



# Dynamics of Local Search Trajectory in Traveling Salesman Problem

WEIQI LI

*School of Management, University of Michigan-Flint, 303 East Kearsley Street, Flint, Michigan 48502, USA*  
email: weli@umflint.edu

*Submitted in March 2002 and accepted by Edmund Burke in July 2005 after 4 revisions*

## *Abstract*

This paper investigates dynamics of a local search trajectory generated by running the Or-opt heuristic on the traveling salesman problem. This study evaluates the dynamics of the local search heuristic by estimating the correlation dimension for the search trajectory, and finds that the local heuristic search process exhibits the transition from high-dimensional stochastic to low-dimensional chaotic behavior. The detection of dynamical complexity for a heuristic search process has both practical as well as theoretical relevance. The revealed dynamics may cast new light on design and analysis of heuristics and result in the potential for improved search process.

**Key Words:** heuristics, local search, traveling salesman problem, dynamical complexity

## **1. Introduction**

Combinatorial optimization studies hard optimization problems in which there can be many possible solutions. In general, the primary task is to find the best element in a finite solution set, with respect to some objective function. Many combinatorial optimization problems are known to be NP-hard, which means that there is no known algorithm that can solve every instance of this problem in a time growing less than a power of problem size, and it is unlikely that such an algorithm can be found (Garey and Johnson, 1979; Papadimitriou, 1994). Researchers have used many strategies to tackle combinatorial optimization problems (Papadimitriou and Steiglitz, 1999). The NP-completeness and intractability of combinatorial optimization problems have led many researchers to employ heuristic search techniques. A *heuristic* seeks good solutions at a reasonable computational cost without being able to guarantee either feasibility or optimality, or even in many cases to state how close to optimality a particular feasible solution is (Reeves, 1993).

There are three important aspects in the analysis of heuristics: search performance, time complexity, and dynamical complexity. A large body of literature has been focused on measurement of performance and determination of time complexity. The third aspect is related to the *behavior* of the heuristic search process. There has been some work in studying search mechanisms or problem structures over the past years to touch this aspect in this way or another (Evans, 1987; Grover, 1992). Some researchers have been developing landscape theory to study dynamics of landscapes (Kauffman and Levin, 1987; Weinberger, 1991).

Here the landscape means the set of allowable configurations in some optimization problem. The  $N - k$  model of Kauffman and Levin (1987) and Kauffman (1993) defines a family of fitness landscapes. Heuristically, the  $N - k$  model is an attempt to understand the dependence of a system with  $N$  elements, on a parameter  $k$  which measures how richly interconnected the elements of the system are (Weinberger, 1991). Weinberger (1990) proposed the correlation functions to define the correlation of points in the search space. Landscape theory has emerged as an attempt to devise suitable mathematical structures for describing the “static” properties of landscapes and their influence on the dynamics of adaptation, and for investigating the connections of landscape theory with algebraic combinatorics and random graph theory (Reidys and Stadler, 2002). The notion of fitness landscapes has been introduced to describe the dynamics of heuristic search methods in optimization (Merz and Freisleben, 1999). Several researchers use fitness distance analysis to investigate the structure of fitness landscapes (Macken, Hagan, and Perelson, 1991; Fontana et al., 1993), identify landscape characteristics which influence the effectiveness of heuristic search methods (Jones and Forrest, 1995), and prove convergence of simulated annealing (Sorkin, 1991). The landscape theory focuses on the dynamics of landscapes, e.g. on the behavior of local optimization on different configuration spaces.

However, the dynamics of heuristic search process is not well understood because the complexity of the search process makes the phenomenon not amenable to simple analysis and leads to differing views on modeling techniques and analytical methods. The major theme of this paper is to study the dynamical behavior of a particular search trajectory in attempts to come to grips with the dynamical complexity of the search process. Specifically, this study quantitatively characterizes the temporal dynamics of a local search process by calculating the dimension of the local search trajectory. The dimensionality of the search trajectory is closely related to the dynamics of the search process. A better understanding of the heuristic search behavior may result in the potential for improved search heuristics.

## 2. The search trajectory

Since most heuristics are based upon intuition and involve randomness, the behavior of a heuristic search process looks erratic, much like the behavior of a system strongly influenced by random “noise” or the complicated behavior of a system with many degrees of freedom. The technical approach of this study views the heuristic search process as a *nonlinear dynamical system*. A heuristic search system takes a problem instance as input, use a heuristic algorithm or hybrid algorithm to search for solutions, and outputs the best solution found during the search process. All local search systems have the iterative descent property when they apply to minimization problems. Typically, a local search algorithm starts off at an initial configuration and then generates a sequence of iterations. Each iteration consists of a possible transition from the current configuration  $c_i$  to a new configuration  $c_j$  selected from the neighborhood  $N(c_i)$  of  $c_i$ . If this neighboring configuration has a better score, the current configuration is replaced by this better configuration; otherwise another neighboring configuration is selected and compared with the current one. The process is repeated in this fashion, generating a series of configurations. This neighborhood search property makes a

local search system locally convergent, that is, it generates a sequence that converges to a point that may be not an optimal configuration.

When describing a system, we usually utilize the “state” to model the dynamics of the system. The *state* is a vector that sufficiently describes the condition of the process or the position of the system in the state space. A state variable is the characteristic variable that directly measures the internal dynamics of the system. A heuristic search system uses the state to determine the possible changes in the position of configuration space or solution space. For example, for the traveling salesman problem (TSP), the state variable is the length associated with a tour, which is used to control the search process in the solution space. The state variable is formulated in such a manner that at any point of time during the search, it measures the performance of the search process in the current position of the solution space.

Typically, a local heuristic search system selects an initial configuration  $c_0$ , which is measured by the state value  $s_0$ , in the configuration space  $C$ . Then the system generates a new configuration  $c_1$ . If the state  $s_1$  of  $c_1$  has an improved value, the search process keeps  $c_1$ , and a new configuration  $c_2$  is generated from  $c_1$ . If  $c_1$  does not have an improved state value, the search process discards it. A new configuration  $c_2$  is still generated from  $c_0$ . The process is repeated and a new configuration  $c_3$  is generated and its state  $s_3$  is tested. Continuing in this way, a sequence of state points  $S = \{s_0, s_1, \dots, s_i, \dots, s_n\}$  is generated. A *trajectory* is the path in state space followed by the process as it evolves with time. The set  $S$  defines the *search trajectory* generated by a local search system in the state space.

The behavior of a dynamical system is typically described by plotting its trajectory in a state space. The general philosophy of this approach is to extract “physical sense” from an experimental data. In fact, the geometrical object described by the points of the trajectory is an infinite set of points (in practice we only use a subset data) in a  $d$ -dimension space. However, many dynamical processes can be observed only through a time series, which is a sequence of data representing the evolution of an observable quantity of the process, indexed by time. Nonlinear dynamics theory shows that a set of sampled values of just one variable can be sufficient to capture the features of a system’s dynamics (Tong and Lim, 1980). In other words, an observed time series often contains the dynamical information of the experimental process, and one can reconstruct the information given in the time series to present information about the whole system. In 1980, David Ruelle suggested the idea of using the value of the coordinate with its values at two previous time, e.g.  $s(t)$ ,  $s(t - \tau)$  and  $s(t - 2\tau)$ , to represent the state space of a dynamical system in a 3-D space (Packard et al., 1980), where  $\tau$  is the time interval. This method, *called time-delay coordinates reconstruction*, has been become a powerful analysis tool for investigating dynamical systems in many scientific and research communities.

In general, we can provide the required coordinates to reconstruct the trajectory for a dynamical system in  $d_E$  dimensions (Abarbanel et al., 1993), where  $d_E$ , called the *embedding dimension*, denotes the dimension of the coordinates for the reconstructed trajectory. Rather than considering  $s(t)$  as a sequence of single value, we group together  $d_E$  successive values for each time  $t$  in the time series, thus the vector

$$v(t) = [s(t), s(t + T), s(t + 2T), \dots, s(t + (d_E - 1)T)] \quad (1)$$

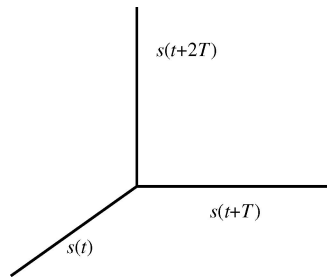


Figure 1. Three-dimensional coordinates for vector  $v(t) = [s(t), s(t + T), s(t + 2T)]$ .

represents a point in a  $d_E$ -dimensional space whose coordinates are just these  $d_E$  values, where  $T$  denotes a time delay that must be a multiple of time interval. When we plot these points in  $d_E$ -dimensional space with connecting line segments, we can obtain the reconstructed trajectory. The interesting feature of the time-delay reconstruction is that the reconstructed trajectory in  $d_E$ -dimension is topologically equivalent to the actual trajectory of the system (Takens, 1981). For example, figure 1 illustrates a system of 3-D time-delay coordinates on which the following sequence of vectors can be plotted:

$$\begin{aligned}
 & [s(0), s(T), s(2T)] \\
 & [s(1), s(1 + T), s(1 + 2T)] \\
 & [s(2), s(2 + T), s(2 + 2T)] \\
 & \dots \\
 & [s(t), s(t + T), s(t + 2T)]
 \end{aligned} \tag{2}$$

Theoretically, dynamical trajectories can always be faithfully reconstructed using this procedure (Takens, 1981).

The choice of the time-delay  $T$  is almost arbitrary (Takens, 1985; Casdagli et al., 1991). However, in practice, if  $T$  is too small, the coordinates  $s(t)$  and  $s(t + T)$  that we wish to use in the reconstruction of data vector  $v(t)$  will not be independent enough. The resulting trajectory will be close to the “diagonal” of the state space. On the other hand, if  $T$  is very large, any connection between the measurement  $s(t)$  and  $s(t + T)$  will be numerically subject to being random with respect to each other. The reconstructed trajectory will appear to wander all around space such that the structure of the trajectories is hard to detect. Thus, in practice, the quality of the reconstruction depends on the value chosen for  $T$ . Therefore, some prescription is needed to identify a proper time-delay  $T$  that is large enough that  $s(t)$  and  $s(t + T)$  are quite independent, but not so large that they are completely independent in a statistical sense. There are a number of suggested techniques for choosing  $T$  (Buzug, Reimers, and Pfister, 1990). The *average mutual information* approach is a commonly-used method to obtain a proper, but not necessarily optimal, time-delay  $T$  (Frazer and Swinney, 1986; Shaw, 1985).

### 3. The dynamics of the heuristic search trajectory

#### 3.1. The correlation dimension

In characterizing a dynamical process quantitatively, the emphasis usually is on the time-dependent dynamical behavior of the system. One approach to characterize the temporal dynamics of a system is to calculate the dimension of the process trajectory. The dimensionality of the process trajectory in a dynamical system is closely related to the dynamics of the system (Hilborn, 1994). Dimensionality can be a measure of the complexity of a dynamical system, which gives us an estimate of the number of active degrees of freedom in the system. This study uses the correlation dimension technique to investigate the dynamical behavior of a particular local search trajectory. In this way we can extract a multidimensional description of dynamics of the local heuristic search system.

Generally speaking, the analysis of dynamical behavior is extremely difficult. In scientific experiments, the behavior of an experimental system is usually ruled by  $k$ -coupled autonomous differential equations, and the state space for the system is  $\mathfrak{R}^k$ . Each vector  $v_i$  represents a possible state of the process. Of course, fundamental understanding of a dynamical process can be achieved by building a dynamical model from the whole set of differential equations. However, tracking the differential equations is not easy. In an experimental setting, it is seldom the case that all relevant dynamical variables can be measured. The set of coupled differential equations that governs the behavior of the process would be unknown, or the dimension  $k$  of state space would be very large (Alligood, Sauer, and Yorke, 1997). Thus, it is very difficult to fully model the dynamics of the process.

The determination of the fractal dimension of trajectory has become a standard diagnostic tool in the analysis of a dynamical system. The dynamical complexity of a system can be described by the calculated dimension of its process trajectories. Most recent work in characterizing dynamical process has focused on the evaluation of fractal dimension by calculating the correlation dimension. The correlation dimension is originally due to Grassberger and Procaccia (1983). It has become a popular technique because of its simplicity, computational speed and minimal storage requirement.

The theoretical basis for the correlation dimension is the embedding theory, which describes the connections between system's full state space, the measurements that comprise a time series, and the reconstructed state space. Takens (1981) proposed a theorem showing that the measurement in time series of a single variable suffices for the characterization of the dynamics of the system evolving in a multivariable space. Takens' Theorem is a crucial development responsible for the realization that multidimensional state space could be reconstructed from measurement of a single scalar time series. For detailed discussions on the embedology theory and Taken's Theorem, one can reference (Casdagli, Sauer, and Yorke, 1991; Ott, Sauer, and Yorke, 1994; Takens, 1981).

Correlations between points on a trajectory are defined in terms of spatial correlation that can be formally measured by the Euclidean distance. The correlation dimension considers the statistics of distances between pairs of points (Parker and Chua, 1990). The procedure of estimating the correlation dimension from a single variable time series is as follows (Takens, 1985):

- (1) The time series are embedded in successively increasing dimensions. The series of scalars is converted into a series of vectors in each embedding.
- (2) For each embedding, the correlation integral  $C(r)$  is computed and then the correlation exponent  $k$  is estimated.
- (3) The procedure is repeated until the estimates of  $k$  converge.
- (4) Whatever value at which  $k$  remains stable over a number of embedding dimensions is taken as the estimate of the correlation dimension  $D_C$  of the underlying trajectory.

If a time series  $s(t)$ ,  $t = 1, 2, 3, \dots$ , is used to create the  $d_E$  embedding dimension, the series of single scalars is converted into a series of vectors with overlapping entries:  $v_t = [s(t), s(t+T), s(t+2T), \dots, s(t+(d_E-1)T)]$ . Let  $V = \{v_0, v_1, \dots, v_n\}$  be a trajectory of the map  $f$  on  $\mathcal{N}^{d_E}$ . One can measure the spatial correlation among the points on the trajectory by calculating the correlation integral,  $C(r)$ . More specifically, for each  $r > 0$ , the correlation function  $C(r)$  is defined as the average number of pairs of data vectors  $(v_i, v_j)$  which are separated by a distance less than  $r$  in the  $d_E$  dimension.

$$C(r) = \lim_{n \rightarrow \infty} \frac{1}{n^2} \{\text{the number of pairs } (v_i, v_j) \text{ such that } \|v_i - v_j\| < r\} \quad (3)$$

where  $n$  denotes the number of points on the trajectory in the  $d_E$  dimension. In principle  $n$  should go infinity, but in practice  $n$  is limited by the availability of the time series points. This places limits on possible values for  $r$  and  $d_E$ . And  $\|v_i - v_j\|$  denotes the distance induced by the selected norm. According to Theorem 2.4 of Brock (Brock, 1986), the correlation dimension is independent of the choice of norm. Thus, the Euclidean distance is used in this study.

Intuitively,  $C(r)$  measures the probability that any particular pair of points in the trajectory is close in the  $d_E$  dimension. Grassberger and Procaccia (1983) have established that the true spatial correlation grows according to the power law:

$$C(r) \approx r^k \quad (4)$$

where  $k$  is the correlation exponent. If  $k$  tends to be a constant as embedding dimension  $d_E$  increases, then  $k$  yields an estimate of the correlation dimension of the trajectory. In this case, the trajectory is consistent with deterministic behavior. If  $k$  increases without bound as the embedding dimension  $d_E$  increases, this suggests that the trajectory is stochastic as the dimension of the dynamical process approaches infinity. Therefore, the correlation dimension  $D_c$  is defined to be the number that satisfies

$$C(r) = r^{D_c} \quad (5)$$

Then the correlation dimension  $D_c$  is obtained by

$$D_C = \frac{\ln C(r)}{\ln r} \quad (6)$$

In the Eq. (3),  $C(r)$  is defined as function of  $r$  such that  $C(r)$  increases from 0 to 1 as  $r$  increases from 0 to  $\infty$ . In other words, if all pairs of points fall within the distance  $r$  of each other, then  $C(r) = 1$ . If  $r$  is smaller than the smallest distance between trajectory points, then  $C(r) = 0$ . Since any real data set consists of a finite number of points, it is not possible to take the limits  $n \rightarrow \infty$ . Hence, in practice the correlation dimension is found using some range of  $r$  values and plotting  $\ln C(r)$  as a function of  $\ln r$ . We can observe that only in some intermediate  $r$  region does  $C(r)$  obey the power law expressed in the equation (5) (Holzfuss and Mayer-Kress, 1986). The region is called the scaling region, and the curve in this region is approximately a straight line. The slope for the scaling region determines the correlation dimension  $D_c$  for a particular  $d_E$ .

Since in practice the presence of noise in the data is likely to result in a choppy appearance in the curve of  $\ln C(r)$  versus  $\ln r$ , one often needs to run a linear regression on the curve. The slope estimate from this regression is the correlation dimension estimate for this value of  $d_E$ . Therefore, there are some uncertainties involved in calculating  $D_c$ , which can cause the computed dimension to differ slightly from the “true” value.

This procedure is repeated for a sequence of increasing values of  $d_E$ . If the time series are purely random, then in an infinite data set, the correlation dimension will equal  $d_E$  for all  $d_E$ . In practice, with a finite data set, the correlation dimension estimate may rise with  $d_E$  at less than a one-for-one rate (Ramsey and Yuan, 1987). If the time series are deterministic, the correlation dimension  $D_c$  should initially increase with increasing values of  $d_E$ . However, at some point  $d_E$ , it should level off and remain constant for all further values of  $d_E$ . One should observe a “plateau” on the plot of the correlation dimension  $C(r)$  in the function of embedding dimension  $d_E$ . This “saturation” value of the slope is the correlation dimension estimate for the underlying system that generates the data. These relationships are illustrated in figure 2.

In practice, the slope estimate may never completely stop rising. Thus, the process of determining whether saturation has occurred can, in general, be rather judgmental. Ding et al (1993) reported that, for large enough data set, the saturation begins when the number of embedding dimension  $d_E$  first exceeds the correlation  $D_c$ . However, a lack of sufficient

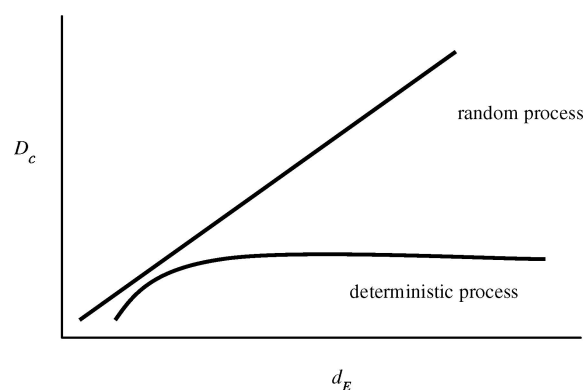


Figure 2. The correlation dimension for random and deterministic processes.

data will delay the saturation onset. Takens' embedding theorem provides a practical clue, which shows that if the true system that generated the time series is in  $d$  dimension, the embedding space need to have  $d_E \geq 2d + 1$  dimension to capture completely the dynamics of the underlying system. Since  $d$  is unknown, an increasing sequence of  $d_E$  values must be tried. Usually  $d_E$  has to be increased to nearly  $2d + 1$  before the value deduced for  $d$  become independent of  $d_E$ .

### 3.2. *The empirical results*

In general, an instance of an optimization problem consists of all the inputs that are needed to compute a solution to the problem and satisfies whatever constraints imposed in the problem statement. This study chooses the TSP to serve as a problem testbed. The TSP is a NP-complete problem that has provided much motivation for the development of complexity theory. Given a set of  $n$  cities and an  $n \times n$  distance matrix  $D$  in which  $d(i, j)$  denotes the distance between city  $i$  and  $j$ ,  $i, j = 1, \dots, n$ . A *tour* is a closed route that visits every city exactly once. A salesman tries to find a tour with minimal total length.

When comparing a proposed approach with other approaches, many researchers use a test instance drawn from a collection of publicly available cases with documented optimal or best-known solutions, or they generate an instance of metric TSP in which the expected length of a minimal tour for the TSP instance can be calculated by using an empirical formula (Bonomi and Lutton, 1984). Since this study does not intend to solve a TSP, but rather focuses on the analysis of search trajectory. To reduce computing time and storage requirement, this study generates a general symmetric TSP instance, which consists of 1000 cities with distance  $d(i, j)$  independently drawn from a uniform distribution of the integers over the interval  $[1, 1000]$ .

There are a huge variety of local search algorithms for the TSP (Johnson, 1990). One of the earliest applications of local search algorithms for the TSP is  $r$ -opt heuristics, where  $r$  is the number of edges exchanged at each iteration. The larger the value of  $r$ , the more likely it is that the final configuration is optimal. However, a large  $r$  requires the testing of a large number of possible exchanges that increases the computing time rapidly. As a result, 2-opt, 3-opt and Lin-Kernighan algorithms are the ones that are most commonly used, in which, respectively, two, three and a variable number of edges are exchanged (Lin, 1965; Lin and Kernighan, 1973). This study chooses the Or-opt algorithm. The Or-opt heuristic (Or, 1976) is a modified 3-opt heuristic which reduces the number of exchanges to be tested by cutting a piece of route and inserting it between two other cities. For each connected string of cities in the current tour, it is tested to see if it can be relocated between two other cities at a reduced length.

When running the Or-opt search algorithm on the TSP instance, this study chooses an initial tour at random and then carries out 10000 iterations. For the local search system in this study, the values of tour length recorded at each iteration yield the sequence of numbers:

$$l_0 = s(t_0), l_1 = s(t_1), \dots, l_i = s(t_i), \dots, l_{10000} = s(t_{10000}) \quad (7)$$



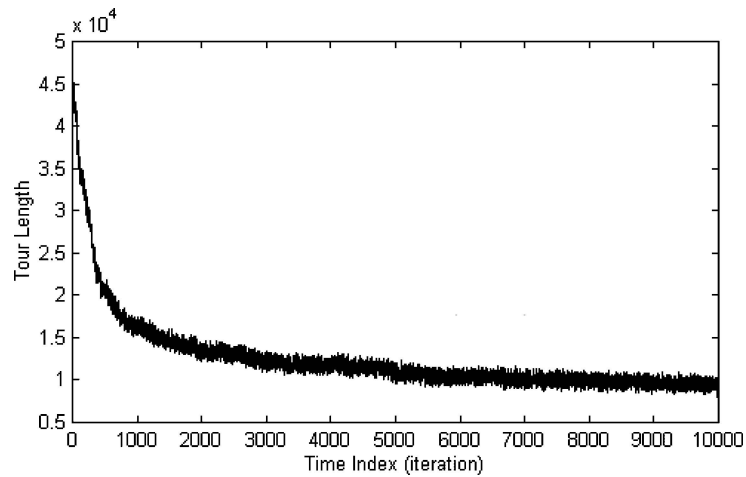


Figure 3. The time series plot for the Or-opt heuristic search system on TSP.

where  $l_0$  is the length of an initial tour and  $l_i$  is the length of the tour at the  $i$ th iteration. If we consider the length value as the state variable and one iteration as a time interval, this sequence forms our experimental time series. For convenience, let  $D$  denote the data set of 10000 points  $l_1, l_2, \dots, l_{10000}$ .

Figure 3 plots the time series  $D$ . In the horizontal axis the number of iteration (time) is marked, while on the vertical axis the state values of solution are given for each iteration. The plot shows an irregular curve with a certain amount of structure. The search pattern of the heuristic search system shows a movement from a descending phase to a plateau phase, indicating that the performance of the local heuristic search slows down as a local optimal point is approached.

Additional knowledge concerning the dynamics of the heuristic search process can lurk in the time series data. What is the dynamical complexity of the heuristic search process? The behavior of a dynamical process is typically described by plotting its trajectories in a state space. The reconstruction of a trajectory can be interpreted as a process of coordinate change. Often the trajectory of a dynamical process is originally defined in some high-dimensional space. The reconstruction amounts to a projection of the original to a smaller dimensional Euclidean space. The goal of trajectory reconstruction is to capture the geometric pattern of points on a trajectory of a dynamical system (Packard et al., 1980). Figure 4 uses the time-delay-coordinates technique to reconstruct the search trajectory in  $d_E = 3$  dimension space, using the same time series as shown in figure 3, through which the geometric structures of the heuristic search trajectory are retrieved (Li, 1998). In figure 4, the plot (a) shows the search trajectory of whole 10000 data points (data  $D$ ), while the plot (b), (c) and (d) present the parts of the search trajectories represented by the first 1000 iterations  $l_1, l_2, \dots, l_{1000}$ , the 4000<sup>th</sup>–5000<sup>th</sup> iterations  $l_{4000}, \dots, l_{5000}$ , and the 8000<sup>th</sup>–9000<sup>th</sup> iterations  $l_{8000}, \dots, l_{9000}$ . The value of the time-delay  $T$  used in the plot (a) is 12, 6 in plot (b), 9 in plot (c) and 14 in plot (d), which are calculated based on average mutual information (Li, 1998).

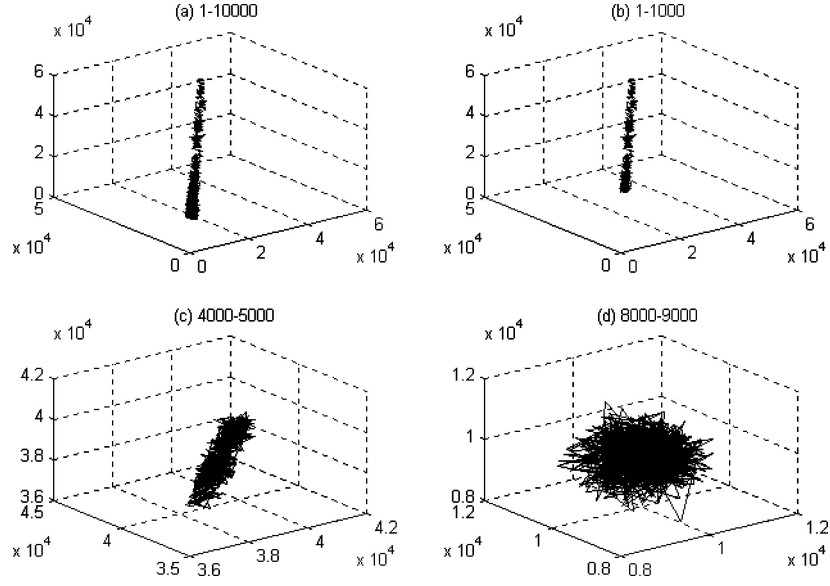


Figure 4. The search trajectory of the Or-opt heuristic search process.

These portraits provide a geometric representation of complex dynamics of the local search process. As shown in plot (b), the search trajectory has a loose structure at beginning of the search, which means the search process has more opportunities to find better solutions and thus has fast convergent speed. The middle part of the trajectory illustrated in plot (c) shows that the structure becomes tight, which indicates that the process has to take longer time searching neighbors to find a better solution. The lower part shown in plot (d) reveals even tighter structure, like a thread ball, which means that the search activity is restricted in an even smaller domain.

Figure 5 plots the ten curves that correspond to the computation of the natural logarithm of the correlation integral as a function of  $\ln r$  in the embedding dimension  $d_E = 1, 2, \dots, 10$  for the data D. The values of  $r$  are experimentally determined in the following manner. Let  $R_A$  represents the size of the trajectory under consideration, which can be obtained from the following equation (Kennel et al., 1992):

$$R_A = \sqrt{\frac{1}{n} \sum_{t=1}^n [s(t) - \bar{s}]^2} \quad (8)$$

where  $n$  is the number of data points and

$$\bar{s} = \frac{1}{n} \sum_{t=1}^n s(t) \quad (9)$$

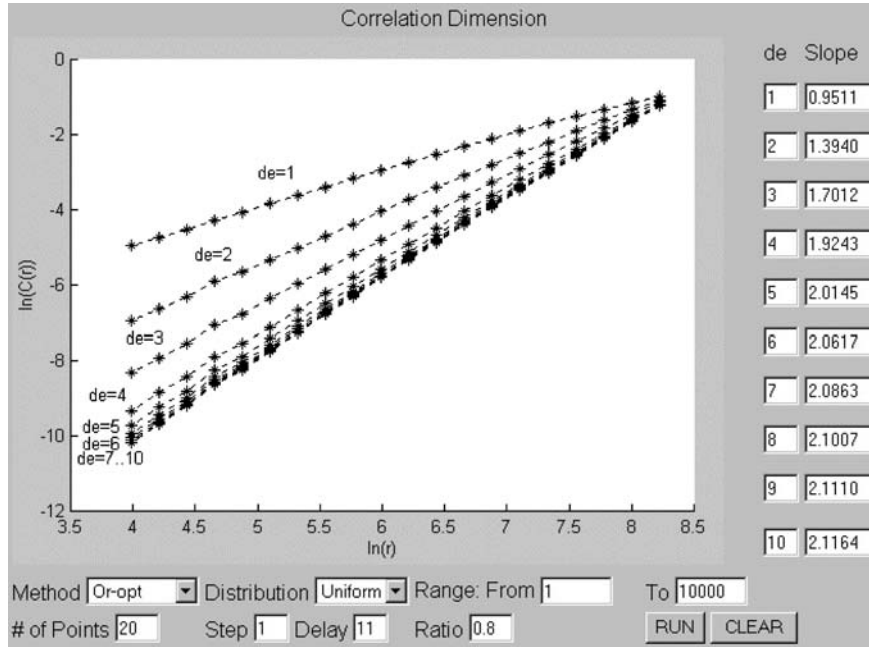


Figure 5. Correlation integers in different embedding dimension  $d_E = 1, 2, \dots, 10$  for the data D.

Then  $r$  is calculate as

$$r = R_A \cdot \alpha^k \tag{10}$$

where the values of  $\alpha$  and  $k$  were chosen experimentally to provide a range of values of  $r$  that would encompass a nearly linear region of the curve. This study selected  $\alpha = 0.8$  by using a trial-and-error program. The portion that lie within the range was calculated for  $r = R_A(0.8)^{0.2}, R_A(0.8)^{0.4}, R_A(0.8)^{0.6}, \dots, R_A(0.8)^4$ , which made 20 points on the curve. In this way, a series of correlation dimensions was calculated over 10 embedding dimensions and plotted. Then the slope for each curve was found by the least squares regression and displayed in the rightmost column in figure 5.

As a practical matter we search to see if the values of correlation dimension stabilize at some value  $D_c$  as embedding dimension  $d_E$  increases. If so, then the value  $D_c$  is the correlation dimension estimate for the process. If a low value for  $D_c$  is found, then the process is substantially deterministic even it is complicated. If, however, as  $d_E$  increases the correlation dimension continues to increase at the same rate, then the process is taken to be stochastic. In principle, an independently and identically distributed stochastic system is infinite dimensional. Each time one increases one available degree of freedom, the random system utilizes that extra freedom. Therefore, it is desirable to calculate the correlation dimension for a “random” system to provide a basis for comparison. Figure 6 compares

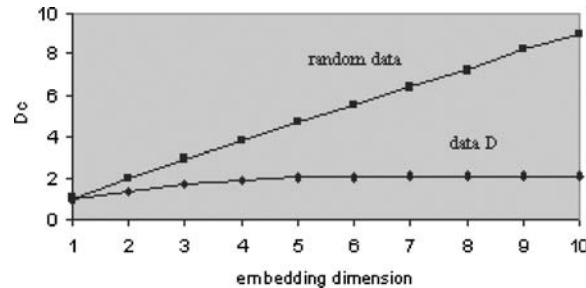


Figure 6. Comparison of correlation dimensions between the heuristic time series D and the random data.

the correlation dimensions of the heuristic time series with the correlation dimensions of a random data, which consists of 10000 data points that are uniformly distributed over the interval  $[1, 1000]$ . As shown in figure 6, the correlation dimension for the time series of the local heuristic search process stabilize when the embedding dimension is greater than about twice the correlation dimension of the data, which indicates that the correlation dimension becomes independent of  $d_E$ . The saturated correlation dimension value  $D_c = 2.0145$  for the data D is determined at the corresponding  $d_E = 5$ . For the random-number data, no saturated correlation dimension can be found.

The correlation dimension estimate for the data D contains information about the dynamics of the local heuristic search process during its first 10000 search steps. In fact, this single value describes an averaged property of the search process. In other words, this correlation dimension is an average measure over the entire search trajectory (10000 points). However, reviewing the time series plot in figure 3 and the search trajectories in figure 4, one can clearly find that, in the heuristic search process, the descending phase has a different geometric structure from the plateau phase's, which may indicate different levels of dynamic complexity. The local contributions from different parts of the search trajectory to the average dynamics are, in general, different. Just an average value does not acknowledge all of the complexity of the search process. We ask what happen to the dynamics of the search trajectory if we examine it at different parts. Thus, this study divides the 10000-point trajectory into 10 parts, each consisting of 1000 points. It is expected that different parts of the search trajectory to be characterized by different values of the correlation dimension. In such a situation, a single value of some fractal dimension is not sufficient to characterize the search trajectory adequately. According to its geometric appearance, the heuristic search trajectory should be an object with a multiplicity of fractal dimension. Figure 7 shows a schematic graph of the correlation dimension as a function of the embedding dimension for five partitions of the heuristic time series. For comparison, the correlation dimension for a 1000-point random data is also plotted.

We can see that, although the curve of correlation dimension for the first 1000 data points are below the curve for the random data, up to  $d_E = 10$ , there is no sign of saturation, indicating that the time series for this partition are indeed of high, if not infinite, dimension. It indicates the presence of a stochastic search process during the first 1000 search steps. However, the correlation dimension of search process decreases as the search continues,

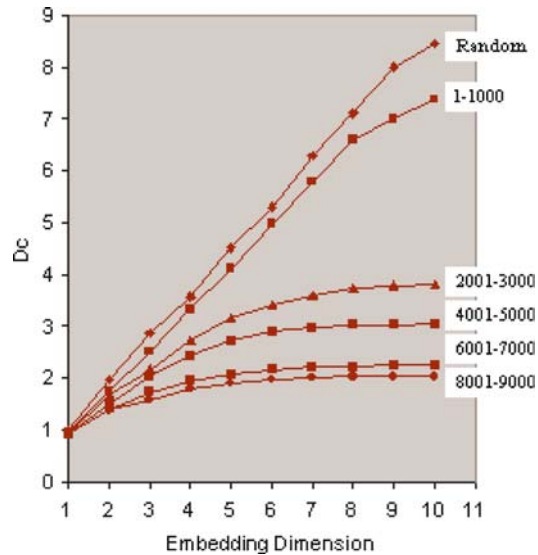


Figure 7. Correlation dimensions for the partitions of the heuristic search data and random data.

indicating that the complexity of search dynamics appears to decrease as the heuristic search is approaching a local optimal point. In summary, in the course of searching for an optimal solution in the configuration space, the dynamics of the heuristic search process changes from a stochastic process to a process with a low-dimensional structure.

#### 4. Conclusions

This study investigates the complex nature of the dynamical behavior of the local heuristic search process. This study uses the correlation dimension technique to quantitatively characterize the dynamical complexity for a particular search trajectory. The results reveal that the heuristic search process exhibits the transition from high-dimensional stochastic to low-dimensional chaotic behavior. The initial phase of search process shows more complexity in dynamics than the later phases of the search. The heuristic search process starts with a high-dimensional searchable space. However, the dimension of the searchable space is rapidly reduced. As the search is approaching to a local optimal point, the searchable space becomes smaller and smaller. This decomplexification of search dynamics appears to be generic response to perturbation associated with the solution configuration. A local heuristic search process searches for an optimal configuration by repeatedly moving from the current configuration to a neighboring configuration. In the early stage of search, the frequency of this configuration perturbation is very high. However, the possibility of finding a better configuration is less and less as the search goes on. Thus, the loss of complexity, as assessed by the measurement of correlation dimension, can be related with this decreasing possibility. The search process looks very much like passing through many levels of

Table 1. Iterations and better solutions.

Iteration	1000-City	2000-City	4000-City
100	62	71	79
1,000	206	359	466
10,000	242	444	659
100,000	250	471	776
1,000,000	250	476	830

laminar. The search begins with a large laminar with many corridors. It is very easy for the process to find a corridor to pass through to next laminar, which is characterized by smaller size and less corridors. Every time the search process gets into a new laminar, it will take a longer time to find a corridor to pass through to the next laminar. The closer to a local optimal point, the longer the average laminar time will take. One issue related with the loss of complexity is the size of solution space that can be effectively searched. Many degrees of freedom  $N$  on the first laminar are reduced to a few degrees of freedom  $n$  ( $n \ll N$ ) on the later laminars. Average over the residual degrees of freedom ( $N - n$ ) is increased by the restriction of neighborhoods. The size of solution space that can be reached may decrease exponentially.

This decreasing size of the solution space perhaps can be modeled to describe the search effectiveness for a heuristic search process. Usually, the more iterations we perform the better final solution we can get. But how many iterations are enough? It is perhaps somewhat surprising that there does not yet exist a useful convergence theory for this question. Let us assume that after, say 175321 iterations, we found another better solution. However, is it the last one we will find if we continue to iterate? There is no way to tell. However, still we must make a decision. For example, we can propose to perform another 10000 iterations, and if we do not find any more new better solution, we may be content and stop the iteration. However, probably only the next iteration beyond the 10000 may prove that we are wrong in this assumption. Table 1 illustrates the results of an experiment. This experiment applies the Or-opt algorithm on three uniformly distributed TSP instances with 1000, 2000, and 4000 cities, respectively. One million iterations are performed on each instance. The number of better solutions found (i.e. the times of solution configuration perturbations) during  $10^k$  ( $k = 2, 3, \dots, 6$ ) iterations is recorded.

We can see that, for example, for the 1000-city instance, 242 better solutions are found in the first 10000 iterations and another eight better solutions in the next 90000 iterations, but no more better solutions found in the next 900000 iterations. In the 2000-city case, 471 better solutions are found in the first 100000 iterations and another five better solutions in the next 900000. How many more iterations should we perform to produce one more better solution? Even if the case of simple local search iterations are very cheap, for a very large problem instance, the number of iterations needed for a satisfied solution may be typically in the order of tens of millions. For many other more sophisticated search heuristics, one million iterations may be well above any reasonable computational time frame. Thus, the problem outlined above really exists in practice, and a solution should be sought. We have

known that this problem may be associated with the loss of complexity in the search process. Correlation dimension technique may help us to capture the relevant frequencies or time scales for a particular heuristic search process on a particular problem. Then, an expected number of better solution ( $B$ ) found during  $n$  iterations, given the size of the problem ( $s$ ), may be predicted by a map  $B = f(n, s)$ . Such a search effectiveness model may answer the question whether we are finding an adequate solution within a time acceptable to the situation, for this size of problem instance, when using this particular search heuristic. Thus, if a heuristic user can expect how many better solutions will be produced during the next, say 100000 iterations, one has great flexibility in deciding whether or not to trade solution quality for time.

Although this study investigates the dynamical behavior of the Or-opt heuristic on the TSP, the result appears to be behavioral archetypes for the class of local search heuristics in combinatorial optimization problems generally. There is no strict mathematical definition to describe the behavior of the heuristic search process concisely. The process exhibits complicated patterns in both space and time. Suppose that, for a heuristic search system, the initial conditions are confined to some volume  $V$  in solution space bounded by a “surface” of all initial points. As the search process evolves, the volume  $V$  will move through solution space, which looks like a movement of a compressible object. The shape of the volume will be distorted, and the size will be decreased. The final volume occupied by the set of local optimal points is very small. This small volume is called a *solution attractor* in the solution space. All search trajectories will be attracted to this volume. In fact, there are two competing effects present in the heuristic search process. In one, due to the sensitive dependence on initial conditions, two different search trajectories will diverge. On the other hand, the trajectories must converge to a finite region of the solution space. This region is fixed. It is clear that the initial segment of search trajectory is a kind of “coherent randomness.” This randomness is subdued by some modulation. After certain search iterations, this randomness is quickly replaced by a low-dimensional mode of chaos. Then, the search process continues to be suppressed slowly by the modulation over a long period of time. Finally, when the system reaches some local optimal configuration, the chaos is replaced by a quasi-periodic mode of motion.

In the classical picture of statistical mechanics, it is believed that a large number of degrees of freedom are necessary for ergodicity (Kuramoto, 1984). When a heuristic search process is in the stochastic movement stage, it has the ability to explore wide regions of the solution space, and thus to utilize information about a large collection of potential solutions of the process. When the process gets into the chaotic stage with a few degrees of freedom, it explores a very limited domain of solution space. The search activity now is rather restrictive. The distribution of solution space being searched is far from “equipartition” and is replete with high-probable regions. In summary, a local search process can start from different initial points in solution space and finally wind up at a small attracting region in the solution space. No wonder Boese et al. (1994) found a “big valley” structure governing the set of local optima. Their research shows the evidence that “very good solutions are located near other good solutions.” In fact, it is the attracting region that creates a globally convex structure for the set of local optima in the optimization solution space.

This type of asymptotic behavior of the heuristic search process can have distinct implications for a wide range of theoretical and practical problems in heuristic theory, design and analysis. For example, this dynamical property of heuristic search can probably be applied to an emerging search methodology: hyper-heuristics. *Hyper-heuristics* are an approach that chooses between heuristics in order to solve a given optimization problem (Burke et al., 2003; Burke, Kendall, and Soubeiga, 2003; Cowling, Kendall, and Soubeig, 2001; Gaw, Rattadilok, and Kwan, 2004). A hyper-heuristic consists of a set of low-level heuristics and a high level selector. The hyper-heuristic manages the choice of which lower-level heuristic method should be applied at any given time, depending upon the characteristics of the heuristics and the region of the solution space currently under exploration. In some hyper-heuristic approaches, a *choice function* is used to decide which heuristic to call next (Cowling, Kendall, and Soubeig, 2001). This is calculated based on information derived from recently called low-level heuristics, such as the recent improvement of each individual heuristic, the recent improvement of each pair of heuristics, and the CPU time used by each heuristic. The dynamics of a search trajectory can be used in the choice function.

## References

- Abarbanel, H., D.I., R. Brow, J.J. Sidorowich, and L. Sh. Tsimring. (1993). "The Analysis of Observed Chaotic Data in Physical Systems." *Reviews of Modern Physics* 65(2), 1331–1392.
- Alligood, K.T., T.D. Sauer, and J.A. Yorke. (1997). *Chaos: An Introduction to Dynamical System*. New York: Springer-Verlag.
- Boese, K.D., A.B. Kahng, and S. Muddu. (1994). "A New Adaptive Multistart Technique for Combinatorial Global Optimizations." *Operations Research Letters* 16(2), 101–113.
- Bonomi, E. and J.L. Lutton. (1984). "The N-city Traveling Salesman Problem: Statistical Mechanics and the Metropolis Algorithm." *SIAM Review* 26, 551–568.
- Brock, W. (1986). "Distinguishing Random and Deterministic Systems: Abridged Version." *Journal of Economic Theory* 40(1), 168–195.
- Burke, E.K., G. Kendall, J. Newall, E. Hart, P. Ross, and S. Schulenburg. (2003). "Hyper-Heuristics: An Emerging Direction in Modern Search Technology." In Glover, F.W. and G.A. Kochenberger (eds.), *Handbook of Metaheuristics*, Kluwer Academic Publishers, pp. 457–474.
- Burke, E.K., G. Kendall, and E. Soubeiga. (2003). "A Tabu-Search Hyperheuristic for Timetabling and Rostering." *Journal of Heuristics* 9(6), 451–470.
- Buzug, T., T. Reimers, and G. Pfister. (1990). "Optimal Reconstruction of Strange Attractors from Purely Geometrical Arguments." *Europhysics Letters* 13(7), 605–610.
- Casdagli, M., T. Sauer, and J.A. Yorke. (1991). "Embedology." *Journal of Statistical Physics* 65(3/4), 579–616.
- Cowling, P.I., G. Kendall, and E. Soubeig. (2001). "A Parameter-Free Hyperheuristic for Scheduling a Sales Summit." *Proceedings of the Third Metaheuristic International Conference (MIC2001)*, 127–131.
- Ding, M., C. Grebogi, E. Ott, T. Sauer, and J.A. Yorke. (1993). "Plateau Onset for Correlation Dimension: When Does It Occur?" *Physical Review Letter* 70(25), 3972–3975.
- Evans, J.R. (1987). "Structural Analysis of Local Search Heuristics in Combinatorial Optimization." *Computer and Operations Research* 14(3), 465–477.
- Fontana, W., P.F. Stadler, E.G. Bornberg-Bauer, T. Griesmacher, I.L. Hofacker, M. Tacker, P. Tarazona, E.D. Weinberger, and P. Schuster. (1993). "RNA Folding and Combinatory Landscapes." *Physical Review E* 47(3), 2083–2099.
- Frazer, A.M. and H.L. Swinney. (1986). "Independent Coordinates for Strange Attractors from Mutual Information." *Physical Review A* 33(2), 1134–1140.
- Garey, M.R. and D.S. Johnson. (1979). *Computers and Intractability*. New York: Freeman.



- Gaw, A., P. Rattadilok, and K. Kwan. (2004). "Distributed Choice Function Hyper-Heuristics for Timetabling and Scheduling." In *Proceedings of the 2004 International Conference on the Practice and Theory of Automated Timetabling (PATAT 2004)*, pp. 495–497.
- Grassberger, P. and I. Procaccia. (1983). "Characterization of Strange Attractor." *Physical Review Letters* 50(5), 346–349.
- Grover, L.K. (1992). "Local Search and the Local Structure of NP-Complete Problem." *Operations Research Letters* 12(4), 235–243.
- Hilborn, R.C. (1994). *Chaos and Nonlinear Dynamics: An Introduction for Science and Engineers*. New York: Oxford University Press.
- Holzfuss, J. and G. Mayer-Kress. (1986). "An Approach to Error-Estimation in the Application of Dimension Algorithm." In G. Mayer-Kress (ed.), *Dimensions and Entropies in Chaotic Systems: Quantification of Complex Behavior*. Berlin: Springer-Verlag, pp. 114–122.
- Johnson, D.S. (1990). "Local Optimization and the Traveling Salesman Problem." *Proceedings of the 17th Colloquium on Automate Languages and Programming, Lecture Notes in Computer Science* 443, Berlin: Springer, 446–461.
- Jones, T. and S. Forrest. (1995). "Fitness Distance Correlation as a Measure of Problem Difficulty for Generic Algorithms." In *Proceedings of the Sixth International Conference on Genetic Algorithms*, Morgan Kaufmann, 184–192.
- Kauffman, S.A. (1993). *The Origins of Order: Self-Organization and Selection in Evolution*, Oxford University Press.
- Kauffman, S.A. and S. Levin. (1987). "Towards a General Theory of Adaptive Walks on Rugged Landscapes." *Journal of Theoretical Biology* 128, 11–45.
- Kennel, M.B., R. Brown, and H.D.I. Abarbanel. (1992). "Determining Embedding Dimension for Phase-Space Reconstruction Using a Geometrical Construction." *Physical Review A* 45(6), 3403–3411.
- Kuramoto, Y. (1984). *Chaos and Statistical Methods*. New York: Springer-Verlag.
- Li, W. (1998). "Dynamics of Heuristic Search System for Combinatorial Optimization Problems: the Traveling Salesman Problem Case." Ph. D. Dissertation, University of Mississippi, Oxford MS.
- Lin, S. (1965). "Computer Solutions of the Traveling Salesman Problem," *The 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.
- Macken, C.A., P.S. Hagan, and A.S. Perelson. (1991). "Evolutionary Walks of Rugged Landscapes." *SIAM Journal on Applied Mathematics* 51(3), 799–827.
- Merz, P. and B. Freisleben. (1999). "Fitness Landscapes and Memetic Algorithm Design." In D. Corne, M. Dorigo and F. Glover (eds.) *New Ideas in Optimization*. London: McGraw-Hill, pp. 245–260.
- Or, I. (1976). "Traveling Salesman-Type Combinatorial Problems and Their Relation to the Logistics of Regional Blood Banking." Ph. D. Thesis, Northwestern University, Evanston IL.
- Ott, E., T. Sauer, and J. A. Yorke (eds.), (1994). *Coping with Chaos: Analysis of Chaotic Data and the Exploitation of Chaotic Systems*. New York: John Wiley & Sons.
- Packard, N., J.P. Crutchfield, J.D. Farmer, and R.S. Shaw. (1980). "Geometry from a Time Series." *Physical Review Letters* 45(9), 712–716.
- Papadimitriou, C.H. (1994). *Computational Complexity*. Reading, MA: Addison-Wesley.
- Papadimitriou, C.H. and K. Steiglitz. (1999). *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications.
- Parker, T.S., and L.O. Chua. (1990). *Practical Numerical Algorithms for Chaotic Systems*. New York: Springer-Verlag.
- Ramsey, J.B. and H.J. Yuan. (1987). "The Statistical Properties of Dimension Calculations Using Small Data Sets." Research Report #87-20, C. V. Starr Center, New York University.
- Reeves, C.R. (1993). *Modern Heuristic Techniques for Combinatorial Problem*. New York: John Wiley & Sons, Inc.
- Reidys, C.M. and P.F. Stadler. (2002). "Combinatorial Landscapes." *SIAM Review* 44, 3–54.
- Shaw, R.S. (1985). *The Dripping Faucet as a Model Chaotic System*. Santa Cruz, CA: Aerial.
- Sorkin, G.B. (1991). "Efficient Simulated Annealing on Fractal Energy Landscapes." *Algorithmica* 6, 367–418.

- Takens, F. (1981). "Detecting Strange Attractors in Turbulence." In D. Rand and L. S. Young (eds.), *Dynamical Systems and Turbulence, Warwick 1980, Lecture Notes in Mathematics*, 898. New York: Springer-Verlag, pp. 366-382.
- Taken, F. (1985). "On the Numerical Determination of the Dimension of an Attractor." In N. Breaksma, H. Broer, and F. Takens (eds.), *Dynamical Systems and Bifurcations*. Berlin: Springer-Verlag, pp. 99-106.
- Tong, H. and K.S. Lim. (1980). "Threshold, Autoregression, Limit Cycles and Cyclical Data." *Journal of the Royal Statistical Society* 42(3), 245-292.
- Weinberger, E.D. (1990). "Correlated and Uncorrelated Fitness Landscapes and How to Tell the Difference." *Biological Cybernetics* 63, 325-336.
- Weinberger, E.D. (1991). "Local Properties of Kauggman's  $N-k$  Model: A Tunably Rugged Energy Landscape." *Physical Review A* 44(10), 6399-6413.