeling Methods, p. 101.
93/12	K.M. van Hee	Systems Engineering: a Formal Approach Part IV: Analysis Methods, p. 63.
93/13	K.M. van Hee	Systems Engineering: a Formal Approach Part V: Specification Language, p. 89.
93/14	J.C.M. Baeten J.A. Bergstra	On Sequential Composition, Action Prefixes and Process Prefix, p. 21.
93/15	J.C.M. Baeten J.A. Bergstra R.N. Bol	A Real-Time Process Logic, p. 31.
93/16	H. Schepers J. Hooman	A Trace-Based Compositional Proof Theory for Fault Tolerant Distributed Systems, p. 27
93/17	D. Alstein P. van der Stok	Hard Real-Time Reliable Multicast in the DEDOS system, p. 19.
93/18	C. Verhoef	A congruence theorem for structured operational semantics with predicates and negative premises, p. 22.
93/19	G-J. Houben	The Design of an Online Help Facility for ExSpect, p.21.
93/20	F.S. de Boer	A Process Algebra of Concurrent Constraint Programming, p. 15.

93/21	M. Codish D. Dams G. Filé M. Bruynooghe	Freeness Analysis for Logic Programs - And Correctness?, p. 24.
93/22	E. Poll	A Typechecker for Bijective Pure Type Systems, p. 28.
93/23	E. de Kogel	Relational Algebra and Equational Proofs, p. 23.
93/24	E. Poll and Paula Severi	Pure Type Systems with Definitions, p. 38.
93/25	H. Schepers and R. Gerth	A Compositional Proof Theory for Fault Tolerant Real-Time Distributed Systems, p. 31.
93/26	W.M.P. van der Aalst	Multi-dimensional Petri nets, p. 25.
93/27	T. Kloks and D. Kratsch	Finding all minimal separators of a graph, p. 11.
93/28	F. Kamareddine and R. Nederpelt	A Semantics for a fine λ-calculus with de Bruijn indices, p. 49.
93/29	R. Post and P. De Bra	GOLD, a Graph Oriented Language for Databases, p. 42.
93/30	J. Deogun T. Kloks D. Kratsch H. Müller	On Vertex Ranking for Permutation and Other Graphs, p. 11.
93/31	W. Körver	Derivation of delay insensitive and speed independent CMOS circuits, using directed commands and production rule sets, p. 40.
93/32	H. ten Eikelder and H. van Geldrop	On the Correctness of some Algorithms to generate Finite Automata for Regular Expressions, p. 17.
93/33	L. Loyens and J. Moonen	ILIAS, a sequential language for parallel matrix computations, p. 20.
93/34	J.C.M. Baeten and J.A. Bergstra	Real Time Process Algebra with Infinitesimals, p.39.
93/35	W. Ferrer and P. Severi	Abstract Reduction and Topology, p. 28.
93/36	J.C.M. Baeten and J.A. Bergstra	Non Interleaving Process Algebra, p. 17.
93/37	J. Brunekreef J-P. Katoen R. Koymans S. Mauw	Design and Analysis of Dynamic Leader Election Protocols in Broadcast Networks, p. 73.
93/38	C. Verhoef	A general conservative extension theorem in process algebra, p. 17.
93/39	W.P.M. Nuijten E.H.L. Aarts	Job Shop Scheduling by Constraint Satisfaction, p. 22.

	D.A.A. van Erp Taalman Kip K.M. van Hee	
93/40	P.D.V. van der Stok M.M.M.P.J. Claessen D. Alstein	A Hierarchical Membership Protocol for Synchronous Distributed Systems, p. 43.
93/41	A. Bijlsma	Temporal operators viewed as predicate transformers, p. 11.
93/42	P.M.P. Rambags	Automatic Verification of Regular Protocols in P/T Nets, p. 23.
93/43	B.W. Watson	A taxomomy of finite automata construction algorithms, p. 87.
93/44	B.W. Watson	A taxonomy of finite automata minimization algorithms, p. 23.
93/45	E.J. Luit J.M.M. Martin	A precise clock synchronization protocol,p.
93/46	T. Kloks D. Kratsch J. Spinrad	Treewidth and Patwidth of Cocomparability graphs of Bounded Dimension, p. 14.
93/47	W. v.d. Aalst P. De Bra G.J. Houben Y. Kornatzky	Browsing Semantics in the "Tower" Model, p. 19.
93/48	R. Gerth	Verifying Sequentially Consistent Memory using Interface Refinement, p. 20.
94/01	P. America M. van der Kammen R.P. Nederpelt O.S. van Roosmalen H.C.M. de Swart	The object-oriented paradigm, p. 28.
94/02	F. Kamareddine R.P. Nederpelt	Canonical typing and Π-conversion, p. 51.
94/03	L.B. Hartman K.M. van Hee	Application of Marcov Decision Processe to Search Problems, p. 21.
94/04	J.C.M. Baeten J.A. Bergstra	Graph Isomorphism Models for Non Interleaving Process Algebra, p. 18.
94/05	P. Zhou J. Hooman	Formal Specification and Compositional Verification of an Atomic Broadcast Protocol, p. 22.
94/06	T. Basten T. Kunz J. Black M. Coffin D. Taylor	Time and the Order of Abstract Events in Distributed Computations, p. 29.
94/07	K.R. Apt R. Bol	Logic Programming and Negation: A Survey, p. 62.

94/08	O.S. van Roosmalen	A Hierarchical Diagrammatic Representation of Class Structure, p. 22.
94/09	J.C.M. Baeten J.A. Bergstra	Process Algebra with Partial Choice, p. 16.
94/10	T. verhoeff	The testing Paradigm Applied to Network Structure. p. 31.
94/11	J. Peleska C. Huizing C. Petersohn	A Comparison of Ward & Mellor's Transformation Schema with State- & Activitycharts, p. 30.
94/12	T. Kloks D. Kratsch H. Müller	Dominoes, p. 14.
94/13	R. Seljée	A New Method for Integrity Constraint checking in Deductive Databases, p. 34.
94/14	W. Peremans	Ups and Downs of Type Theory, p. 9.
94/15	R.J.M. Vaessens E.H.L. Aarts J.K. Lenstra	Job Shop Scheduling by Local Search, p. 21.
94/16	R.C. Backhouse H. Doornbos	Mathematical Induction Made Calculational, p. 36.
94/17	S. Mauw M.A. Reniers	An Algebraic Semantics of Basic Message Sequence Charts, p. 9.
94/18	F. Kamareddine R. Nederpelt	Refining Reduction in the Lambda Calculus, p. 15.
94/19	B.W. Watson	The performance of single-keyword and multiple-keyword pattern matching algorithms, p. 46.
94/20	R. Bloo F. Kamareddine R. Nederpelt	Beyond β-Reduction in Church's $\lambda\rightarrow$, p. 22.
94/21	B.W. Watson	An introduction to the Fire engine: A C++ toolkit for Finite automata and Regular Expressions.
94/22	B.W. Watson	The design and implementation of the FIRE engine: A C++ toolkit for Finite automata and regular Expressions.
94/23	S. Mauw and M.A. Reniers	An algebraic semantics of Message Sequence Charts, p. 43.
94/24	D. Dams O. Grumberg R. Gerth	Abstract Interpretation of Reactive Systems: Abstractions Preserving $\forall$CTL*, $\exists$CTL* and CTL*, p. 28.
94/25	T. Kloks	$K_{1,3}$-free and W_4-free graphs, p. 10.
94/26	R.R. Hoogerwoord	On the foundations of functional programming: a programmer's point of view, p. 54.

94/27	S. Mauw and H. Mulder		Regularity of BPA-Systems is Decidable, p. 14.
94/28	C.W.A.M. van Overveld M. Verhoeven		Stars or Stripes: a comparative study of finite and transfinite techniques for surface modelling, p. 20.
94/29	J. Hooman		Correctness of Real Time Systems by Construction, p. 22.
94/30	J.C.M. Baeten J.A. Bergstra Gh. Ştefanescu		Process Algebra with Feedback, p. 22.
94/31	B.W. Watson R.E. Watson		A Boyer-Moore type algorithm for regular expression pattern matching, p. 22.
94/32	J.J. Vereijken		Fischer's Protocol in Timed Process Algebra, p. 38.
94/33	T. Laan		A formalization of the Ramified Type Theory, p.40.
94/34	R. Bloo F. Kamareddine R. Nederpelt		The Barendregt Cube with Definitions and Generalised Reduction, p. 37.
94/35	J.C.M. Baeten S. Mauw		Delayed choice: an operator for joining Message Sequence Charts, p. 15.
94/36	F. Kamareddine R. Nederpelt		Canonical typing and Π-conversion in the Barendregt Cube, p. 19.
94/37	T. Basten R. Bol M. Voorhoeve		Simulating and Analyzing Railway Interlockings in ExSpect, p. 30.
94/38	A. Bijlsma C.S. Scholten		Point-free substitution, p. 10.
94/39	A. Blokhuis T. Kloks		On the equivalence covering number of splitgraphs, p. 4.
94/40	D. Alstein		Distributed Consensus and Hard Real-Time Systems, p. 34.
94/41	T. Kloks D. Kratsch		Computing a perfect edge without vertex elimination ordering of a chordal bipartite graph, p. 6.
94/42	J. Engelfriet J.J. Vereijken		Concatenation of Graphs, p. 7.
94/43	R.C. Backhouse M. Bijsterveld		Category Theory as Coherently Constructive Lattice Theory: An Illustration, p. 35.
94/44	E. Brinksma R. Gerth W. Janssen S. Katz M. Poel C. Rump	J. Davies S. Graf B. Jonsson G. Lowe A. Pnueli J. Zwiers	Verifying Sequentially Consistent Memory, p. 160
94/45	G.J. Houben		Tutorial voor de ExSpect-bibliotheek voor "Administratieve Logistiek", p. 43.

94/46	R. Bloo F. Kamareddine R. Nederpelt	The λ-cube with classes of terms modulo conversion, p. 16.
94/47	R. Bloo F. Kamareddine R. Nederpelt	On Π-conversion in Type Theory, p. 12.
94/48	Mathematics of Program Construction Group	Fixed-Point Calculus, p. 11.
94/49	J.C.M. Baeten J.A. Bergstra	Process Algebra with Propositional Signals, p. 25.
94/50	H. Geuvers	A short and flexible proof of Strong Normalazation for the Calculus of Constructions, p. 27.
94/51	T. Kloks D. Kratsch H. Müller	Listing simplicial vertices and recognizing diamond-free graphs, p. 4.
94/52	W. Penczek R. Kuiper	Traces and Logic, p. 81
94/53	R. Gerth R. Kuiper D. Peled W. Penczek	A Partial Order Approach to Branching Time Logic Model Checking, p. 20.
95/01	J.J. Lukkien	The Construction of a Small Communication Library, p. 16.
95/02	M. Bezem R. Bol J.F. Groote	Formalizing Process Algebraic Verifications in the Calculus of Constructions, p. 49.
95/03	J.C.M. Baeten C. Verhoef	Concrete process algebra, p. 134.