An analysis of Random Restart and Iterated Improvement for Global Optimization with an application to the Traveling Salesman Problem

F. Mendivil[1], R. Shonkwiler[2], and M. C. Spruill[2]

Abstract.

The optimization method employing iterated improvement with random restart (I2R2) is studied. Associated with each instance of an I2R2 search is a fundamental polynomial, $f(x) = p_0x + p_1x^2 + \ldots + p_dx^{d+1} - 1$, in which the coefficient p_k is the probability of starting a search k improvement steps from a local minimum. The positive root η of f can be used to calculate the convergence and speedup properties of that instance.

Since the coefficients of f are naturally related to the search, it is possible to estimate them online if an a priori estimate of the size θ of the goal basin is available, for example by analysis or prior experience. In this case, the runtime statistical estimate of η converges many times faster than the estimates of the coefficients themselves.

The foregoing is illustrated with an application to the traveling salesman problem, TSP, using k-change as the improvement discipline. Among other things it is shown that a k-change improvement can be affected by k 2-changes, that $\theta = 1$ for convex city sets, and that good estimates of θ can be made from a reduced TSP related to the given one.

Keywords:

iterated improvement, random restart, traveling salesman problem.

§1 Introduction

We present a new study of the Random Restart method for global optimization over a discrete set. Random Restart has the virtue that it is natural, amendable to analysis, and yet robust and effective. The method, and its analysis, can serve as a prototype for gaging more sophisticated optimal search methods. Indeed, most of the development derived here, including the power series analogue of the fundamental polynomial, has been carried

[1] Professor, Mathematics Department, Acadia University, Wolfville, Nova Scotia B4P 2R6, Canada. Partially supported by the National Sciences and Engeineering Research Council of Canada (NSERC) in the form of a discovery grant.

[2] Professor, School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332

over to more general restarting methods [1], which are not always Markov Chains. Our approach also invites the possibility of a classification of discrete optimization problems in terms of the character of the of the search tree generated by the imposed topology on the problem. For the Traveling Salesman Problem (TSP), for example, the search tree is short and bushy, see Figures 12 and 13.

The *optimization problem* for the objective f defined on the (very large but finite) set Ω can be organized in terms of three separate problems:

1. finding the globally optimal value $f_* = \min_{x \in \Omega} f(x)$.
2. finding at least one global optimizer $x_* \in S_*$ among the set of optimizers $S_* = \{x \in \Omega : f(x) = f_*\}$.
3. assuring that (1) is correct.

If f is unrestricted in the sense that for each $x \in \Omega$, $f(x)$ can have any value at all, then the optimization problem cannot be solved except by the method of *exhaustion* in which each and every point of Ω is examined. For only after examining each and every point can (1) be assured.

A major result of Simulated Annealing, Hajek's Theorem, settles all three problems, but not in finite time, since it asserts

$$\lim_{t \to \infty} \Pr(X_t \notin S_*) = 0,$$

where X_t is the random variable denoting the state of the process at time t, provided the anneal is cooled, $T \to 0$, according to

$$T = \frac{d}{\log(t+1)}.$$

In this, d is the depth of the deepest non-goal basin, [2],

It should be noted that for the class of problems known as *inverse problems* the desired result $f(x_*)$ is known in advance; it only remains to find where it occurs, x_*. An inverse problem is turned into an optimization problem by assigning a distance $d(f(x), f(x_*))$ between objective values, then the global optimum is $d = 0$. In this case, item (1) is not an issue.

If the optimal value cannot be assured by the method at hand, then settling issue (1) falls within the realm of an optimal stopping problem. That is, under what conditions should the search be terminated. Since the global optimum cannot be known with certainty, the answer can only be probabilistically asserted, one is forced to settle for a solution which is, hopefully, near optimal.

Under these conditions, the emphasis shifts to dealing with issue (2), that is, the search itself and designing strategies for searching efficiently. A traditional measure of search efficiency is the convergence rate to the set S_*, or S_* enlarged to include near optimal points, the *goal set*. This is the rate of decrease in probability that a goal state has not been found by time t

$$\Pr(X_i \notin S_*, i = 1, 2, \ldots, t).$$

See [3], [2], [4].

Another useful measure of search efficiency is the expected hitting time E to the set S_*. The expected hitting time measure is given in terms of meaningful units, namely the number of iterations that should be required to reach the goal basin thereby turning the problem into a deterministic excursion to the global set. This translates into an expected running time needed by the algorithm. For a search method amendable to Markov Chain analysis, the expected hitting time E is given by

$$E \approx \frac{1}{s} \frac{1}{1-\lambda} \tag{1}$$

in terms of two scalar parameters, *retention* λ, and *acceleration* s, see [5]. Retention is the Perron-Frobenius eigenvalue of the goal basin deleted transition matrix – denoted by $\hat{P}$ – and s is the dot product of the goal basin deleted starting/restarting distribution with the normalized Perron-Frobenius right eigenvector. Conveniently, some convergence rate assertions can be reworked into expected hitting times. For example, if the convergence rate is geometric, then the corresponding expected hitting time is finite (and computable from the rate).

But if the convergence assertion is only asymptotic, as in Hajek's Theorem, then it is possible for the expected hitting time to be infinite [5].

The use of expected hitting time has the additional benefit in that it can help in deciding issue (1). If the run time has exceeded the expected hitting time, one can assert a probability that the current best is the global best, in this case with probability .63, see §3.

I2R2

In this work we will examine the search method which combines iterated improvement with random restart, referred to as I2R2. In this method a strategy is furnished for attempting to improve any given solution. Such a strategy, also known as a local improvement or greedy algorithm, is usually problem specific and its design may involve the specialized insight of an expert in the problem domain. The prototype for I2R2 is the optimization of a differentiable objective defined on an open subset of Euclidean space. In this case an obvious improvement strategy is gradient descent.

Given an improvement strategy, it remains to provide it with a starting point. The simplest and most universal method for this is to select starting points uniformly at random from the domain.

The I2R2 approach to global optimization is not new. What is essentially the same is called multi-start in [6].

I2R2, like many global optimization methods, is a Markov chain over Ω and so has an associated transition probability matrix P. We will see that P has a very simple form, so simple in fact that the convergence rate and expected hitting time can be calculated directly. This stems partly from the fact that in I2R2, stochasticity only arises in the restart step, which we will always take to the uniformly at random. Therefore I2R2 is a homogeneous Markov chain. Convergence is geometric and expected hitting times are always finite in this case. It also stems from the fact that, between restarts, the fate of the process is completely determined by the start/restart choice, x_0. Consequently a central role in the theory is played by the probability θ that the restart is in the goal basin.

In Section 2 we derive the goal deleted transition matrix for I2R2 and exactly calculate its principle eigenvalue, λ, along with the normalized left and right eigenvectors. It is shown that $\eta = 1/\lambda$ is the sole positive root of a polynomial

$$f(x) = p_0 x + p_1 x^2 + \ldots + p_{n-1} x^n + p_n x^{n+1} - 1. \tag{2}$$

which is simply and naturally related to the improvement algorithm. The coefficient p_k of this polynomial is the probability of restarting k improvement steps from a local minimum. Not only is retention determined by f, so are s, E and the important parameter θ, the size of the goal basin. It is shown that acceleration s is strictly greater than 1 thereby endowing I2R2 with excellent parallelization properties, see [5].

In Section 3 we show that errors in computing the root η of (2) are many times less than errors which may be in the coefficients themselves. Contrary to intuition, our results indicate the coefficients play an equal role in contributing to the root error see Fig. 1 and eqn's (10) and (15). This opens the possibility of estimating η, and hence also s and E, dynamically. If the search algorithm keeps track of the number of steps taken to reach local minimums during the course of the search, then estimates of the polynomial coefficients can be made. The resulting error in η decays according to $R^{-1/2}$ where R is the number of restarts. Specifically the variance of the error is given by

$$\mathrm{var}(\Delta\eta) = \frac{(f(\eta^2)/f'(\eta))}{R} \approx \frac{(\theta/\beta)}{R}$$

where θ is the goal basin success probability as above and β is the expected number of iterations between restarts.

Unfortunately the parameter θ can not itself be obtained dynamically but must be estimated a priori. This is not unlike Hajek's parameter d [2], Azencott's parameter α [7] or Boender and Rinooy Kan's parameter w [6]. The balance of the paper is devoted to this problem for the Traveling Salesman Problem using 2-change as the improvement algorithm.

In Section 4 we show that swapping k links of a salesman's tour can be affected by no more than k 2-changes, hence it makes sense to concentrate on 2-change. In general, a 2-change locally minimal tour has no self-intersections. When the cities are confined to the hull of a convex set, then the number of self-intersections is an upper bound for the descent path using only special kinds of 2-changes. For an N city problem, this number in turn is bounded by $(N-3)N/2$. Further, in the convex case, $\theta = 1$ meaning that every tour improves to the globally optimal tour.

With a view to revealing how θ behaves under different instances of the same problem (for example the TSP with approximately the same values of N), we obtain empirical results when the cities are randomly selected in the unit square. Generally both θ and its variance decrease as N increases, see Fig. 5. Most remarkably, for the cases we looked at (N up to 35), the distribution of the number of descent steps is quite invariant both over randomly generated problems and database problems, see Fig. 12. This indicates that a good estimate of θ is available from historical experience.

In this section we also investigate adding cities to a TSP. We show that every adjacent pair of cities, A and B, along an optimal tour has an associated "maintenance" region into which a new city can be added with the result that the new optimal tour is merely a detour of (A,B). Surprisingly, θ is not necessarily stable under this process.

However, in the last Section we apply to foregoing ideas to a database TSP, Bays29. We show it is possible to judiciously delete cities one by one accompanied by only a small change in θ, see Fig. 13 and Table 2.

§2 Iterated Improvement + Random Restart

We envision a process combining a deterministic downhill operator g acting on points of the solution space and a uniform random selection operator U. The process starts with an invocation of U resulting in a randomly selected *starting* point x_0. This is followed by repeated invocations of g until a local minimizer is reached, this is the *descent sequence* starting from x_0.

Definition 1. The point $x \in \Omega$ is a *local minimizer* of f if for every neighbor y of x, $f(y) \geq f(x)$. Then $f(x)$ is a *local minimum.*

No neighbor of a local minimizer has a strictly improved objective. In general flat spots in the domain are a problem; under the definition above, a point within a flat spot might have no neighbors.

Upon reaching a local minimizer, the process is restarted with another invocation of U. Thus the domain is partitioned into *basins* B_i, $i = 0, 1, \ldots$ as determined by the equivalence relation $x \equiv y$ if and only if $g^k(x) = g^j(y)$ for some k, j. The local minimizer or *settling point* b of basin B is given as the limit $\lim_{k\to\infty} g^k(x)$ where x is any point of B; of course, since the domain is finite, this sequence is eventually constant.

Graph theoretically, a basin is organized as a tree with the settling point being the root of the tree. Different basins are connected only by the restart process, and so, exclusive of restart, I2R2 enforces a topology on the domain whose graph is a forest of trees.

Denote by ℓ the number of non-goal basins for the problem. Let B_0 refer to the goal basin. We take the *depth* of a tree to be its maximum path length. Denote by d the maximum tree depth over the non-goal basins.

By indexing the points of Ω according to basins, and starting with the goal basin, the $|\Omega| \times |\Omega|$ transition matrix for such a process assumes the following block form

$$P = \begin{bmatrix} B_0 & 0 & \dots & 0 \\ Q & B_1 & \dots & Q \\ \vdots & \vdots & \ddots & \vdots \\ Q & Q & \dots & B_\ell \end{bmatrix},$$

To conserve notation, we also use B_i to denote the sub-matrix corresponding to basin B_i. Within a basin we index points according to increasing path length from the settling point. Then each sub-matrix B_i has the form

$$B_i = \begin{bmatrix} p & p & p & \dots & p \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix},$$

with $p = 1/|\Omega|$, the uniform restarting probabilities. The 1's in this matrix are in the lower triangular part but not necessarily on the sub-diagonal. The blocks designated by Q are generic (of appropriate size) for the form

$$Q = \begin{bmatrix} p & p & \dots & p \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}.$$

Definition 2. By the expected hitting time E, we mean the expected number of iterations t (improvements and starts/restarts) until the process X_t first achieves a point in the basin B_0 containing a global minimizer, the *goal basin.*

We now calculate E, see eqn (1).

Solving for λ and s, the Fundamental Polynomial

The deleted transition matrix $\hat{P}$ is P with the rows and columns corresponding to B_0 deleted. Before computing this, we first simplify P by considering an equivalent process.

In the forest of trees model, it is clear that all states which are a given number of steps from a settling point are equivalent as far as the algorithm is concerned. Let $r_j(i)$ be the number of vertices j steps from the local minimizer of basin i, $0 \le j \le \text{depth}(B_i)$. Then put $r_j = \sum_{i=1}^{\ell} r_j(i)$ with the sum being taken over the non-goal basins, this for $0 \le j \le d$. Thus r_j denotes the total number of vertices which are j steps from a local, non-global, minimizer. In particular, $r_0 = \ell$ is the number of local minimizers.

Therefore the forest of trees model, in which each vertex counts 1, is equivalent to a single, linear tree in which the vertex j edges from a local minimizer counts equal to r_j. Under this equivalency, the $\hat{P}$ matrix is replaced by

$$P' = \begin{bmatrix} p_0 & p_1 & p_2 & \dots & p_{d-1} & p_d \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix}. \tag{3}$$

in which

$$p_j = r_j/|\Omega|, \qquad \text{for} \quad 0 \le j \le d. \tag{4}$$

With respect to P', the 1's are in fact on the subdiagonal and consequently P' has the form of a companion matrix. Its characteristic polynomial therefore is

$$-\lambda^{d+1} + p_0\lambda^d + p_1\lambda^{d-1} + \ldots + p_{d-1}\lambda + p_d.$$

Definition 3. Upon setting $x = 1/\lambda$ in the characteristic polynomial for the goal deleted transition matrix, we get a polynomial which we will refer to as the *fundamental polynomial*

$$f(x) = p_0x + p_1x^2 + \ldots + p_{d-1}x^d + p_dx^{d+1} - 1. \tag{5}$$

Notice that the degree of the fundamental polynomial is the depth of the deepest basin plus 1 or, equivalently, equal to the number of vertices on the longest path to a local minimizer.

Theorem 1. *All non-zero eigenvalues of $\hat{P}$ are eigenvalues of P', more exactly*

$$\text{det}P' = \lambda^k \text{det}\hat{P}$$

where k is the number of merged nodes. Therefore, the reciprocal of every eigenvalue of P' is a root of (5), and conversely, the reciprocal of every root of (5) is an eigenvalue of P'.

Proof. We show that two nodes may be merged. Since the complete merging of nodes can be achieved by successively merging two at a time, this will prove the theorem. Hence suppose nodes x_s and x_t lead to x_m. Then both rows s and t of $\hat{P}$ have the form $(\delta_m) = (\,0 \;\; 0 \;\; \ldots \;\; 0 \;\; 1 \;\; 0 \;\; \ldots \;\; 0\,)$ with the 1 in the mth position and 0's elsewhere. Under the equivalence, assume that node t is merged with node s whose new chance of being selected for restart is therefore the sum $\mu_s + \mu_t$. Row s of P' will be (δ_m) while column s of P' will be the sum of columns s and t of $\hat{P}$, namely $\mu_s + \mu_t$, for rows corresponding to local minima and 0 for the other rows.

To simplify, we may re-index both matrices. For $\hat{P}$ write x_s first and x_t next followed by the other nodes in their same order. Then $\hat{P} - \lambda I$ will be

$$\begin{bmatrix} -\lambda & 0 & \ldots & 0 & 1 & 0 & \ldots & 0 \\ 0 & -\lambda & \ldots & 0 & 1 & 0 & \ldots & 0 \\ p_{31} & p_{32} & \cdots & p_{3,m-1} & p_{3m} & p_{3,m+1} & \cdots & p_{3d} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{d1} & p_{d2} & \cdots & p_{d,m-1} & p_{dm} & p_{d,m+1} & \cdots & p_{dd} - \lambda \end{bmatrix}$$

To proceed, expand $\det(\hat{P} - \lambda I)$ by the first row in minors, then expand each subdeterminant in the same way. Upon factoring $-\lambda$ in common to all members and adding the last two determinants, we get

$$(-\lambda)\det \begin{bmatrix} p_{33} - \lambda & \ldots & p_{3d} \\ \vdots & \ddots & \vdots \\ p_{d3} & \ldots & p_{dd} - \lambda \end{bmatrix} +$$

$$(-1)^m \det \begin{bmatrix} p_{31} + p_{32} & p_{33} - \lambda & \ldots & p_{3,m-1} & p_{3,m+1} & \cdots & p_{3d} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ p_{d1} + p_{d2} & p_{d3} & \cdots & p_{d,m-1} & p_{d,m+1} & \cdots & p_{dd} - \lambda \end{bmatrix}.$$

But this final result is exactly the expansion of $\det(P' - \lambda I)$ for the reduced matrix P' proving the theorem. In reducing the degree of the characteristic polynomial by one, only the zero root $-\lambda = 0$ has been lost; it occurred in the factor step above. ▮

For the record we state some obvious facts.

Fact. *All coefficients p_j, $0 \le j \le d$ of the fundamental polynomial are strictly positive.*

This is because, if there is a point j steps from a local minimum, there must also be a point $j - 1$ steps away as well.

Let θ denote the probability of a start or restart in the goal basin.

Fact.

$$\theta + p_0 + p_1 + \ldots + p_d = 1. \tag{6}$$

This merely expresses that a restart must select some point in Ω.

In light of this result, evaluating the fundamental polynomial at $x = 1$ yields the following.

Corollary 1. *$f(1) = -\theta$.*

Proposition 1. *The fundamental polynomial f has a unique greater than 1 root.*
Proof.

In fact the derivative $f'(x)$ of f has all positive coefficients and so is itself positive for $x \geq 0$. Therefore f increases for $x \geq 0$ unboundedly. From the fact that $f(1) = -\theta$ the conclusion follows. ▮

Definition 4. Let η be the unique greater than 1 root of the fundamental polynomial. Then $1/\eta$ is the unique maximal root of the characteristic polynomial of $\hat{P}$ and hence η is the reciprocal of the retention λ.

To calculate the acceleration s, we first find the left and right eigenvectors of $\lambda = 1/\eta$. The right Perron-Frobenius eigenvector, χ, of $\hat{P}$ is easily calculated. From (3) we get the recursion equations

$$\chi_k = \lambda \chi_{k+1} \qquad k = 0, \ldots, d-1.$$

And so each is given in terms of χ_0,

$$\chi_k = \eta^k \chi_0, \qquad k = 1, \ldots, d.$$

Similarly, we get recursion equations for the components of the left eigenvector ω in terms of ω_0,

$$\omega_k = \omega_0(\eta p_k + \eta^2 p_{k+1} + \ldots + \eta^{d+1-k} p_d).$$

We may normalize ω so as to be a probability vector, $\sum \omega_i = 1$, from which it follows that

$$\omega_0 = \frac{\eta - 1}{\eta\theta}.$$

We normalize χ to have unit inner product with ω, $\sum \omega_i \chi_i = 1$, from which it follows that

$$\chi_0 = \frac{1}{\omega_0 \eta f'(\eta)} = \frac{\theta}{(\eta - 1) f'(\eta)}.$$

But $s = 1/(\chi \cdot \hat{\alpha}_0)$ where $\hat{\alpha}_0$ is the non-goal partition vector of the starting distribution,

$$\hat{\alpha}_0 = \begin{pmatrix} p_0 & p_1 & \ldots & p_d \end{pmatrix}.$$

Substituting from above, we get

$$s = \frac{\eta(\eta - 1) f'(\eta)}{\theta}. \tag{7}$$

Theorem 2. *For I2R2 $s > \eta > 1$.*

Proof. The average slope of the line between $(1, -\theta)$ and $(\eta, 0)$ is less than the slope of f at η because f' is increasing, therefore $f'(\eta) > \theta/(\eta - 1)$. Hence, since $\eta > 1$,

$$s = \eta\frac{(\eta - 1)f'(\eta)}{\theta} > \eta > 1.$$

∎

Remark 1. Since acceleration exceeds 1, I2R2 is superlinearly sped-up under independent, identical processes parallelization, see [5].

Goal attainment probabilities

Restarting in a goal basin can be regarded as a Bernoulli trial with success probability θ. Accordingly, the expected number of starts/restarts to find such a basin is $1/\theta$ and the probability of not finding a goal basin after k starts/restarts is $(1 - \theta)^k$. Taking k to be a fraction m of the expectation we have

$$\begin{aligned}\Pr(\text{goal basin has not been found after } \tfrac{m}{\theta} \text{ restarts}) &= (1-\theta)^{\frac{m}{\theta}} \\ &\approx e^{-m} \qquad \text{as } \theta \to 0.\end{aligned}$$

Therefore the probability of not having found a goal basin within the expected number of starts/restarts, $1/\theta$, is $e^{-1} = 37\%$. Alternatively, to find the goal basin with 50% chance requires $m = 69\%$ of the expected restarts.

§3 Run time estimation of retention, acceleration and hitting time

Consider again the coefficient p_j of the term x^{j+1} of the fundamental polynomial, see eqn (4). Since p_j is the ratio of the number of points j steps from a local minimizer divided by the number of points in the space, we see it is exactly the probability of starting or restarting j steps from a local minimizer. Thus the linear coefficient is the probability of restarting on a local minimum, the quadratic coefficient is the probability of restarting one downhill step from a local minimum and so on.

As a result, we consider the possibility of estimating the fundamental polynomial during an execution of the algorithm, that is a *run*. Toward this end, for each $j = 0, 1, \ldots, d$, we can maintain a count of the number of restarts, $r_j(t)$, requiring j steps to reach a local minimizer up to the tth iteration of the run as well as the total number of restarts, $R(t)$, altogether. We shorten these to r_j and R if t is understood.

Since the $r_j(t)$ might well include multiple counts, and since, in any case, $|\Omega|$ may not be available or be so large that a significant fraction of this size would be needed to obtain reasonable coefficient estimates, eqn (4) is impractical to use directly. On the other hand, the ratio $r_j(t)/R(t)$ does estimate the associated *conditional* probability conditioned on the restart being among the non-goal basins. We define q_j to be the conditional probability that a restart will require j steps to reach a local minimum given the restart is in a non-goal basin. The relationship between p_j and q_j is given by

$$p_j = q_j(1-\theta) = \lim_{t\to\infty} \frac{r_j(t)}{R(t)}(1-\theta). \tag{8}$$

The quotient on the right hand side is the empirical estimate of the conditional probability of p_j conditioned on starting in some non-goal basin and the factor $(1-\theta)$ is that very probability. Summing the estimated coefficients gives

$$\sum_j \frac{r_j(t)}{R(t)}(1-\theta) = 1-\theta$$

since the $r_j(t)$ sum to $R(t)$. Thus we obtain the required condition of eqn (6).

As explained in the Introduction, θ must be estimated by independent means. Here we obtain results about the convergence rates and statistical properties of the convergence of the coefficient estimates and, more importantly, of η. As we have seen, from the knowledge of this root, estimates of λ, s and E follow.

Connection between root error and coefficient error.

In this section we show that the error $\Delta\eta$ in the root η of the fundamental polynomial is much less than the errors Δp_j in its coefficients. Toward that end, regard the fundamental polynomial as a function of its coefficients and η as well as x, thus

$$0 = f(\eta, p_0, p_1, \ldots, p_d).$$

Differentiating

$$0 = \frac{\partial f}{\partial \eta}\Delta\eta + \frac{\partial f}{\partial p_0}\Delta p_0 + \ldots + \frac{\partial f}{\partial p_d}\Delta p_d$$

and from this we derive

$$\Delta\eta = \frac{-1}{f'(\eta)}\left[\eta\Delta p_0 + \ldots + \eta^{d+1}\Delta p_d\right]. \tag{9}$$

Let j' be the subset of indices among $0,1,\ldots,d$ for which $\Delta p_j \geq 0$ and j'' those for which $\Delta p_j < 0$. Let $\sigma = \sum_{j'} \Delta p_{j'}$. Since the algebraic sum of the errors will be zero, $\sum_j \Delta p_j = 0$, then also

$$\begin{aligned}\sigma &= -\sum_{j''} \Delta p_{j''} = \sum_{j''} |\Delta p_{j''}| \\ &= \frac{1}{2}\sum_{j=0}^{d} |\Delta p_j|.\end{aligned}$$

The latter is the ℓ_1-norm of the vector of the Δp's. Then

$$\begin{aligned}
|\Delta\eta| &= \frac{1}{f'(\eta)}\Big|\sum_{j=0}^{d}\eta^{j+1}\Delta p_j\Big| \\
&= \frac{1}{f'(\eta)}\Big|\sum_{j'}^{d}\eta^{j'+1}\Delta p_{j'} - \sum_{j''}\eta^{j''+1}|\Delta p_{j''}|\Big| \\
&< \frac{1}{f'(\eta)}\big[\sigma\max_{0\le j\le d}\eta^{j+1} - \sigma\min_{0\le j\le d}\eta^{j+1}\big] \\
&= \frac{1}{f'(\eta)}\sigma\big[\eta^{d+1}-\eta\big] \\
&= \frac{\eta(\eta^d-1)}{2f'(\eta)}\sum_{j=0}^{d}|\Delta p_j|.
\end{aligned}$$

We have proved the following theorem.

Theorem 3.

$$\frac{|\Delta\eta|}{\sum_{i=0}^{d}|\Delta p_j|} < \frac{\eta(\eta^d-1)}{2f'(\eta)}. \tag{10}$$

The effect of θ on η

The number of iterations between restarts equals the number of downhill steps plus the restart itself, and therefore is at least 1, and so we see that

$$h(x) = q_0x + q_1x^2 + \ldots + q_dx^{d+1}.$$

is the probability generating function for the number of iterations between restarts conditioned on the restart occurring among the non-goal basins. Let β denote the expected number of iterations between restarts. Then β is the derivative

$$\beta = h'(1) = q_0 + 2q_1 + \ldots + (d+1)q_d. \tag{11}$$

Since the expected number of iterations to find the goal basin, E, is the expected number of *restarts* to find the goal basin, $1/\theta$, times the expected number of iterations between restarts, we have

$$E = \frac{\beta}{\theta}. \tag{12}$$

Unless each non-goal state is its own basin, it must necessarily be that $\beta > 1$ and this we will assume throughout.

Recall that we do not directly estimate p_j but instead we estimate q_j. Thus we see that

$$\Delta p_j = \Delta q_j(1-\theta) - q_j\Delta\theta$$

so strictly speaking we need some estimate on the error in our estimation of θ. However, this is very difficult to ascertain. Nevertheless, it is important to try to understand the effect of θ on the root η. In terms of h, f is given by $f(x) = (1-\theta)h(x) - 1$ and so to find η we are solving $h(x) = 1/(1-\theta)$.

Applying the generalized Arithmetic-Geometric Mean inequality to eqn (11) we have

$$\begin{aligned} x^\beta &= (x)^{q_0}(x^2)^{q_1}(x^3)^{q_2}\cdots(x^{d+1})^{q_d} \\ &\le q_0x + q_1x^2 + q_2x^3 + \cdots + q_dx^{d+1} \end{aligned} \tag{13}$$

for $x > 0$. Thus, the root η of $f(x)$ is less than or equal to the root $\hat{\eta}$ of $x^\beta = 1/(1-\theta)$. This proves the following

Theorem 4.

$$1 \le \eta \le \hat{\eta} = \left(\frac{1}{1-\theta}\right)^{1/\beta}. \tag{14}$$

In practice $\beta >> 1$ usually, see Fig. 12.

From a Taylor Series expansion we see that

$$\hat{\eta} = (1 + \theta + \theta^2 + \cdots)^{1/\beta} = 1 + \theta/\beta + O(\theta^2)$$

so that for small θ we have $\hat{\eta} \approx 1 + \theta/\beta$. Similarly since $h'(1) = \beta$, by the Inverse Function Theorem,

$$\begin{aligned} \eta = h^{-1}\left(\frac{1}{1-\theta}\right) &= h^{-1}(1) + \left(h^{-1}\right)'(1)\left(\frac{1}{1-\theta} - 1\right) + \ldots \\ &= 1 + \frac{1}{\beta}\left(\frac{\theta}{1-\theta}\right) + \ldots \end{aligned}$$

and so also

$$\eta = 1 + \theta/\beta + O(\theta^2). \tag{15}$$

For small values of θ we know $\hat{\eta}$ is a very good approximation to η. Using this, we see that for small values of θ we have $\lambda \approx 1 - \theta/\beta$. Notice that by eqn (12) we have $\lambda \approx 1 - 1/E$ as well.

Remark 2. The expression $\lambda \approx 1 - \theta/\beta$ nicely illustrates the effects of both θ and the p_j's on λ (the p_j's through β). If β increases, then the average number of downhill steps increases so it is more likely the process will be "retained" in the non-goal states for additional iterations of the Markov Chain. Similarly, if θ decreases, then upon restart it is less likely to find the goal basin so more likely to remain in the non-goal states. Similar reasoning applies if β decreases or θ increases.

We can also use this estimate to obtain another estimate of the error ratio from Theorem 3. For this we assume that β, the average number of downhill steps taken in non-goal basins, is held constant and we only keep terms first order in θ.

From Theorem 3 we have the following.

Theorem 5. *Assuming that θ is small and $\beta \approx d/2$ is constant then*

$$\frac{|\Delta\eta|}{\sum_{i=0}^{d} |\Delta p_j|} < \frac{\eta(\eta^d - 1)}{2f'(\eta)} < \frac{\theta}{\beta}$$

to first order in θ.

Proof. Since $\eta \approx 1+\theta/\beta$ we see that $\eta^d \approx (1+\theta/\beta)^d \approx 1+d(\theta/\beta)$ so that $\eta(\eta^n - 1) \approx d(\theta/\beta)$ (to first order). Using the fact that $f'(\eta) > f'(1) \approx d/2$, we obtain the desired result. ∎

In practice θ is very small and β is large, so this ratio is quite good, see Fig. 11. In fact, if $\beta \approx d/2$, then this says that we should expect

$$|\Delta\eta| < 2\theta \text{ average } (|\Delta p_j|) \tag{16}$$

which gives us an estimate of the size of $\Delta\eta$ in terms of the average size of $|\Delta p_j|$. This is very nice, since it tells us that usually the error in the root is much smaller than the errors in any of the coefficients.

In Table 1 we present the results of an empirical study of the error ratio. The first two columns are for $\theta = 0.1$, the third and fourth for $\theta = 0.01$ and the fifth and sixth for $\theta = 0.001$. The number reported is 1,000 times the actual ratio (to conserve space). Each pair of columns presents the (maximum) simulated error ratio and the error bound predicted by Theorem 5. For each simulation we obtained 10,000 samples from the appropriate distribution. The table shows that the simulated ("actual") ratio is never worse than the predicted ratio. That is, in actuality the ratio of the errors is much better than the theory says (as this is based on a first order approximation only).

Table 1 Maximum simulated error ratio vs predicted for various θ

$\theta = 0.1$		$\theta = 0.1$		$\theta = 0.01$		$\theta = 0.01$		$\theta = 0.001$		$\theta = 0.001$	
Sim	Pred	Sim	Pred	Sim	Pred	Sim	Pred	Sim	Pred	Sim	Pred
5.77	11.1	1.45	7.06	0.67	1.25	0.14	0.53	0.02	0.05	0.01	0.05
1.17	7.74	1.33	6.02	0.55	1.14	0.17	0.64	0.05	0.17	0.01	0.06
1.93	5.70	3.19	8.06	0.15	0.59	0.25	0.96	0.02	0.07	0.01	0.06
3.12	10.4	3.30	9.64	0.47	1.31	0.09	0.48	0.03	0.07	0.06	0.17
4.96	14.9	4.61	10.7	0.81	1.77	1.54	1.99	0.05	0.14	0.01	0.05
2.25	6.80	2.66	8.51	0.11	0.55	0.18	0.78	0.02	0.09	0.02	0.09
4.03	9.91	2.27	5.97	0.11	0.62	0.15	0.58	0.02	0.07	0.08	0.15
4.67	12.5	2.64	9.65	0.21	0.64	0.23	0.71	0.02	0.08	0.01	0.05
1.76	5.10	2.36	5.97	0.15	0.66	0.23	1.02	0.07	0.18	0.02	0.06
2.54	9.20	3.66	13.1	0.15	0.55	0.37	1.10	0.05	0.13	0.03	0.07

Run time estimation

During the course of a run the coefficient estimates are random variables. We will continue to assume that conditioning with respect to the non-goal basins is compensated for, for example by utilizing the factor $1-\theta$ or by noting a hit on the goal basin and continuing the run. Let r_j, $j=0,1,\ldots,d$, denote the number of restarts at depth j and let R be the number of starts/restarts altogether. The r_j are multinomially distributed with parameters p_j, $j=0,1,\ldots,d$ and R. We have the following for the variance of the error in η.

Theorem 6. *The variance of $\Delta\eta$ decreases in inverse proportion to the number of restarts, R^{-1}, and is given by*

$$\begin{aligned}\mathit{var}(\Delta\eta) &= \frac{1}{R}\frac{f(\eta^2)}{f'^2(\eta)} \\ &\approx \frac{(\theta/\beta)}{R}.\end{aligned} \tag{17}$$

Proof. From first principles

$$\operatorname{var}(r_j) = Rp_j(1-p_j) \qquad \text{and} \quad E(r_i r_j) = R(R-1)p_i p_j,$$

and the error variables obey

$$\operatorname{var}(\frac{r_j}{R}-p_j) = \frac{1}{R}p_j(1-p_j) \qquad \text{and} \quad E\left[(\frac{r_i}{R}-p_i)(\frac{r_j}{R}-p_j)\right] = -\frac{1}{R}p_i p_j.$$

Recall eqn (9),

$$\Delta\eta = \frac{\eta}{f'(\eta)}\left(\frac{r_0}{R}-p_0\right)+\ldots+\frac{\eta^{d+1}}{f'(\eta)}\left(\frac{r_d}{R}-p_d\right).$$

Put $\xi_j = (\frac{r_j}{R}-p_j)$ and $a_j = \frac{\eta^{j+1}}{f'(\eta)}$. Since the mean vanishes, $\mu(\Delta\eta)=0$, from the above we have

$$\begin{aligned}\operatorname{var}(\Delta\eta) &= E\left[(a_0\xi_0+\ldots+a_d\xi_d)^2\right] \\ &= \frac{a_0^2}{R}p_0(1-p_0) - 2\frac{a_0a_1}{R}p_0p_1 - \ldots - 2\frac{a_0a_d}{R}p_0p_d + \\ &+ \frac{a_1^2}{R}p_1(1-p_1)+\ldots+\frac{a_d^2}{R}p_d(1-p_d) \\ &= \frac{1}{R}\left(a_0^2p_0+\ldots+a_d^2p_d\right) - \frac{1}{R}\left(a_0p_0+\ldots+a_dp_d\right)^2\end{aligned}$$

Replacing the definitions of the a_j in the last term and noting that the second term is just $1/Rf'^2(\eta)$, we have

$$\begin{aligned}&\frac{1}{R}\frac{1}{f'^2(\eta)}\left[(\eta)^2p_0+\left(\eta^2\right)^2p_1+\ldots+\left(\eta^{d+1}\right)^2p_d-1\right] \\ &\qquad = \frac{1}{R}\frac{f(\eta^2)}{f'^2(\eta)}.\end{aligned}$$

This proves the equality of eq. (17). Using eq. (15), we see that $f(\eta^2) \approx f(\eta)+f'(\eta)\eta(\eta-1)$ from which we obtain the first order approximation for small θ. ∎

Equation (17) is illustrated in Fig. 1. The upper figure tracks the estimate, $r_7(t)/R(t)$ of one of the polynomial coefficients, while the lower figure tracks the estimate of η.

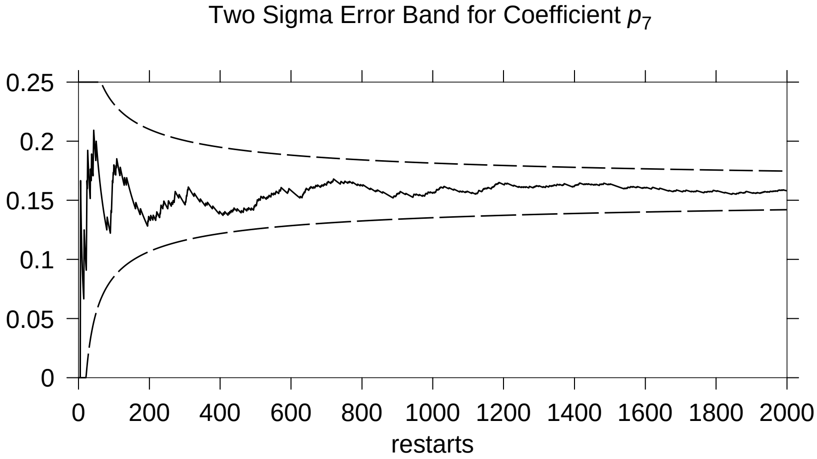


Fig. 1(a) Typical coefficient error and 95% confidence band.

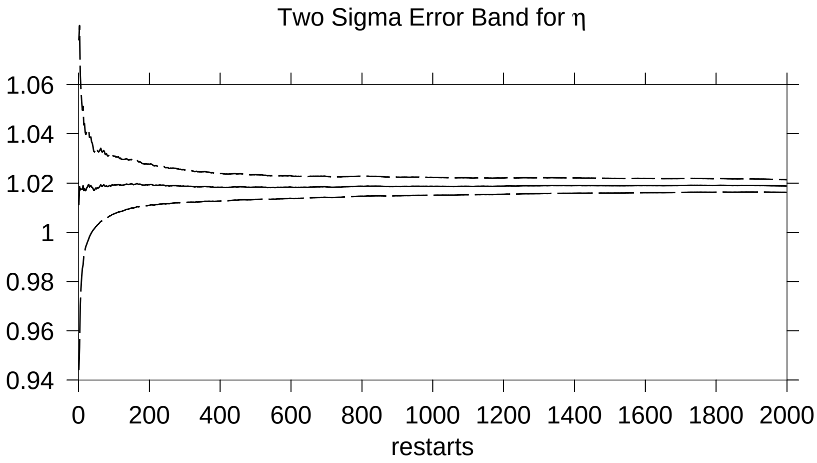


Fig. 1(b), Typical empirical error for η and 95% confidence band.

§4 I2R2 Parameters for the TSP

The *k-change* algorithm was used effectively along with random restart for the traveling salesman problem (TSP) by Lin and Kernigan [8]. A *tour* t is a feasible subset of the set S of all edges joining pairs of cities of the problem. To be feasible, the edges must form a Hamiltonian cycle. Then a k-change is the exchange of k elements of t with a disjoint set of k elements of S forming a new tour t'.

In their paper popularizing simulated annealing, Kirkpatrick, Gelatt, and Vecchi restrict to 2-change operations only. Use permutations of the set $\{1, 2, \ldots, N\}$ to describe tours and consider the N city tour

$$t = \left(i_1, \ldots, i_k, \underbrace{i_{k+1}, \ldots, i_m}_{backside}, i_{m+1}, \ldots, i_N, i_1 \right).$$

The 2-change which exchanges links (i_k, i_{k+1}) and (i_m, i_{m+1}) in favor of (i_k, i_m) and (i_{k+1}, i_{m+1}) produces the modified tour t'

$$i_1, \ldots, i_k, i_m, i_{m-1}, \ldots, i_{k+1}, i_{m+1}, \ldots, i_N, i_1.$$

This is illustrated in Fig. 2. We indicate the *frontside sub-tour* as the dotted path from i_{m+1} through i_1 and onto i_k. The links to be broken (i_k, i_{k+1}) and (i_m, i_{m+1}) are shown as solid directed line segments. The tour is closed by the *backside sub-tour* from i_{k+1} to i_m and is also illustrated as a dotted line segment. Note that the tails of both the old and new tours meet at node i_k, which we indicate with the notation (tt), and the heads of the old and new meet at i_{m+1} indicated by (hh). One can think of link (i_k, i_{k+1}) rotating around i_k across the backside tour to become the link (i_k, i_m) and the same for (i_m, i_{m+1}) becoming (i_{k+1}, i_{m+1}).

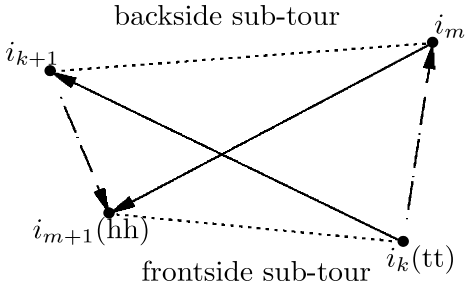


Fig. 2, butterfly 2-change

Upon the choice of initial link to be removed, (i_k, i_{k+1}), of which there are N possibilities, two of the 4 nodes of a 2-change are decided. The replacement link to be constructed must proceed from i_k but its other end, i_m, is arbitrary except that nodes i_k, i_{k+1} and

i_{k-1} are ineligible, thus leaving $N-3$ choices. (The latter choice creates a i_{k-1}, i_k loop.) From i_m there are two links, but only the one joining the backside sub-tour is eligible in order to avoid a short circuit. In total there are

$$\frac{N(N-3)}{2} \tag{18}$$

2-change modifications to t if forward and reverse tours are regarded as identical. Note that one of the replacement edges joins the tails of the original edges and the other joins the heads. The replacement edges are themselves directed and, together with the original links, create two special nodes, the tt node from which the tails of the links issue and the hh node to which the links are directed.

A side effect of the 2-change is to reverse the backside sub-tour. Hence a 2-change may be referred to as a *partial path reversal* or PPR. In this case, we say tour t' is a *neighbor* of the tour t. Since a second reversal of the same sub-tour brings back t, we see that t is a neighbor of t' as well. From eq. (18), each tour has $N(N-3)/2$ neighbors.

Using the permutation notation for tours and partial path reversal representation for 2-change, it is easy to see that by judicious choice of a sequence of partial path reversals, any tour may be converted to any other. But more is true, we now show that any k-change can be realized as a sequence of at most k PPR's.

Let a k-change be given from the identity permutation $1, 2, \ldots, N$; we seek to restore the identity using PPR's. In writing the permutation form of a tour, without loss of generality, we may start with city 1.

In general the modified tour will consist of *segments* or subsequences of cities already in correct identity order and isolated cities. As a segment is incorporated into a tour, we identify the segment with either the representation TH or HT. The former is used if the segment is from low to high order, for example the segment $2, 3, ...$ would be designated TH. If the segment is traversed in high to low order, then we use HT. In this scheme, city 1 counts as the high end. For example the 3 segment tour $1, 5, 4, 2, 3, 6, 1$ would be written $H|HT|TH|T$ where we have used $|$ to denote the gaps between segments. It is clear that for segments two or more nodes in length, a gap of the form $H|H$ or $T|T$ cannot denote a link of the identity tour.

A PPR which keeps segments intact will take place in the gaps. In the 3 segment tour above, the PPR with backside 5 to 3 produces the tour $1, 3, 2, 4, 5, 6, 1$ whose parity is the same as the original because it reversed the $HT|TH$ partial path.

Theorem 7. *Any k-change can be effected as a sequence of at most k 2-changes.*

Proof. We proceed by induction. Obviously the statement is correct for $k = 2$. It is worthwhile to note that a 2-change cannot have an isolated city, that is a length 1 segment, because a 2-change can not change both links incident on a city. Further, in the parity representation of its segments, it must have both an $H|H$ gap and a $T|T$ gap. The PPR is performed between these gaps and brings two desired edges into place at once.

Now assume $k > 2$. If the given tour has an isolated city then clearly the PPR with it as the terminus of the backside subtour, bringing it into correct identity position, occurs without breaking any existing correct links. Hence the number of incorrect links is reduced by 1 and induction may proceed.

Next assume t consists of segments only and has an $H|H$ or a $T|T$ gap. Then there must by necessity be one of the other type as well; in fact they must occur in balancing pairs. For if not, consider the partition of the multi-set of the H's and T's defined by the gaps themselves. The gaps are directed edges and so partitions this set according to antecedents and succedents of the gaps. If there is an $H|H$ pair but no $T|T$ pair, then there must be 2 more H labels than T labels which is impossible.

Now partition the segments according their orientation, let F be those that are of TH type and let R be those of HT type. Since a tour must reach all cities, there are always at least two links of the identity tour with one city in F and the other in R. Further, since in the given tour, the segments containing these nodes are oppositely oriented, there must necessarily be a PPR which joins them. Performing this PPR reduces the number of incorrect links by at least one and induction may proceed in this case as well.

Finally consider the mono-mode case in which the pattern is

$$(1)H|TH|\ldots|TH|T,$$

(we use (1)H to indicate the first city is 1, which is an H) and we may assume without loss of generality that the next city is not 2. A PPR based on gaps cannot make a correct link in this case and we must break some segment to proceed. We do this by a PPR which breaks 23 and makes 12, thereby not reducing the number of incorrect links. But this PPR does create an $H|H$ gap and a matching $T|T$ gap; the pattern will be

$$(12)H|HT|\ldots|HT|(3)TH|\ldots|TH|T. \tag{19}$$

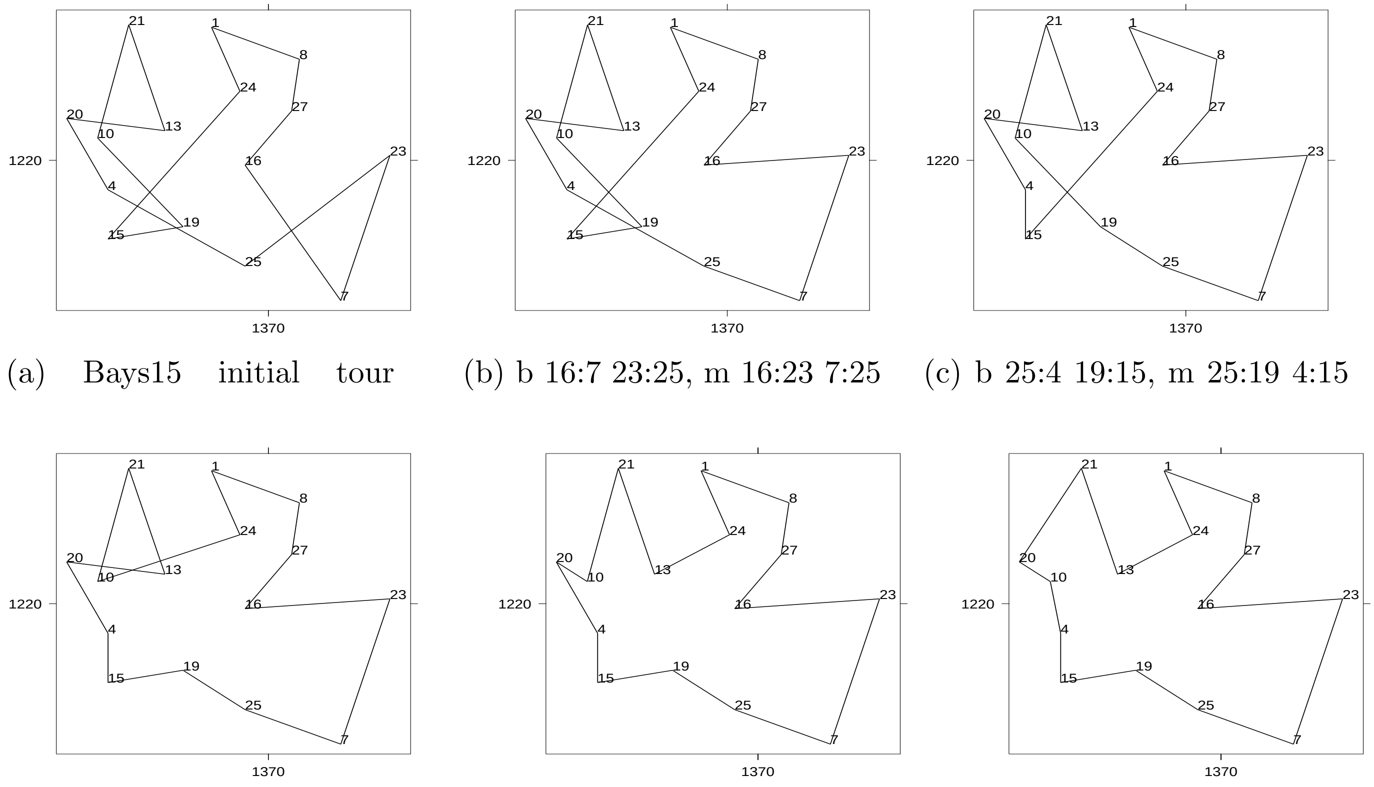
An inspection of (19) shows that the F and R partition as defined above are in fact the subtours from just after 2 up to 3 for R and the remainder for F. As above there must be at least two identity links to be made between them, and moreover, the PPR will not destroy the $H|H$, $T|T$ pattern, only making the 23 link will do that. Continue such PPR's until only 2 incorrect links remain. At that point, a single PPR will reduce the number of incorrect links to zero. Since there will be at least one out of place reduction until the last step, which brings about a reduction of two, we have achieved a reduction to the identity tour in k PPR's as required. ∎

Remark. *This result is sharp as the 3-change* $1, 5, 6, 7, 2, 3, 4, 8, 9 = 0$ *can not be converted to the identity using 2 2-changes.*

Definition 5. A *steepest downhill step* from tour t is any neighbor t' of t whose tour length, or weight, $wt(t')$ is minimum. Since we assume the cities of a problem lie in Euclidean space, the probability that two distinct tours have the same length is zero. Therefore we will assume that the steepest downhill step is uniquely determined. The *steepest descent* starting from t is the sequence of steepest downhill steps $t = t_0$, t_1, …, t_m, the descent sequence, such that no neighbor of t_m has strictly smaller tour length. In this case t_m is a *local minimum*.

Definition 6. A tour which cannot be improved by a 2-change operation is called a 2-opt tour.

In Fig. 3 we show a typical descent sequence for the Bays15 problem (cf. the last section "Application to the Bays29 problem") and give its "break" and "make" links.



(a) Bays15 initial tour

(b) b 16:7 23:25, m 16:23 7:25

(c) b 25:4 19:15, m 25:19 4:15

(d) b 19:10 15:24 m 19:15 10:24

(e) b 20:13 10:24 m 20:10 13:24

(f) b 4:20 10:21 m 4:10 20:21

Fig. 3

Order of Magnitude calculations for PPR

The k-change algorithm was studied extensively by Chandra, Karloff, and Tovey [9] who obtained two-sided performance ratio order of magnitude estimates. Below we summarize their results as specialized to 2-change and to Euclidean TSP of N cities in the plane. Let t denote a 2-change tour and t_g a globally optimal tour.

Theorem 8. *(CKT)*

$$\frac{wt(t)}{wt(t_g)} \leq 4\sqrt{N}. \tag{i}$$

For infinitely many values of N there is an n-city TSP and a 2-change tour t for which

$$\frac{wt(t)}{wt(t_g)} \geq \frac{1}{4}\sqrt{N}. \tag{ii}$$

There is a constant c such that for infinitely many values of N there is an N-city TSP and a 2-change tour t for which

$$\frac{wt(t)}{wt(t_g)} \geq c\frac{\log N}{\log\log N}. \tag{iii}$$

Let $\mathcal{I}_N$ denote an instance of an N-city TSP in which the cities are selected i.i.d. uniform randomly in the unit square. For a constant C, the expected number of 2-changes made by the 2-change algorithm on $\mathcal{I}_N$ is at most

$$\left(8C\sqrt{N}\right) N^{10}\log_2 N. \tag{iv}$$

Assertion (iv) is derived from the following result due to Kern [10]

Theorem 9. *(Kern) There is a constant c such that the probability that 2-change on $\mathcal{I}_N$ does more than N^{16} iterations is at most c/N.*

Rearranging (i) to the form

$$wt(t_g) \geq \frac{1}{4\sqrt{N}}wt(t)$$

allows the run time calculation of a lower bound for the globally optimal tour. Assertions (ii) and (iii) allow for estimates of an upper bound for the globally optimal tour, for example

$$wt(t_g) \leq \frac{4}{\sqrt{N}}wt(t).$$

Assertion (iv) provides a very coarse order of magnitude estimate for the depth of a 2-change iteration and hence of the degree of the fundamental polynomial.

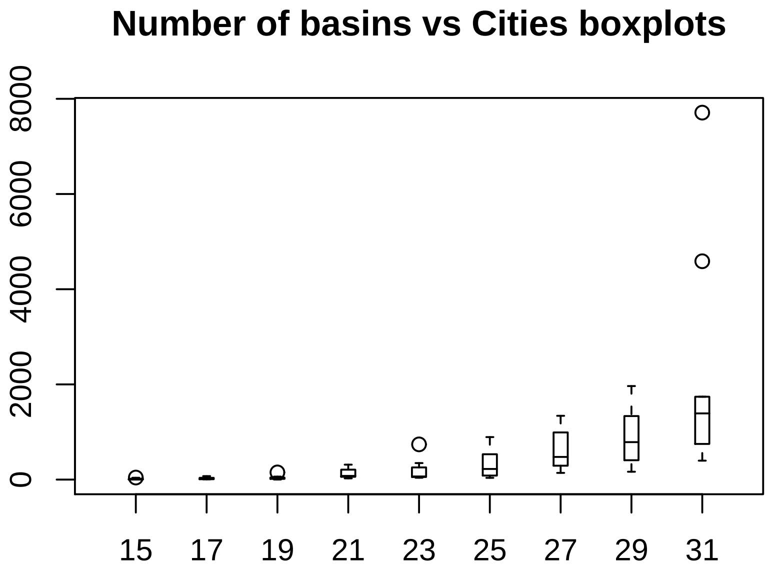


Fig. 4

Unfortunately these results do not speak to the nature of the basins due to the 2-change topology. However in the next two figures we present some empirical results about this for the case that the cities are chosen uniformly at random in the square.

In Fig. 4 we show how the mean and distribution of the number of basins varies with the number of cities for the $\mathcal{I}_N$ problem.

In Fig. 5 we present some statistical results for θ versus the number of cities for the $\mathcal{I}_N$ problem.

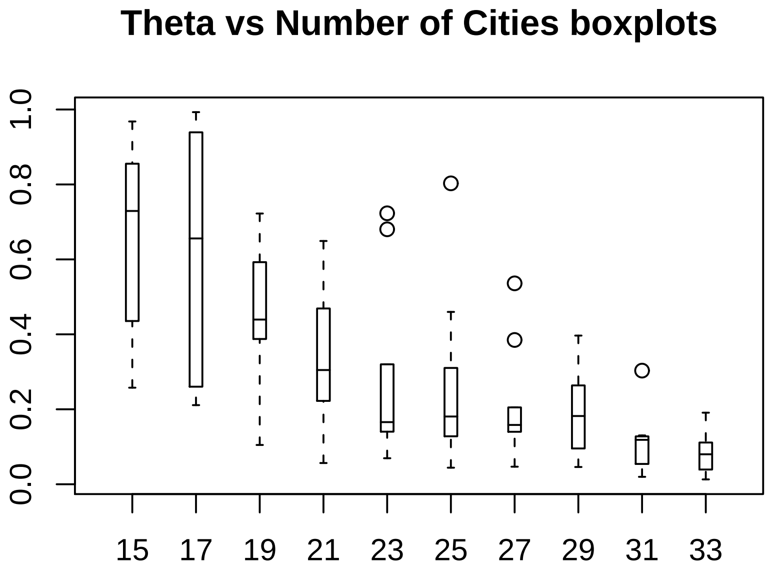
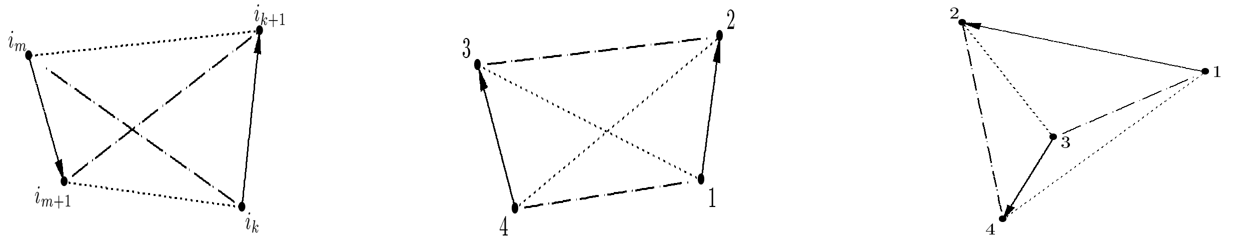


Fig. 5

Cities lying on a convex polygon

It will be helpful to make some observations about the geometry under which PPR improves the tour. Besides the geometrical relationship between the new and old links depicted in Fig. 2, there are, conceptually, two other possibilities as shown in Fig. 6a,b,c. Descriptively, the 3 possible geometrical arrangements of the cities and links are: (1) the original links intersect as in Fig. 2, (2) the replacement links intersect as in Fig. 6a, and (3) neither of these, for example, Fig. 6b or 6c. We refer to Case 1, (i.e. Fig. 2) as the *butterfly* case.



(a) new links cross (b) external "links" cross (c) another configuration

Fig. 6

Theorem 10. *If the 2-change edges of the original tour intersect, the butterfly case, then the modified tour is strictly shorter. Similarly, if the replacement edges intersect, case 6a, then the original tour is shorter.*

Proof. If the edges of the original tour intersect, then, together with the replacement edges, two triangles are formed. The original edges form two sides of each triangle and the replacement edges form the third side of each, see Fig. 2. The result now follows by the triangle inequality. ▮

Corollary 2. *A 2-opt tour has no intersecting edges.*

Theorem 11. *If the cities lie on the hull of a closed convex set, then the globally optimal tour is any parameterization of the hull and every tour improves under PPR to the globally optimal tour.*

Proof. Assume not and let tour t improve to a local minimum different from $1, 2, \ldots, N, 1$. We may assume without loss of generality that t is this local minimum tour. Starting with city 1, let city i be the first not to link with city $i + 1$. We may assume without loss of generality that $i = 1$. Then 1 links to k for $k \neq 2$ and $k \neq N$ (where N labels the city at the end of the tour linking back to 1).

By convexity and the hypothesis, the extended line $\overline{1k}$ bisects the convex hull in such a way that city 2 is on one side, or on, this line and city N is on the other. Hence some link must join a city on the 2 side, say j, with a city on the N side, say m. Again by the convexity hypothesis, the links $\overline{jm}$ and $\overline{1k}$ intersect. Therefore tour t cannot be locally minimal. ▮

Theorem 12. *Given an N-city problem lying on a convex polygon, an upper bound for the number of butterfly 2-changes required to achieve optimality is the number of self-intersections of the tour.*

Proof. From above, a tour is optimal if and only if it has no self intersections.

In the following we will use only butterfly type improvements. We note that each butterfly 2-change reduces the number of self intersections by at least 1. This is because, (see Fig. 2) any tour link that intersects one of the improved edges, must also intersect at least one of the original edges. Further, any link that intersects both improved edges, also intersects both original edges. Then since the new edges do not intersect while the original ones do, it follows that the improvement has at least one less intersection. ▮

Lemma 1. *Given an N-city problem, an upper bound for the number of butterfly 2-changes required to achieve optimality is $(N-3)N/2$.*

Proof. Note that any given edge can intersect at most $N-3$ other edges because it cannot intersect itself nor the two edges incident on its end-points. Since each intersection is counted twice the number of intersections per edge is $(N-3)/2$ per edge. As the number of edges of a tour equals the number of cities, we obtain the result. ▮

Remark 3. Note that if N is odd, $N = 2k+1$ and the cities lie on a convex polygon, then the maximum number of intersections is achieved when each edge skips $k-1$ cities.

In Fig. 7 we present some statistical results on the number of self-intersections and the number of descent steps for cities arranged on a convex polygon versus the number of cities.

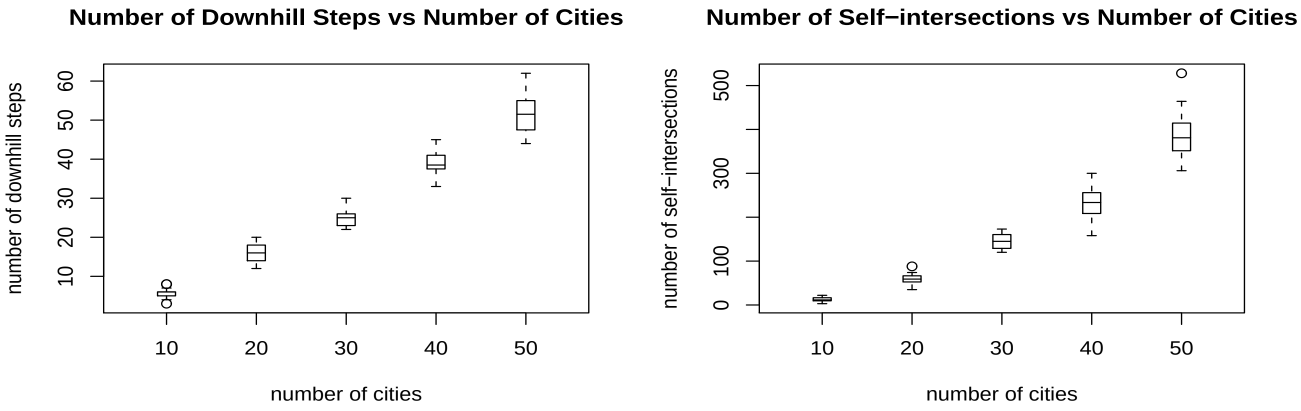
Combining these figures we get Fig. 8.

Hyperbolas

Now imagine the cities A, C, D are fixed but X is variable. We want to know where in the plane it is that a 2-change will replace $AX + CD$ by $AD + CX$, we refer to this as the *replacement region.* The complementary set is where the original links are maintained and we refer to it as the *maintenance* or M-region. Evidently

$$\text{replacement region} = \{X : |AX| + |CD| \geq |AD| + |CX|\}. \tag{20}$$

Assume first that $|AD| > |CD|$ and put $R^2 = |AD| - |CD|$. The set of points $\{X : |AX| - |CX| = R^2\}$ is a hyperbola with focii A and C. The two branches of the



(a) downhill steps

(b) self-intersections

Fig. 7

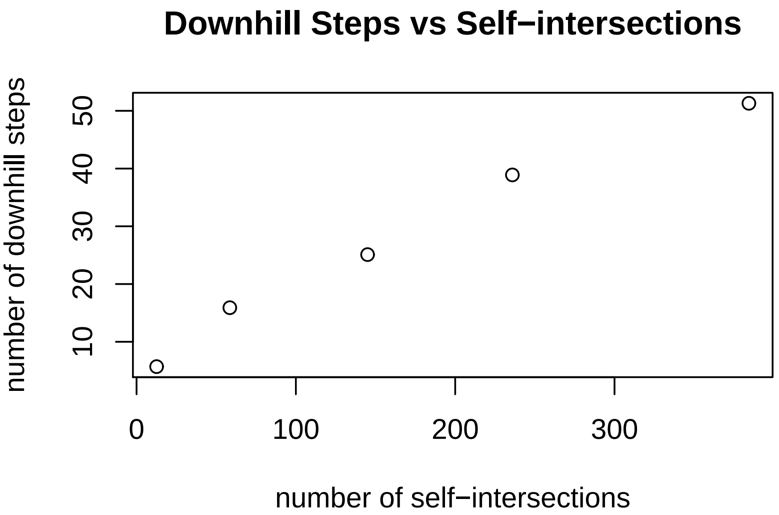


Fig. 8, Number of downhill steps vs the number of self-intersections

hyperbola partition the plane into 3 regions, one containing A, another containing C and the region between the two branches. By continuity, the sense of the inequality in (20) is constant throughout each region. Direct substitution shows that D itself lies on the hyperbola and in fact on the branch closest to C by our assumption.

It is easy to see that X may be selected in the hyperbolic region containing C in such a way that the edges AX and CD intersect hence this region belongs to the replacement region. But crossing the branch of the hyperbola reverses the sense of the inequality in (20) and therefore this is the entire replacement region. This situation is shown as the shaded portion in Fig. 9a, the maintenance region is the unshaded part.

In the degenerate case that D lies on the x-axis, then the maintenance region is the entire plane except the portion of the major axis of the hyperbola from $\min(D, C)$ and to the right.

If $|CD| > |AD|$ then the situation is as shown in Fig. 9b; D now lies on the branch closest to A. In general the replacement region extends from the branch containing D and the hyperbolic regions toward the focus C.

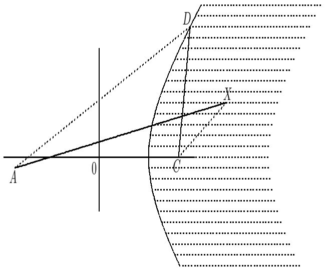


Fig. 9a, $|AD| > |CD|$

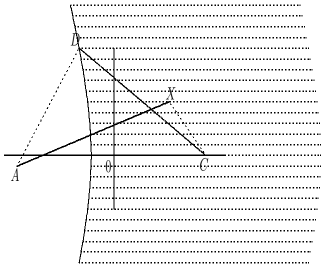


Fig. 9b, $|AD| < |CD|$

Adding cities to a TSP

Much light can be shed on a TSP by following a strategy of deleting cities one-by-one arriving at a "core" subset. If done with care, this core will share properties with the original city set. For one, PPR descent paths between the city sets at each stage will be similar only differing in the form of "detours" to the added city. As a consequence, θ will be approximately equal between the two.

We proceed in the opposite direction, we will investigate adding cities one-by-one to a TSP.

Definition 7. Let $\mathcal{C}_N$ be an N city TSP and t a tour which includes the link AB joining adjacent cities A and B. A point in the plane X, distinct from $\mathcal{C}_N$, along with the links $AX + XB$ in place of AB will be called a *detour*.

Theorem 13. *Let AB be a link of an optimal tour, globally or locally with respect to 2-change, for an N city TSP $\mathcal{C}_N$. Then there is an open region M containing the open interval AB^o such that if a new city X is placed in M, then the detour AXB is optimal, globally or locally, for the $N+1$ city TSP $\mathcal{C}_{N+1}$.*

Proof. In the case of global optimality, for any tour t_i of $\mathcal{C}_{N+1}$, let $f_i(X)$ denote its length as a function of X. Then f_i is continuous in X. The length, $f(X)$, of the optimal tour as a function of X is the minimum of the f_i, therefore f is also continuous. But if X is on AB^o, then t is optimal. The conclusion follows by continuity.

For the case of local optimality, we must show that the detour solution is 2-change optimal. Let CD be a candidate link for 2-change replacement along with AX in favor of CX and AD. It is assumed that CD is oriented so that A joins C externally. The 2-change will be accepted if the inequality $|AX| + |CD| > |CX| + |AD|$ obtains. Since we must also compare the link XB with CD, the modification of the above requires replacing f_i by

$$f_{CD}(X) = \min\{|AX| + |CD| - |CX| - |AD|, |BX| + |CD| - |CX| - |BD|\}.$$

As before, for each link CD, f_{CD} is continuous and negative for X on AB^o. Hence for each CD there is an open region M_{CD} containing AB^o in which f_{CD} is negative. The intersection of these regions over all links CD gives the required open region M. See Fig. 10. ∎

Remark 4. The maintenance region is especially large on the side of AB away from the city set (for a link having an away side), see Fig. 10.

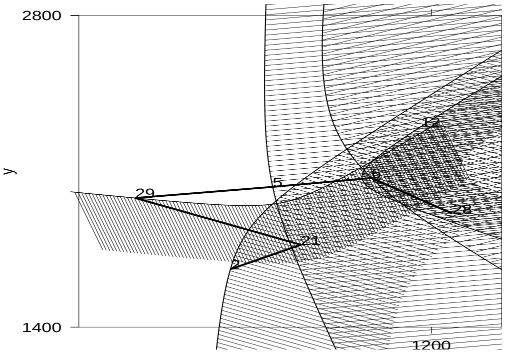


Fig. 10

Theorem 14. *Let AB be a link of a locally optimal tour τ for an N city TSP $\mathcal{C}_N$ and let X be a point on the segment AB. Then τ', the τ induced detour AXB, is locally optimal for the $N+1$ city TSP, $\mathcal{C}_N \cup \{X\}$. Further let t be a tour of $\mathcal{C}_N$ in the basin of τ which contains AB and whose 2-change descent preserves this link at each step. Then starting with t', the AXB detour of t, the 2-change descent sequence is the AXB detour of the descent sequence for t.*

Proof. That τ' is optimal is easy to see since by adding a city the tour length cannot decrease. But the detour AXB solution to $\mathcal{C}_{N+1}$ has the same length as the solution τ to $\mathcal{C}_N$. So it must be minimal.

For the second part, we show that the detour AXB is maintained at each step. Consider a 2-change attempting to break the link (oriented) AX (or XB), and let CD be the (oriented) companion link to be replaced. The 2-change is an improvement if

$$|CX| + |AD| < |AX| + |CD|.$$

Adding $|XB|$ to both sides gives

$$|XB| + |CX| + |AD| < |XB| + |AX| + |CD| = |AB| + |CD|.$$

But we know that $|AB| + |CD| < |BC| + |AD|$ for otherwise the links AB and CD would be replaced by BC and AD in the N city descent. (Moreover AB remains intact under 2-change with respect to any other link $C'D'$ as well.) Hence

$$|XB| + |CX| + |AD| < |BC| + |AD|.$$

This implies that $|XB| + |CX| < |BC|$ which is false by the triangle inequality. Hence link AX is maintained. For the same reason, so is XB. ▮

Corollary 3. *By continuity of the 2-change improvement function $f(X)$ as introduced above, the result holds in some neighborhood of the segment AB^o as well.*

Remark 5. Remarkably, adding a city along an optimal link can have a large effect on θ and it could go either up or down. The globally optimal tour is shown for a 9 city problem in Fig. 11; $\theta = .62$ for this problem. But adding city 10 near city 9 and between 9 and 4 makes for a 10 city problem with $\theta = .52$ while adding the 10th city between cities 2 and 5 yields a problem with $\theta = .73$.

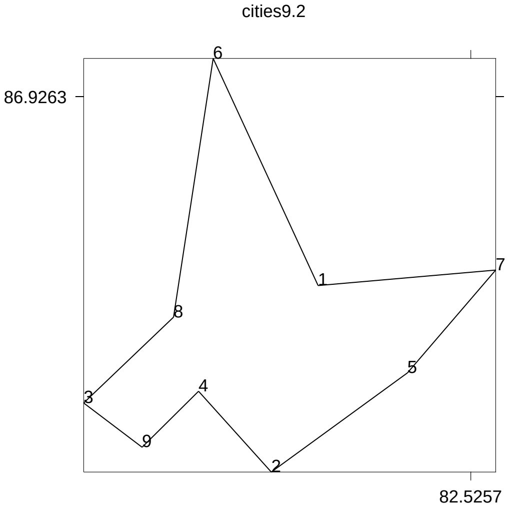


Fig. 11, adding cities along link 94 or 25 greatly changes θ

In Fig. 12 we illustrate the remarkable quality of PPR to produce a very characteristic distribution for the magnitudes of the coefficients of the fundamental polynomial, and hence the distribution for the number of steps from a local minimum, despite wide variations in θ (and in N for that matter as well). Illustrated is the coefficient distribution for three different 29 city problems from $\mathcal{I}_{29}$. Since the third problem illustrated has a large θ, its density comes in below that of the other two, but if normalized, the distributions are nearly identical. Although we show only four examples in this figure, all coefficient distributions we have produced are similar in that the great majority of descent paths are of intermediate length, that is, PPR basin trees are bushy. Results sheding light on the PPR coefficient distribution is an open problem.

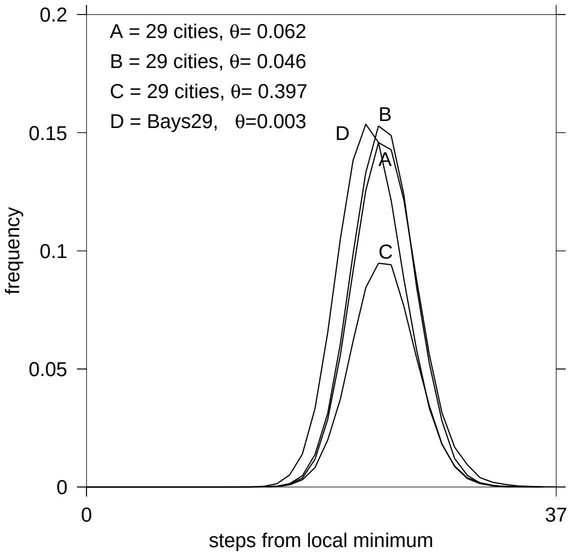


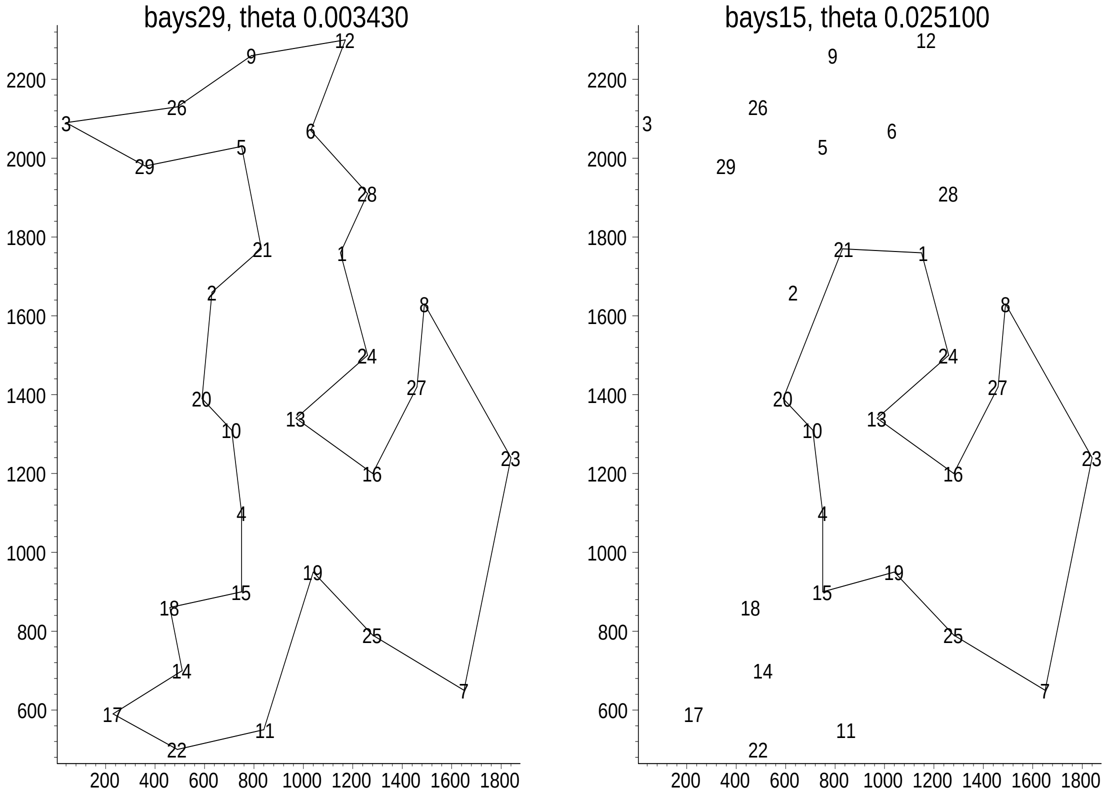
Fig. 12, 29 city distributions

§5 Application to the Bays29 Problem

The Bays29 problem is a standard problem available on the Traveling Salesman Problem database at *http://softlib.rice.edu/softlib/catalog/tsplib.html.* The figures below show the application of the above principles, in reverse, to reduce the problem from 29 cities to 15. The number of tours of the original problem is $28!/2 = 1.5 \times 10^{29}$ while the number for the reduced problem is $14!/2 = 4.3 \times 10^{9}$. This is a 10^{20} magnitude reduction. The global basin size for Bays29 is $\theta = 0.0032$ while that for the reduction is $\theta = 0.025$ or a 10 fold difference. Thus the technique can have considerable efficacy.

To demonstrate the technique, first the optimal tour for the full size problem must be guessed since, in general, that will not be known. (However, once the reduced city set is reached, the results of the last section can be applied, step by step, to build back to the original problem as a check.) Then a city is selected for removal. In accordance with the theorems of the last section, we seek a city lying on a detour link or in the hyperbolic maintanence region of the remaining city set. In the Bays29 problem we chose city 9. Having removed a city, repeat the foregoing until there are no remaining suitable candidates for removal. As a check on the removal, we estimated θ at each step via the Monte Carlo method. Having worked down to 15 cities, we were then able to calculate the optimal tour exactly and see that it was what we had arrived at.

At this point one can use θ for the reduced city set as an estimate for the original, or better, use the trend analysis of the Monte Carlo estimates obtained during the reduction. In Fig. 13 we show the original as well as the final city sets along along with the optimal tours.



(a) Original Bays29

(b) Reduced problem to 15 cities

Fig. 13

Our objective here has been to estimate θ, the global basin size parameter. In our experiments we often observed that θ for the goal basin is larger than the basin size for non-goal basins. In such an event, estimates obtained in this way are underestimates for θ.

In Table 2 we show the results of the step by step removal of carefully chosen cities according to the guidance of the previous sections. We indicate which city is removed at each step and the estimated value of θ for the reduced problem.

In Fig. 12, curve “D”, we show the coefficient distribution.

Table 2

Table 2 θ vs City Removal for Bays29

number cities	city removed	θ	size Ω
29	–	.0032	1.6×10^{29}
28	9	.0049	5.4×10^{27}
27	26	.0053	2.0×10^{26}
26	22	.0062	7.8×10^{24}
25	14	.0061	3.1×10^{23}
24	3	.0066	1.3×10^{22}
23	12	.0072	5.6×10^{20}
22	17	.0101	2.6×10^{19}
21	18	.0217	1.2×10^{18}
20	11	.0135	6.1×10^{16}
19	29	.0229	3.2×10^{15}
18	5	.0298	1.8×10^{14}
17	6	.0341	1.0×10^{13}
16	2	.0268	6.5×10^{11}
15	28	.0251	4.4×10^{10}

Acknowledgement

The authors would like to thanks the referees for their helpful suggestions and for the timelyness of their review.

References

[1] Mendivil, F.,Shonkwiler, R.,Spruill, C., *Restarting Search Algorithms with Applications to Simulated Annealing*, Advances in Applied Probability, Vol. 33, pp. 242–259, 2001.

[2] Hajek, B., *Cooling schedules for optimal annealing*, Math. Operat. Res. Vol. 13, No. 2, pp. 311–329, 1988.

[3] Chiang, T. and Chow, Y., *On the convergence rate of annealing processes*, SIAM J. Control and Optimization, Vol. 26, pp. 1455–1470, 1988.

[4] Catoni, O., *Rough large deviation estimates for simulated annaling - application to exponential schedules*, Ann. of Prob. Vol. 20, pp. 1109–1146, 1992.

[5] Shonkwiler, R. and Van Vleck, E., *Parallel Speed-up of Monte Carlo Methods for Global Optimization*, J. of Complexity Vol. 10, pp. 64-95, 1994.

[6] Boender, G. and Rinnooy Kan, A., *Bayesian Stopping Rules for Multistart Optimization Methods*, Math. Programming, Vol. 37, pp. 59–80, 1987.

[7] Azencott, R., *Simulated Annealing, Parallelization Techniques*, John Wiley and Sons, New York, New York, (1992).

[8] Lin, S., and Kernighan, B. W., *an Effective Heuristic Algorithm for the Traveling Salesman Problem*, Operations Research, Vol. 21, pp. 489–516, 1973.

[9] Chandra, B., Karloff, H., Tovey, C., *New Results on the OId k-Opt Algorithm for the TSP*, SIAM J. Comp., Vol. 28, pp. 1998–2029, 1999.

[10] Kern, W., *A Probabilistic Analysis of the Switching Algorithm for the Euclidean TSP*, Math. Programming, Vol. 44, pp. 213–219, 1989.

[11] Kirkpatrick, S., Gelatt, C., Vecchi, M., *Optimization by simulated annealing*, Science Vol. 220, pp. 671-680, 1983.