Analysis on the Collaboration between Global Search and Local Search in Memetic Computation

Jih-Yiing Lin
Ying-ping Chen

NCLab Report No. NCL-TR-2010005
April 2010

Natural Computing Laboratory (NCLab)
Department of Computer Science
National Chiao Tung University
329 Engineering Building C
1001 Ta Hsueh Road
HsinChu City 300, TAIWAN
`http://nclab.tw/`

Analysis on the Collaboration between Global Search and Local Search in Memetic Computation

Jih-Yiing Lin and Ying-ping Chen
Department of Computer Science
National Chiao Tung University
HsinChu City 300, Taiwan
{jylin, ypchen}@nclab.tw

April 30, 2010

Abstract

The synergy between exploration and exploitation has been a prominent issue in optimization. The rise of memetic algorithms, a category of optimization techniques which feature the explicit exploration-exploitation coordination, much accentuates this issue. While memetic algorithms have achieved remarkable success in a wide range of real-world applications, the key to successful exploration-exploitation synergies still remains obscure as conclusions drawn from empirical results or theoretical derivations are usually quite algorithm specific and/or problem dependent. This paper aims to provide a theoretical model that can depict the collaboration between global search and local search in memetic compuatation on a broad class of objective functions. In the proposed model, the interaction between global search and local search creates a set of local search zones, in which the global optimal points reside, within the search space. Based on such a concept, the Quasi-Basin Class (QBC) which categorizes problems according to the distribution of their local search zones is adopted. For the behavior of local search, the subthreshold seeker, taken as a representative archetype of memetic algorithms, is analyzed on various QBCs to develop a general model for memetic algorithms. As the proposed model not only well describes the expected time for a simple memetic algorithm to find the optimal point on different QBCs but also consists with the observations made in previous studies in the literature, the proposed model may reveal important insights to the design of memetic algorithms in general.

1 Introduction

Memetic algorithms (MAs), a category of optimization techniques which coordinate population-based global search and local search, are widely used nowadays. With an appropriate coordination, memetic algorithms cannot only exhibit a good explorative ability as a population-based global search algorithm does but also deliver a good exploitive performance as a local search algorithm does. As a result, memetic algorithms perform better than pure population-based global search algorithms or standalone local search algorithms. As the design of adopting local searchers or memes into population-based global search algorithms thrives, a variety of memetic algorithms have been proposed to solve NP problems, real-parameter problems, multi-objective problems, and manifolds of real-world problems. Among these memetic algorithms, in addition to the selection of the global search component and the local search operator, the synergy between global search and local search has been one of the key design issues.

The design of most memetic algorithms follows the seminal studies on memetic algorithms proposed in [1, 2]. In these studies, the authors observed that memetic algorithms favor infrequent starts and long running time of local search. They also proposed several renowned strategies for selecting solution candidates on which the local search operator is applied: the fitness based selection, the diversity based selection, and the local search potential based selection. These design guidelines are widely adopted in numerous studies [3, 4, 5, 6, 7, 8, 9]. However, with the aids of these guidelines, designing a memetic algorithm for a specific problem still requires considerable time as the optimal design is not only algorithm specific but also problem dependent. To cope with this issue, the concept of systematically adjusting the parameters of local search proposed [10]. Although this technique is robust, it does not guarantee the best performance. Another line of research is regarding the concept of memes [11, 12, 13, 14]. In these studies, the local search algorithms, encoded as memes, can adapt to the underlying problem and thus improve the efficiency as the memetic algorithm progresses. This framework is robust as well as efficient with the expense of the learning cost of memes.

In spite of the light shed on the design issue of memetic algorithms by the aforementioned studies, the question of how one can achieve the optimal design of memetic algorithms on a specific problem remains. The key to achieve this ultimate goal apparently include a full awareness of the physics behind the algorithm and the problem. As theoretical studies can help to understand the internal mechanism of algorithms, they can provide important insights to the design issue. Due to the complexity of memetic algorithms, most theoretical models are developed based on the behavior of a much simplified algorithm when the algorithm is applied to handle several classes of functions. In the continuous domain, the theoretical models are usually developed based on the convergence theory [15, 16, 17], while in the discrete domain, the (1+1)-evolutionary algorithm (EA) has been widely adopted in theoretical analysis as the archetype of evolutionary algorithms [18, 19, 20, 21, 22, 23]. The theoretical results on (1+1)-EA were further extended to that on (1+1)-MA and (μ+λ)-MA to describe the behavior of memetic algorithms on several specific function classes [24, 25]. As these theoretical models are developed based on different classes of functions, they depict different algorithm behaviors from different perspectives and therefore cannot provide a unified principle to the design of memetic algorithms.

The concept of basins of attraction provides another perspective [26] and gives an opportunity to conduct more general analyses of memetic algorithms. This concept views the landscape of the underlying problem as a union of basins of attraction. Thus, problems can be categorized by the basin distribution on the landscape. A similar concept, a quasi-basin defined by subthershold seeker, was adopted to prove the searchability of general functions [27] and to investigate the subthreshold seeking behavior [28]. In the present work, we aim to establish a theoretical model for the algorithm-problem complex in order to depict the behavior of memetic algorithms on a wide range of problems. To achieve this, we adopt the concept of quasi-basins and consider quasi-basins as the local search zones which the local search algorithm exploits. In this perspective, these local search zones are defined by the landscape of the problem and the collaboration between global search and local search. Hence, we define the Quasi-Basin Class (QBC) which categorizes problems by their quasi-basin distributions as the basis on which memetic algorithms are investigated. Then, we analyze the performance of the subthreshold seeker, which is regarded as a representative archetype of memetic algorithms, to develop a general theoretical model for the global-local search collaboration in memetic computation. The derived theoretical model can well describe how the distribution of local search zones and the efficiency of the local search algorithm are related to the expected time for the memetic algorithm to find the optimal solution. Because this model, empirically verified, consists with the observations made in many previous studies in the literature, it is valid for various kinds of memetic algorithms on a broad range of problems and may give some important insights to the design of advanced memetic algorithms.

The rest of this paper is arranged in the following manner. Section 2 gives a survey on the current progress of analysis on memetic algorithms and elaborates the need of a general theoretical model which can describe the behavior of memetic algorithms on a broad range of problems. Section 3 expounds the fundamental concepts on the analysis of memetic algorithms and provides the definitions of Quasi-Basin Classes and the subthreshold seeker to form the basis for further derivation. As a memetic algorithm comprises global search and local search, we firstly analyze the global search component of the subthreshold seeker and discuss how this analysis is related to the behavior of common global search algorithms in section 4. Based on the analysis of global search and the concept of QBC, we derive and empirically verify the formula that describes the behavior of the subthreshold seeker working with local search operators of different efficiency on various QBCs in section 5. A thorough discussion on how the established model is related to other previous studies and what the model implies for memetic algorithms is presented in section 5. Finally, we recap the significance of our model and give the potential future work in section 6.

2 Background

Designing a memetic algorithm requires not only selecting a global search mechanism as well as a local search operators but also establishing a subtle coordination to exhibit the vantage of both ends. Hart [1] in his seminal study for designing efficient memetic algorithms investigated the following four questions on continuous optimization problems:

1. How often should local search be applied?
2. On which solutions should local search be used?
3. How long should local search be run?
4. How efficient does local search need to be?

In his framework, he noted that the memetic algorithms that employ elitism will be most efficient with large population sizes and infrequent local search. He also proposed two strategies, fitness based selection and diversity based selection, for selecting solution candidates to apply local search. He concluded that these two strategies help much. Land [2] extended Hart's study to combinatorial domains. In his study, he adopted steady state genetic algorithms as global search and proposed a local search potential based strategy in selecting local search candidates. The local search potential strategy turned out to be not very useful. Yet, he observed that his steady state memetic algorithm favored smaller rates and longer runn time for local search, consistent with Hart's study.

Although limited to specific problems, the studies of Hart and Land gave some insights to the first three questions and have inspired the successive memetic algorithms in a wide variety of applications. The concepts of selecting the best or some qualified individuals for local search which resemble the fitness based selection have been adopted in [3, 4, 5]. The steady state memetic algorithm with adaptive local search probability has been applied in [6, 7, 29], while other studies exhibit the vantage of utilizing the diversity information in their design of memetic algorithms [30, 31]. Investigations into the balance between global search and local search for some applications available in the literature also accord with Hart's and Land's observations [8, 9]. Despite that the accordance of these results reveals some essential design principles of efficient memetic algorithms, designing a memetic algorithm still requires a considerable amount of effort due to the lack of detailed knowledge on how the key mechanism of memetic algorithms, the synergy between global search and local search, working on the underlying problem. An interesting technique of adapting local search intensity in a simulated annealing way was proposed to cope with the MA parameterizing issue [10]. More robust than the fixed local search intensity setting, this method still requires a range setting and does not guarantee the best performance.

In addition to the parameterizing issue caused by using memetic algorithms to handle different problems, the efficiency of a local search operator is particularly problem dependent. [32] suggested that local search operators of memetic algorithms should be chosen according to the analysis of local structures. Following this guideline, the concept of memes [11, 12, 13] was proposed. In these frameworks, the local search component is designed to adapt to the underlying problem as the optimization progresses. These memetic algorithms are robust regardless of the underlying problem with the expense of meme evolving or learning cost.

Besides empirical investigations, theoretical analysis has always been a prevalent way in providing clues to the design of algorithms. Because the algorithm to analyze is usually complicated, theoretical models are often developed on a simplified prototype working on some narrow problem classes. For continuous optimization methods, convergence theory is widely adopted in performance assessment [15, 16, 17]. To use the convergence theory, the algorithm under analysis requires simplification as well as the underlying problem requires definite specification. Consequently, the convergence theory based algorithmic models give insights to the optimization behavior via delineating the behavior of an archetype of the algorithm in an impractical world. For discrete optimization methods, the (1+1) evolutionary algorithm (EA) has been widely adopted in theoretical analysis of evolutionary algorithms [18, 19, 20, 21, 22, 23]. The (1+1)-EA is rather simple with one individual and an evolutionary operator flipping each bit of the individual with an uniform probability. These studies analyze the effect of evolutionary operators and complexity of the (1+1)-EA on several specific discrete problems. Extending these studies, the theoretical analysis of memetic algorithms starts from the (1+1)-MA and goes to the (μ+λ)-MA [24, 25]. On three discrete functions, Sudholt investigated the behavior of the (1+1)-MA and the (μ+λ)-MA. His study reaffirmed that the parameterizing of memetic algorithms is extremely hard.

Theoretical models developed in this way are specific to narrow classes of problems. These models may draw different conclusions because of different problem classes. [33] illustrated that different problems favor different population sizes. [34] and [35] which investigated the effect of recombination operators suggested counter perspectives. The issue of the aforementioned analysis technique is that the derived theoretical behavior is confined by those adopted objective functions. As an undesirable result, the different conclusions obtained from various theoretical models cannot form a unified guideline to the design of memetic algorithms.

Another line of analysis involves the concept of basins of attraction. The basin of attraction of a local optimum is the set of points in the search space such that a local search starts from any member within a basin will eventually find the local optimum in that basin [36]. In this line of research, problems are categorized by the distribution of basins of attraction. In [26], a search space is considered to be partitioned by basins of attraction. This work investigates the relationship between the optimal local search run time and the basin traits including the distribution of basins and the corresponding time to find the local optimum. Thus, via acquiring the information of the basin traits, one can assign the optimal local search run time to make memetic algorithms efficient.

A similar concept was introduced by [27] in investigating searchable functions in which the No Free Lunch theorem does not hold. The submedian seeker which starts local search when hitting a point with a submedian value and turns to do random search when hitting a point with a surpermedian value. By applying the submedian seeker to functions with a certain degree of self-similarity, that the functions exhibiting self-similarity are searchable was proved. Whiteley and Rowe further proposed the subthreshold seeker, a generalize one-dimensional submedian seeker, and investigated the subthreshold seeking behavior [28]. In their work, the subthreshold seeking behavior was used as a performance index. Their theoretical analysis detailed the conditions under which the subthreshold seeker could outperform random search and showed that a higher bit-precision could improve the performance.

Finally, in this paper, we aim to provide a general model for the collaboration between global search and local search in memetic algorithms on a broad class of problems. The proposed model will describe how the expected performance of a memetic algorithm is related to the efficiency of the local search operator, the landscape of the problem, and the collaboration between global search and local search. In order to achieve this goal, we adopt the concept of quasi-basin and consider quasi-basins are local search zones defined by the collaboration of global search and local search. These local search zones are the region which the local search operator exploits and the global optimal point resides in. In this perspective, the Quasi-Basin Class (QBC) is employed to formulate the situations faced by search algorithms. The subthreshold seeker, taken as a representative archetype of memetic algorithms, is analyzed over different QBCs as a general theoretical model for memetic algorithms. Thus, the proposed model can depict the essence of the collaboration between global search and local search in memetic algorithms on various problems and may shed light on the design of memetic algorithms.

3 Quasi-Basin Classes and Subthreshold Seeker

In our perspective, applying a memetic algorithm to handle a problem forms a set of local search zones, defined by the landscape of the problem as well as the collaboration between global search and local search. The size ratio between local search zones and the whole search space, the number of local search zones, and the efficiency of the local search operator will much influence the performance of a memetic algorithm. Basically, the size ratio between local search zones and the search space determines the global search time to find a point in a local search zone and the total local search time to find the global optimal point in these zones. The number of zones and the efficiency of local search operators dictate the balance between global search and local search. Based on this way of thinking, we adopt the concept of quasi-basin and define the Quasi-Basin Classs (QBC) to represent different problem classes by their local search zone distributions. Then, we take the subthreshold seeker as a representative archetype of memetic algorithms and analyse its behavior on various QBCs to develop a general theoretical model for the core operation of memetic algorithms.

In this section, we introduce the QBC and the generalized subthreshold seeker on which the theoretical analysis is based. The QBC conceptually defines classes of problems according to the number and the partition of their local search zones, which can be referred to as the partition of subthreshold points when a subthreshold seeker is applied. The subthreshold seeker globally searches the space until it encounters a point with a subthreshold value. Once such an event occurs, it exploits the quasi-basin by visiting the neighbors of the current point. In Whitley and Rowe's work, their subthreshold seeker was applicable only to one-dimensional functions. In our present work, we generalize their subthreshold seeker to more dimensions by utilizing the graph representation in the definition of QBCs for more general applications.

3.1 Quasi-Basin Classes

The task to handle an optimization problem is to optimize a given objective function $f : \mathcal{X} \to \mathcal{Y}$. For convenience, we specify our optimization goal as to find a point $x^* \in \mathcal{X}$ with the minimum value $y^* \in \mathcal{Y}$. We assume that both $\mathcal{X}$ and $\mathcal{Y}$ are finite sets. Such an assumption makes practical sense because optimization problems are generally numerically solved on digital computers. For generality, we interpret the domain $\mathcal{X}$ as the vertex set of a graph $V(G)$ of a graph G to represent its spatial structure. Thus $deg(v)$ and $N(v)$ are used to denote the degree and the neighborhood of a vertex v, respectively. In the rest of this paper, the term $\mathcal{X}$ and $V(G)$ are used exchangeably.

The following gives the essential definitions for the Quasi-Basin Class:

Definition 1 *(Fundamental Definition)*

1. *For any function f, the function value of the m-th largest point is defined as $\beta_m(f) := \arg_y (| \{x \in \mathcal{X} \mid f(x) \leq y\} | = m)$*
2. *For any function f, the set that contains all points with their objective value less than $\beta_m(f)$ is defined as $S_m(f) := \{x \in \mathcal{X} | f(x) \leq \beta_m(f)\}$*
3. *For any function f, a quasi-basin QB is defined as a maximal subset in $S_m(f)$ such that there exists a path between all pairs of vertices.*

As generally the points reside in the local search zones are better than the other points in the search space, the $S_m(f)$ conceptually defines the set of points reside in the local search zones with a size m while a quasi-basin QB can represent a local search zone in the search space. Based on the fundamental definitions, we define the Quasi-Basin Class (QBC) and the uniform Quasi-Basin Class (uQBC) in Definition 2 and Definition 3:

Definition 2 *(Quasi-Basin Class, QBC). Given a connected graph G and $\mathcal{Y}$, the corresponding discrete quasi-basin problems with b distinct quasi-basins and m subthreshold vertices is defined as*

$$\begin{aligned}&\mathcal{Q}(G, \mathcal{Y}, m, b) := \\ &\{f : V(G) \to \mathcal{Y} \mid S_m(f) = \bigcup_{i=1}^{b} QB_i, \bigcap_{i=1}^{b} QB_i = \emptyset, \\ &|QB_i| \geq 1, 1 \leq i \leq b\}\end{aligned}$$

Definition 3 *(uniform Quasi-Basin Class, uQBC). Given a connected graph G and $\mathcal{Y}$, the corresponding uniform discrete quasi-basin problems with b distinct quasi-basins and m subthreshold vertices is defined as*

$$\begin{aligned}&\mathcal{Q}_u(G, \mathcal{Y}, m, b) := \\ &\{f : V(G) \to \mathcal{Y} \mid S_m(f) = \bigcup_{i=1}^{b} QB_i, \bigcap_{i=1}^{b} QB_i = \emptyset, \\ &\left\lfloor \frac{m}{b} \right\rfloor \leq |QB_i| \leq \left\lceil \frac{m}{b} \right\rceil, 1 \leq i \leq b\}\end{aligned}$$

The QBC defines a class of problems with their m smallest points distributed among b distinct quasi-basins. In other words, the problems with their total size of local search zones m and number of local search zones b will be in the same QBC. The uniform QBC further restricts the sizes of quasi-basins should be uniform. Note that m is naturally restricted to be less than or equal to $|\mathcal{X}|$ and greater than or equal to b, and b is a positive integer that is less than the minimum of m and $|\mathcal{X}| - m$.

3.2 Subthreshold Seeker

Global search and local search of the subthreshold seeker are coordinated by the threshold. A subthreshold seeker globally searches by sampling the space uniformly at random (u.a.r.) until it encounters a subthreshold point. Once this occurs, the subthreshold seeker starts local search to exploit the quasi-basin in which the subthreshold point resides. After local search in the quasi-basin is done, the subthreshold seeker continues global search until another subthreshold pointer

procedure SUBTHRESHOLD-SEEKER($\mathcal{X}$, $\mathcal{Y}$, $N : \mathcal{X} \to 2^{\mathcal{X}}$, $f : \mathcal{X} \to \mathcal{Y}$, m)
    while the stopping criterion is not satisfied **do**
        if Queue is not empty **then**
            $x \leftarrow$ Queue.pop();
        else
            Select x from $\mathcal{X}$ u.a.r.
        end if
        if $f(x) \leq \beta_m(f)$ **then**
            Queue.push($N_s(x)$)
        end if
    end while
end procedure

Figure 1: Generalized Subthreshold Seeker.

is encountered. Fig. 1 illustrates how the subthreshold seeker proceeds. The local search part keeps visiting the neighbors of the current vertex until it could no longer walk on a subthreshold vertex.

The notation $N_s(x)$ denotes the set of neighbors of vertex x defined by the local search step size s. When $s = 1$, the neighbors of vertex x seen by local search are exactly the same as its neighbors defined by G. In other words, when $N_1(x)$, all the points in the quasi-basin under local search will be eventually visited. In the rest of this paper, we refer the exhaustive local search as the local search with a step size one. When $s > 1$, the neighbors of vertex x seen by local search are those vertices who are s distance away from x. Here s distance refers to the length of a path consisting s edges on the graph. With the local search step size s, only $1/s$ of the points in the quasi-basin will be visited in one local search run. Note that the distinction between global search and local search is that local search always samples all the neighbors, defined by the local search step size, of the current point, while global search samples the whole search space regardless of the spatial relevance. Note also that this subthreshold seeker does not sample visited points to avoid the performance declination caused by repeated sampling.

4 Stochastic Global Search Time

Before we start to analyze the collaboration between global search and local search in the subthreshold seeker, we firstly investigate the behavior of the global search part employed by the subthreshold seeker. In this section, we theoretically and empirically analyze the behavior of the global search part with respect to the number of subthreshold points m and the number of quasi-basins b. We will derive the expected number of visited points required by the random search, the global search part, to find the first subthreshold point. The expected number of visited points is referred to as the expected first global search time $E(T_\theta)$, where θ is $\beta_m(f)$. Then, we will further approximate the expected k-th global search time. The theoretical results will be empirically verified, and discussions on the implication of the model will be presented.

4.1 The First Global Search Time

In this section, we estimate the expected number of visited points before global search find the first point x with $f(x) <= \theta$, referred to as the first global search time T_θ. The first global search time can be interpreted in the following manner: Since θ=$\beta_m(f)$, the number of points with their function values less than or equal to θ is m. Let N be the size of $\mathcal{X}$. As global search is uniform random sampling without repetition, the search space is of size N and contains m

desired points, the probability for q visited points to contain exactly one subthreshold point follows the hypergeometric distribution with parameters N, m, and q:

$$P(X = 1; N, m, q) = \frac{\binom{m}{1}\binom{N-m}{q-1}}{\binom{N}{q}} .$$

The probability to hit a subthreshold point at the q-th visited points is therefore

$$\frac{1}{q} P(X = 1; N, m, q) .$$

Let $E(T_\theta)$ be the expected first global search time. We have

$$\begin{aligned} E(T_\theta) &= \sum_{i=1}^{N-m+1} i \cdot \frac{1}{i} P(X = 1; N, m, i) \\ &= \sum_{i=1}^{N-m+1} \frac{\binom{m}{1}\binom{N-m}{i-1}}{\binom{N}{i}} \\ &= m \cdot \sum_{i=1}^{N-m+1} \frac{i}{N} \prod_{j=0}^{i-2} \frac{N-m-j}{N-1-j} . \end{aligned} \tag{1}$$

Fig. 2 illustrates the expected value of T_θ with respect to m when $N = 100$. We compare Equation (1) (Theo1) with N/m (Theo2) and the average first global search time in 1000 independent simulation runs (Exp). The N/m is the expected T_θ for allowing sampling visited points which is obviously an upper bound of Equation (1). In the figure, we can find that Equation (1) consists with the empirical result perfectly, while the trend of N/m gradually converges toward the other two. For non-repeated random sampling, half of points in the search space are expected to be visited before finding the minimum point. As m increases, indicating that $S_m(f)$ contains more points, time to meet a point in $S_m(f)$ decreases rapidly regardless of whether or not sampling visited points is allowed. It indicates that although finding several specific points in a search space via random search will cost considerable time, finding a point in a small but large enough set can be attained within a relatively shorter time.

Fig. 3 illustrates the differences among the actual T_θ averaged over 1000 runs, and the two theoretical expected T_θ in ratio. The circle represents the difference between the empirical result and N/m in ratio with respect to the empirical result, and the cross represents that between Equation (1) and N/m. From this figure we can find that Equation (1) can be approximated by N/m as it only deviates significantly from Equation (1) when m is rather small. As Equation (1) is a complicated formula and difficult to analyze, we approximate the expected T_θ with N/m.

4.2 The k-th Global Search Time

In this section, we further measure the expected time for global search to find a subthreshold point after $k-1$ runs of local search have been executed. In other words, we estimate the time for the k-th global search. As the local search frequency and the global search frequency are related to the landscape of the problem, for simplicity, we derive the model on the uniform Quasi-Basin Class $\mathcal{Q}_u(G, \mathcal{Y}, m, b)$. All the problems in this class have their quasi-basin sizes fixed to $\left\lfloor \frac{m}{b} \right\rfloor$ or $\left\lceil \frac{m}{b} \right\rceil$. Because each local search run in a quasi-basin will eventually visit about $1/s$ of the points in the quasi-basin, each local search run will visit $\left\lfloor \frac{m}{bs} \right\rfloor$ or $\left\lceil \frac{m}{bs} \right\rceil$ points. For derivation convenience, we adopt $\frac{m}{bs}$ instead of $\left\lfloor \frac{m}{bs} \right\rfloor$ or $\left\lceil \frac{m}{bs} \right\rceil$ for the number of points visited by

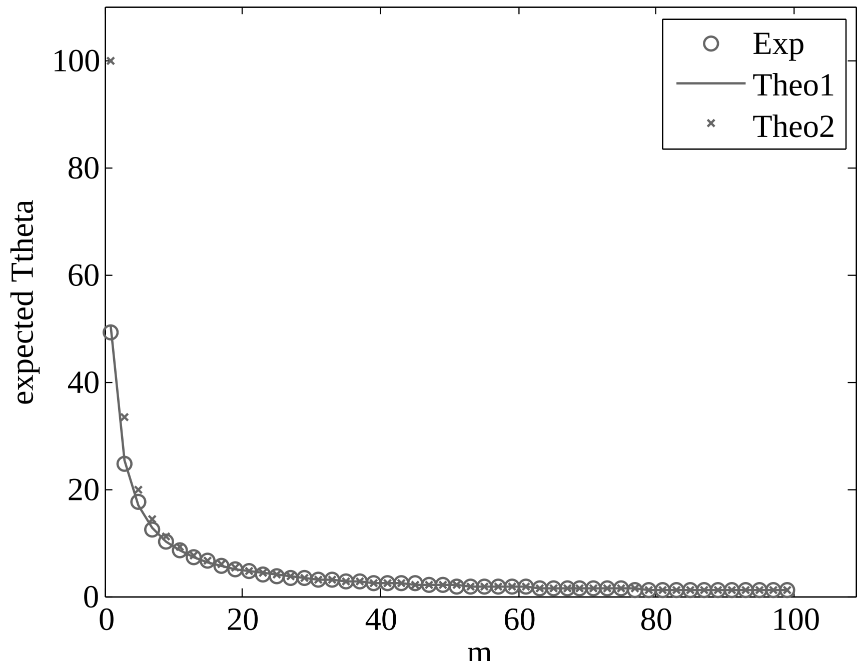


Figure 2: The expected T_θ with respect to m when $N = 100$. Exp represents the actual average T_θ over 1000 independent simulation runs. Theo1 represents the theoretical expected T_θ of non-repeated uniform random sampling, and Theo2 represents the theoretical expected T_θ of uniform random sampling that allows to sample visited points.

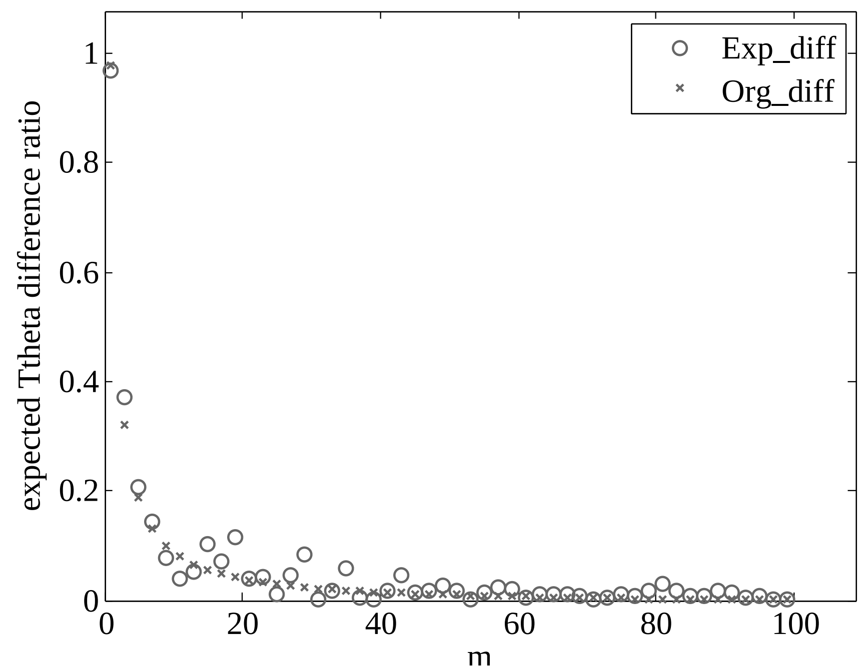


Figure 3: The difference ratio of the expected T_θ with respect to m when $N = 100$. The Exp_diff represents the difference ratio between the actual average T_θ and N/m. The Org_diff represents the difference ratio between the theoretical expected T_θ and N/m.

a local search run. For non-resampling search, since the first global search time is approximated as N/m, when in the second global search run, there will be $N - N/m - m/bs$ non-visited points and $m - m/bs$ non-visited subthreshold points, the time required for the second global search run is

$$\frac{N - F(1) - \dfrac{m}{bs}}{m - \dfrac{m}{bs}} .$$

Let us denote the i-th global search time as $F(i)$, where i is referred as to the number of global search runs. We have

$$\begin{aligned}
F(1) &= \frac{N}{m} \\
F(2) &= \frac{N - F(1) - \dfrac{m}{bs}}{m - \dfrac{m}{bs}} \\
F(3) &= \frac{N - (F(1) + F(2)) - \dfrac{2m}{bs}}{m - \dfrac{2m}{bs}} \\
F(k) &= \frac{N - \sum_{i=1}^{k-1} F(i) - \dfrac{(k-1)m}{bs}}{m - \dfrac{(k-1)m}{bs}} .
\end{aligned} \tag{2}$$

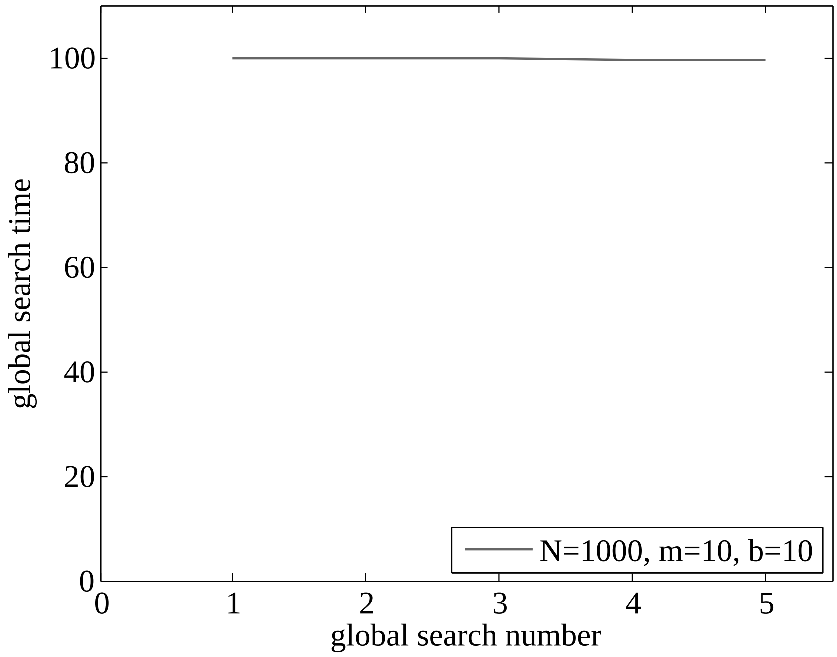
Fig. 4 illustrates the global search time, estimated by Equation (2), with respect to the number of global search runs for different distributions of quasi-basins. Fig. 4(a), with N=1000, m=10, b=10, and s=1, represents a case of a scarce small quasi-basin distribution. In this case, the global search time is large and does not change much as the number of global search runs increases. The second case illustrated in Fig. 4(b) represents a case of a scarce large quasi-basin distribution with N=1000, m=900, b=10, and s=1. The global search time is small and slightly increases as the number of runs increases. The last case illustrated in Fig. 4(c) represents a case of fully uniform distributed quasi-basins with N=1000, m=500, b=500, and s=1. In this case, the global search time is also small and slightly decreases as the number of runs increases.

We are now ready to calculate the upper bound for the last global search run. As the QBC only defines the number of subthreshold points and the number of quasi-basins, local search of the subthreshold seeker can be considered as stochastic non-repeated sampling in the set of subthreshold points. Since the minimum resides in $S_m(f)$ with m points, for the stochastic non-repeated sampling, it is expected to sample $(m+1)/2$ points before the minimum point can be found. In the uniform Quasi-Basin Class, because each quasi-basin are about the same size and a local search run visits $1/s$ of the points in a quasi-basin, it is expected to require $k = \lceil bs/2 \rceil$ local search runs to find the minimum, which implies $k = \lceil bs/2 \rceil$ global search runs are required. When bs is even, the k-th global search run requires

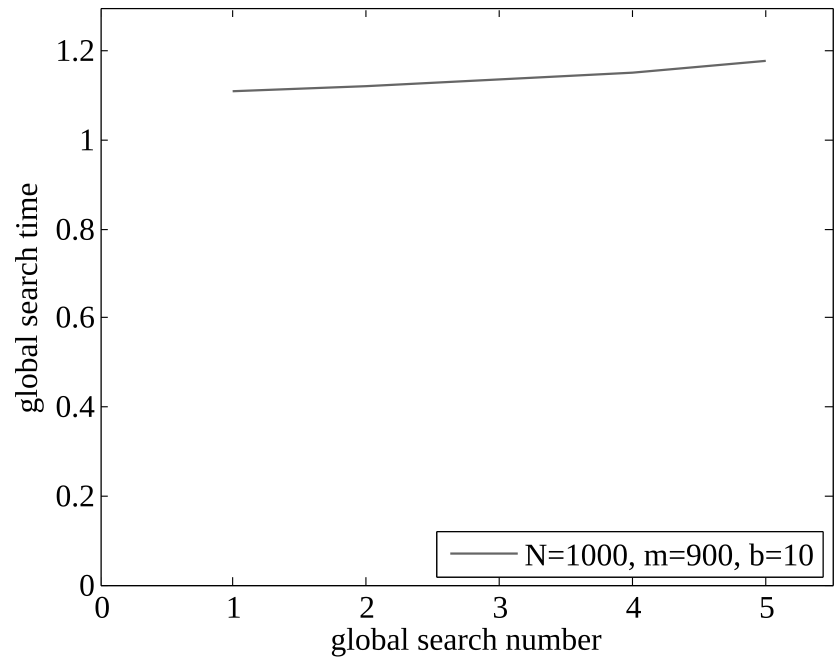
$$\begin{aligned}
F(k) &< \frac{bsN - (k-1)m}{bsm - (k-1)m} \\
&= \left(\frac{2bs}{bs+2}\right)\frac{N}{m} - \left(\frac{bs-2}{bs+2}\right) .
\end{aligned} \tag{3}$$

When bs is odd, the k-th global search run requires

$$\begin{aligned}
F(k) &< \frac{bsN - (k-1)m}{bsm - (k-1)m} \\
&= \left(\frac{2bs}{bs+1}\right)\frac{N}{m} - \left(\frac{bs-1}{bs+1}\right) .
\end{aligned} \tag{4}$$

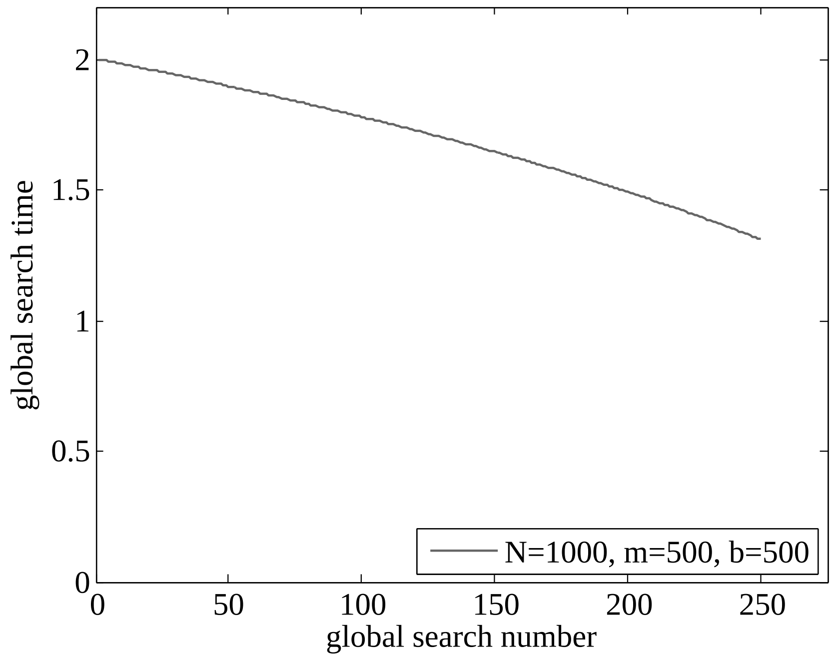


(a)



(b)

Figure 4: Global search time with respect to the number of global search runs when the exhaustive local search is applied. Fig. 4(a) illustrates a case of a scarce small quasi-basin distribution. Fig. 4(b) illustrates a case of a scarce large quasi-basin distribution. Fig. 4(c) illustrates a case of fully uniform distributed quasi-basins.



(c)

Figure 4: Global search time with respect to the number of global search runs when the exhaustive local search is applied. Fig. 4(a) illustrates a case of a scarce small quasi-basin distribution. Fig. 4(b) illustrates a case of a scarce large quasi-basin distribution. Fig. 4(c) illustrates a case of fully uniform distributed quasi-basins.

Both Equations (3) and (4) indicate that the final global search time would be no more than twice of the amount of the first global search. Fig. 5 illustrates the last global search time divided by the first global search time, T_θ, with respect to m when the exhaustive local search is applied. All the last global search times, b=10, b=100, and b=500, initially increases as m increases and reaches a peak followed by gradually degradation. The smaller b is, the smaller m the peak appears at with a greater peak value. Generally, when the number of quasi-basins is considerably large, the smaller the last global search time is. Overall, the last global search time are within 0.4 to 1.6 times of T_θ. As indicated in Fig. 4, the variation of the global search time with respect to the number of global search runs are approximately linear. Thus, we can approximate the average global search time as the average of the first global search time and the last global search time. Observing the case of b=10 in Fig. 5, we can find that for scarce quasi-basin distributions, the average global search time will be slightly larger than the first global search time T_θ, within $0.3T_\theta$.

4.3 Discussion

Overall, in this section, we can see that the expected global search time to hit a subthreshold point in local search zones is inversely proportional to the partition of local search zones in the search space. Because the uniform random search is employed as the global search component, such results illustrate a baseline behavior of global search in common definitions. It can be observed that when the size ratio between local search zones and the whole search space is very small, the expected global search time will be immensely long because finding a local search zone is very difficult. If the ratio is not very small and permits an acceptable probability to be hit by global search, the expected global search time will drop dramatically. In this case, since

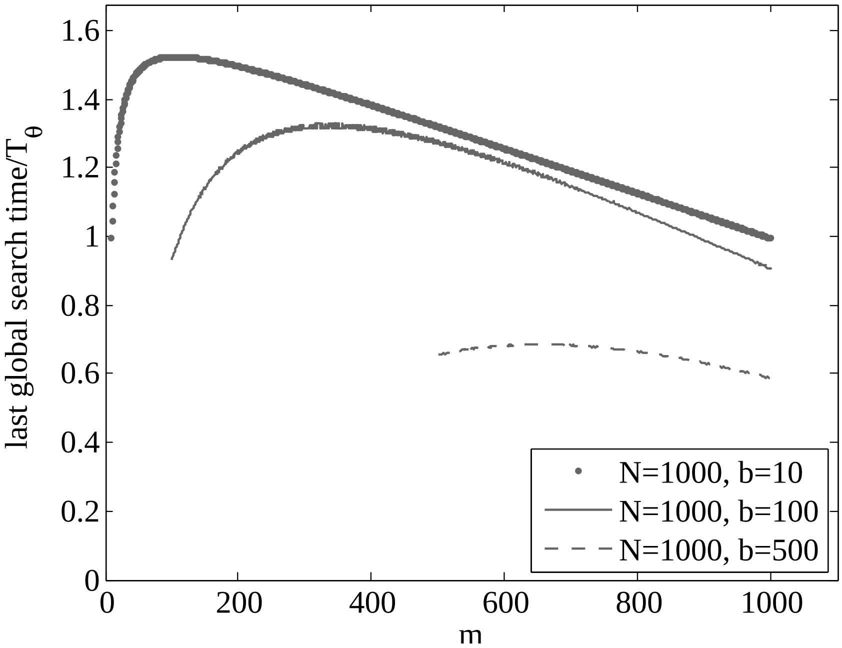


Figure 5: The last global search time divided by the first global search time, T_θ, with respect to m when the exhaustive local search is applied.

the size of local search zones is still small, the local search operator requires a relatively short time to find the optimum solution.

5 Subthreshold Seeker on QBC

In this section, we will formulate the expected evaluation time for a subthreshold seeker on a $\mathcal{Q}(G, \mathcal{Y}, m, b)$ as the sum of expected total global search time and the expected total local search time. The expected total global search time is the product of the expected time for global search to enter a local search zone and the expected number of global search runs. The expected total local search time is merely the expected time for local search to find the global optimal points among the local search zones which is proportional to the partition of the local search zones in the search space. In this manner, the derived formula can depict how the collaboration between global search and local search influences the performance of memetic algorithms. Then, we will propose a sampling test scheme to empirically verify the behavior of the subthreshold seeker on various QBCs. Finally, we will discuss what the derived model implies in the design of memetic algorithms and compare the observations to that made by other previous studies in the literature.

5.1 Evaluation Time of Subthreshold Seeker

With the global search time ready, we can now estimate the time to find the minimum point, i.e., the evaluation time of subthershold seeker T, with the equation

$$T = \frac{cN}{m} \cdot \left\lceil \frac{bs}{2} \right\rceil + \frac{m+1}{2} \ . \tag{5}$$

The expected total time over a QBC is considered as the sum of the expected total global search time, the first term, and the expected total local search time, the second term. As discussed in the previous section, it is expected to apply $\frac{bs}{2}$ local search runs in order to find the global optima, and thus, $\frac{bs}{2}$ global search runs. $\frac{cN}{m}$ represents the average global time with c varies between 0.4 and 1.6, and $\frac{m+1}{2}$ corresponds to the expected time to find the minimum among subthreshold points. By differentiating the formula, we can find that if we set m to about $\sqrt{bscN}$, the subthreshold seeker can achieve a minimum evaluation time T about $\sqrt{bscN} + 0.5$. Note that the total global search time and the total local search time are near identical when the overall evaluation time is minimum. The following sections verify Equation (5) with the results obtained by our experiments as well the results in the related studies.

5.2 Sampling Test Scheme

For empirical convenience, we implement the simplest case of QBC, pathwise Quasi-Basin Class (PQBC). PQBC is the class of functions with their spatial structure of a simple path and the value of every vertex is a distinct integer in $\mathcal{Y} = \{1, 2, \cdots, n\}$, where $n = |\mathcal{X}|$. PQBC is formally defined as

Definition 4 (Pathwise Quasi-Basin Class, PQBC) *Given a finite set* $\mathcal{Y} = \{1, 2, \cdots, n\} \subset \mathbb{N}$ *and a simple path* $G = \overline{v_1 v_2 \ldots v_n}$*, the pathwise Quasi-Basin Class with* b *distinct quasi-basins and* m *subthreshold vertex is defined as* $\mathcal{Q}^+(G, \mathcal{Y}, m, b)$*.*

To investigate the expected subthreshold seeker behavior over a specific PQBC, we sample functions from a specific PQBC via the PQBC sampler of which the pseudo code is shown in Fig. 6 to generate functions in the pathwise Quasi-Basin Class with uniform basins and non-uniform basins. Function $UniformPick$ in Fig. 6 samples the input set uniformly at random, returns the sampled value, and removes that value from the input set. Function Pop outputs the first value of a sequence and removes the first value from the sequence. The pathwise QBC sampler separates the input values $1, 2, \ldots, n$ into two sets: the set with superthreshold points (GT) and the set with subthreshold points (ST) and then uniformly randomly picks one point from GT and one from ST to construct the basic sequence of a quasi-basin. After every quasi-basin has its basic sequence, the next step is to assign all the subthreshold points to each quasi-basin. When the input boolean parameter $Usize$ is set to $True$, the sampler uniformly assigns subthreshold points to every quasi-basin. Otherwise, each subthreshold point is assigned to an arbitrary quasi-basin. We then uniformly randomly pick members in the quasi-basin sequences and GT. When a quasi-basin sequence is picked, this sequence is assigned as the values of next vertices. Fig. 7 illustrates an example of functions in the pathwise Quasi-Basin Class which consists 20 points with three quasi-basins containing eight subthreshold points.

Now, to empirically verify the time for a subthreshold seeker to find the minimum point of a given PQBC $\mathcal{Q}^+(G, \{1, 2, \cdots, n\}, m, b)$, we set the subthresold seeker's threshold to $\beta_m(f)$, i.e. m. In the following sections, we verify Equation (5) with the average time for a subthreshold seeker to find the minimum on various PQBCs. For each PQBC, the performance of the subthreshold seeker is measured by averaging 50 function instances with 20 independent runs on each function instance.

5.3 Experimental Results

Fig. 8 compares the average evaluation time for a subthreshold seeker with the exhaustive local search and the theoretical evaluation time derived by Equation (5) with respect to m on different pathwise uQBCs with $n = 1000$ and b= 1, 10, and 250. The solid lines, Ttheo and Ttheo1, indicate the theoretical evaluation time derived from Equation (5) with $c = 1$, while

```
procedure PATHWISE QBC SAMPLER(v1v2...vn, Y = {1, 2, ..., n}, m, b, Usize)
    ST ← {1, 2, ..., m}
    GT ← {m + 1, m + 2, ..., n}
    i ← 1
    while i ≤ b do
        gt ← UniformPick(GT)
        st ← UniformPick(ST)
        qb_i ← (gt, st)
        i + 1
    end while
    while ST ≠ ∅ do
        st ← UniformPick(ST)
        if Usize then
            i ← i + 1  mod b
        else
            i ← Uniform([1, b])
        end if
        qb_i ← (qb_i, st)
    end while
    QB ← ∪qb_i
    S ← GT ∪ QB
    i ← 1
    while i ≤ n do
        v ← UniformPick(S)
        while length(v) > 1 do
            f(v_i) ← Pop(v)
        end while
        f(v_i) ← v
        i ← i + 1
    end while
    return f
end procedure
```

Figure 6: Pathwise QBC Sampler.

the dashed line Ttheo2 indicates that with $c = 1.5$. The circle (ls) and the cross (gs) represent the average total number of sampling used by local search and global search respectively.

Because there is only one quasi-basin in Fig. 8(a), one global search is required. The proposed model matches the empirical result in this case. In Fig. 8(b), the empirical result matches the proposed model with $c = 1.5$. Such a situation may be caused by the significant global search time growth we observed in Fig. 5. The global search time grows as high as 1.5 when both b and m are quite small. Fig. 8(c) illustrates with large b, the subthreshold seeker performs worse than random search with its evaluation time exceed half of the search space size. Such a result, consisting with the proposed model, indicates that when the number of basins are greater than a quarter of the search space, the problem is unsearchable. Note that in these three cases, the average total local search time and the average total global search time also consist with our theoretical model. The average local search time is about $m/2$ while the average global search time matches Equation (1).

Fig. 9 illustrates the evaluation time of a subthreshold seeker with the exhaustive local search on a non-uniform pathwise QBCs with n=1000 and b=10. Compare the results to that shown in

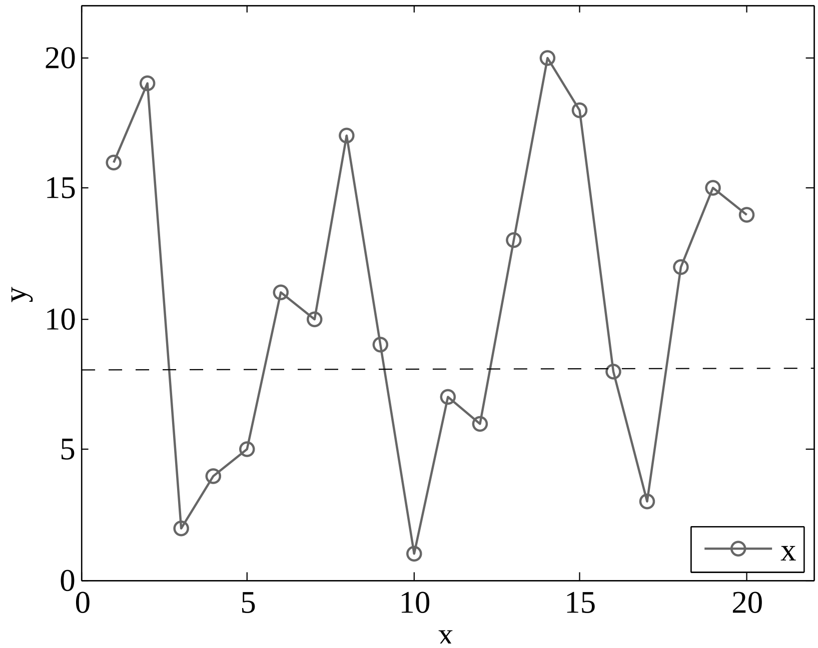


Figure 7: An example of functions belonging to $\mathcal{Q}^{+}(G, \{1, 2, \cdots, 20\}, 8, 3)$.

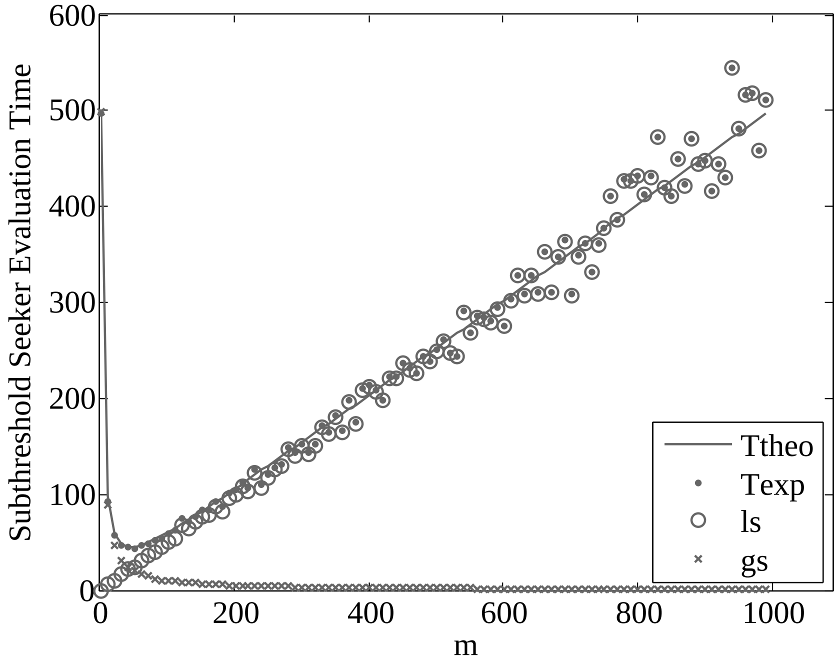
Fig. 8(b), we can observe that although the deviations of the empirical results on non-uniform QBCs is slightly larger than that on uniform QBCs, the two sets of results basically resemble each other. Because the non-uniform pathwise QBCs have every basin's expected size identical, i.e., the mean value, it can be expected that a subthreshold seeker behave statistically similarly on non-uniform and on uniform QBCs.

Figs. 10 and 11 compare the theoretical optimal evaluation time with the empirical results with respect to b and n, respectively. In both cases, the exhaustive local search is applied. The solid lines in both figures indicate the optimal theoretical evaluation time predicted by Equation (5) with $c = 1$, and the dashed line indicates that with $c = 1.5$. Both figures demonstrate that the theoretical prediction and the empirical results are in good agreement, and therefore, Equation (5) is dimensionally validated for different factors.

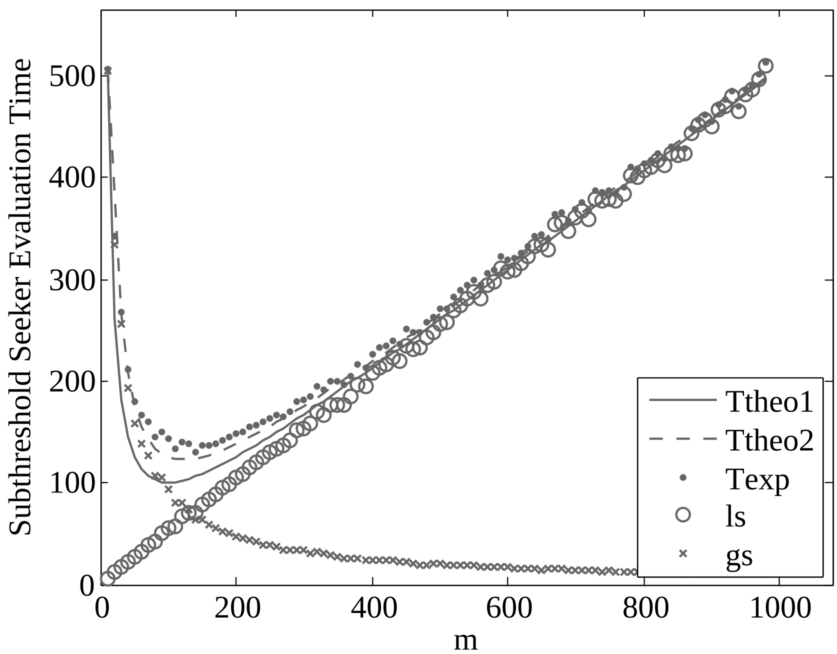
Fig. 12 illustrates the evaluation time with respect to m when non-exhaustive local search components, i.e., $s > 1$, are used. In both cases, the solid lines represent the theoretical evaluation time predicted by Equation (5) with $c = 1.5$. These empirical results also well match the proposed theoretical model Equation (5).

5.4 Discussion

Since the proposed model of subthreshold seekers on different QBCs has been validated by the empirical results in the previous section, in this section, we discuss more about what this theoretical model implies in the design of memetic algorithms. We will start from the behavior of the subthreshold seeker which is taken as a representative archetype of memetic algorithms. Conceptually, the subthreshold seeker splits the search space into two sub-spaces: one for the global search algorithm to explore and the other for the local search algorithm to exploit. The criterion to switch between these two sub-spaces is the objective function value of search points. Employed by the subthreshold seeker as global search, the uniform random search provides the maximal diversity that other heuristic biased global search algorithms usually cannot provide.

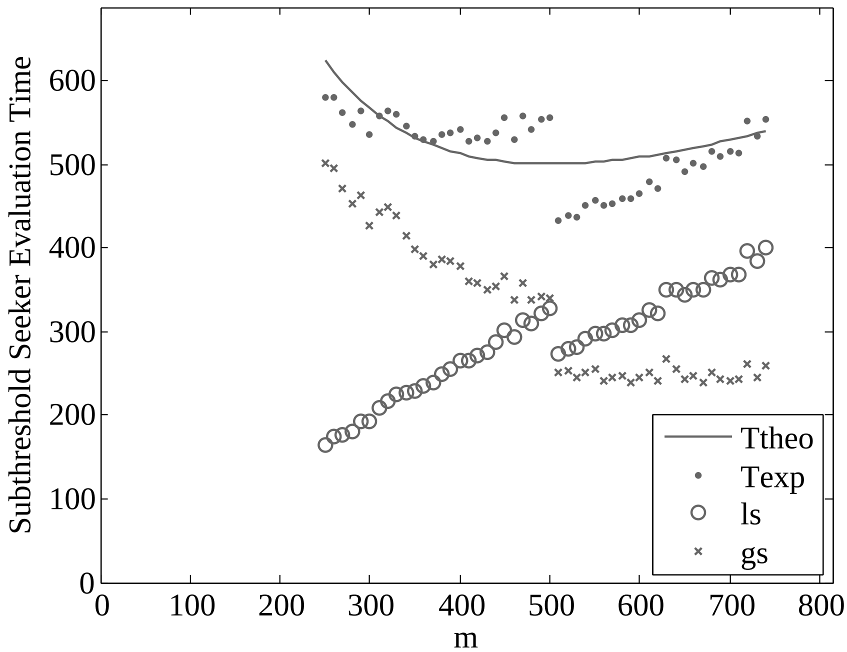


(a) n=1000, b=1, uniform quasi-basin



(b) n=1000, b=10, uniform quasi-basin

Figure 8: The time for a subthreshold seeker to find the minimum with respect to m when (a) $n = 1000$ and $b = 1$; (b) $n = 1000$ and $b = 10$; (c) $n = 1000$ and $b = 250$. The solid line, Ttheo and Ttheo1, represents the theoretical value derived from Equation (5), and the dot, Texp, represents the average time for a subthreshold seeker to find the minimum on different PQBCs. The average total sampling counts used by local search and global search are also recorded as ls and gs, respectively.



(c) n=1000, b=250, uniform quasi-basin

Figure 8: The time for a subthreshold seeker to find the minimum with respect to m when (a) $n = 1000$ and $b = 1$; (b) $n = 1000$ and $b = 10$; (c) $n = 1000$ and $b = 250$. The solid line, Ttheo and Ttheo1, represents the theoretical value derived from Equation (5), and the dot, Texp, represents the average time for a subthreshold seeker to find the minimum on different PQBCs. The average total sampling counts used by local search and global search are also recorded as ls and gs, respectively.

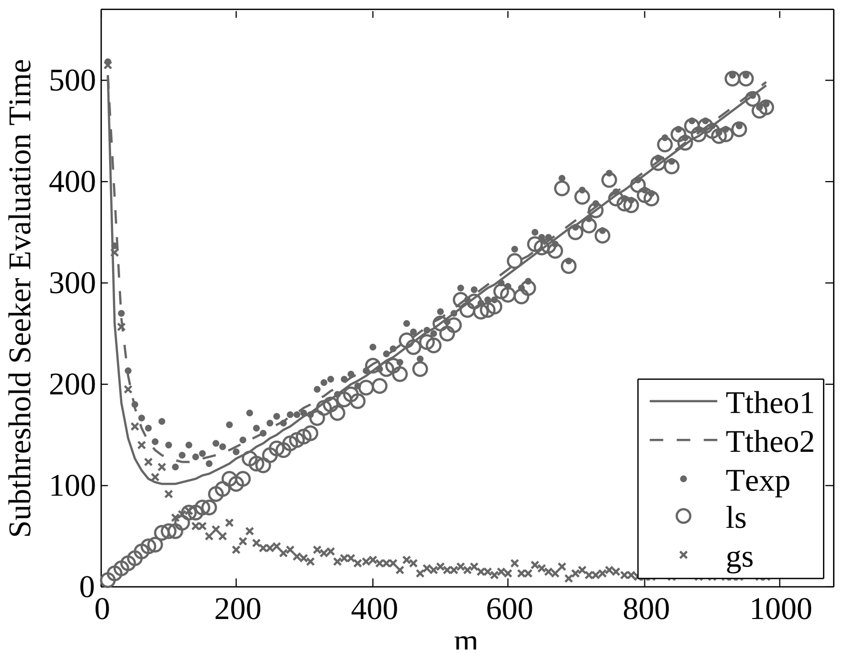


Figure 9: n=1000, b=10, non-uniform quasi-basin.

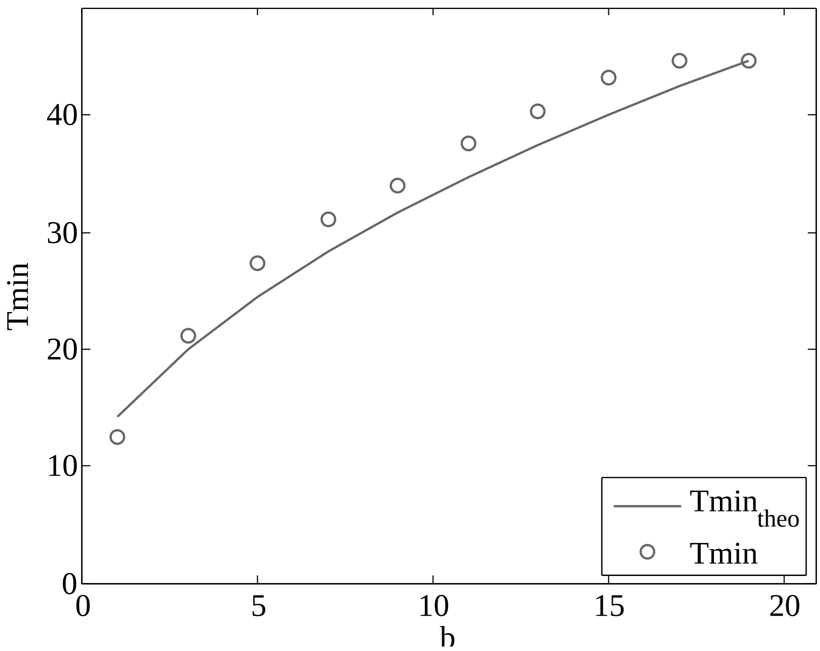


Figure 10: The optimal evaluation time versus b when n=100. The solid line indicates the theoretical value predicted by Equation (5) with $c = 1$, and the dots indicate the empirical results.

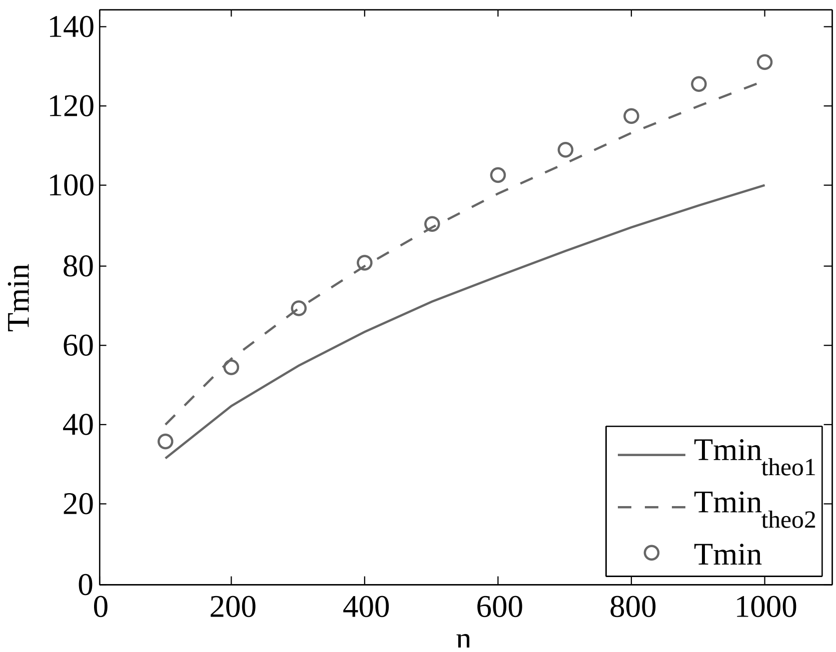
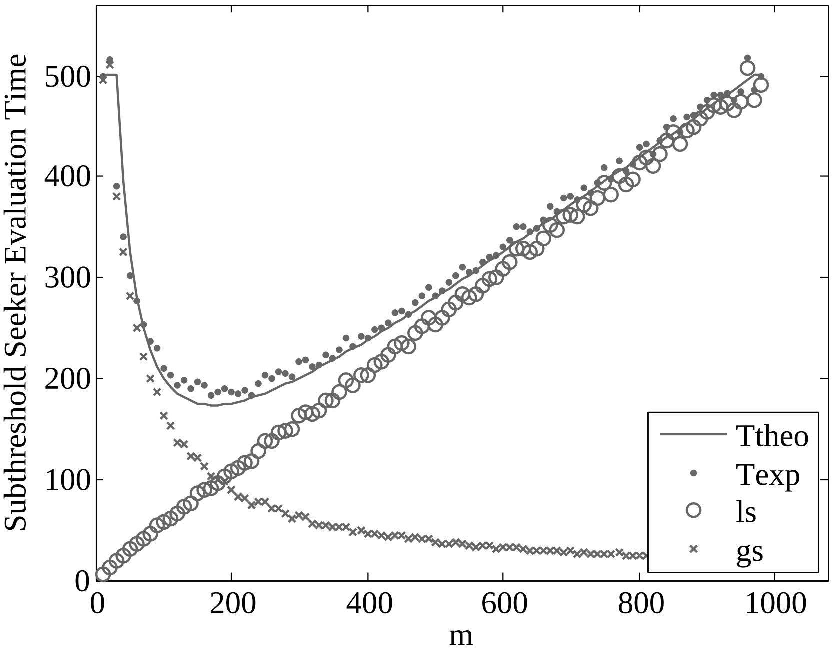
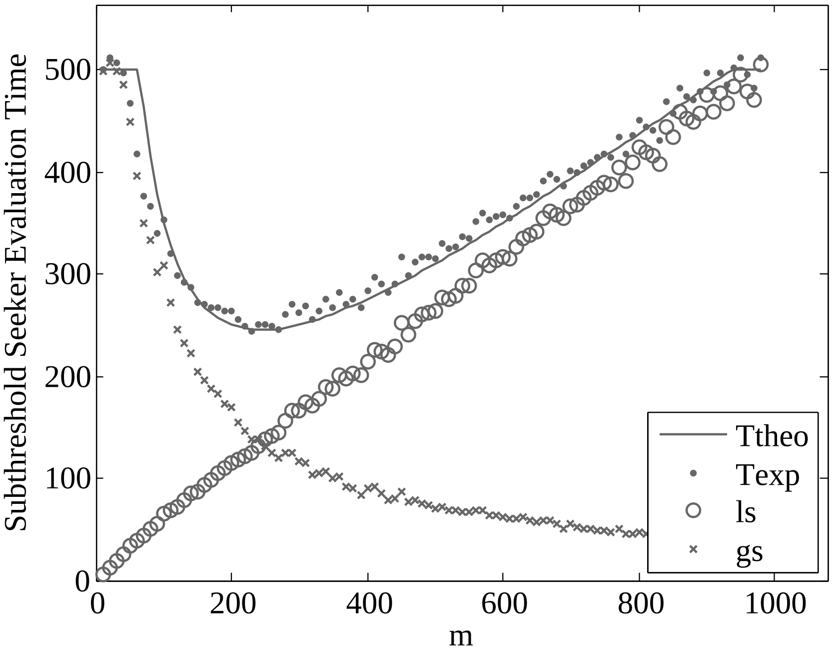


Figure 11: The optimal evaluation time versus n when b=10. The solid line indicates the theoretical value predicted by Equation (5) with $c = 1$, the dashed line indicates that with $c = 1.5$, and the dots indicate the empirical results.



(a) n=1000, b=10, s=2, uniform quasi-basin



(b) n=1000, b=10, s=4, uniform quasi-basin

Figure 12: The evaluation time for subthreshold seekers with a non-exhaustive local search component ($s > 1$).

Hence, it can be considered as a perfect explorer.

On the other hand, the local search component adopted by the subthreshold seeker exploits a quasi-basin via visiting the neighbours, defined by its step size, of current search point. The local search step size of the employed local search operator is connected to how well a quasi-basin is exploited. Recall that when s=1, the exhaustive local search will eventually visit all the points in a quasi-basin. In this case, the local minimum of a quasi-basin, which may be the global minimum, will be visited, and thus, only one local search run for each basin is required. For other local search step sizes greater than one, there are chances for one local search run to miss the global minimum in a quasi-basin, and thus, more local search runs on this quasi-basin and more global search runs to hit this quasi-basin are required. The cost will be the extra global search time to enter the quasi-basin again when the algorithm guarantees non-repeated sampling. Fig. 12 and the factor s in Equation (5) demonstrate the effect of the degree of exploitation of a basin. Such an effect implies that a good local search operator ought to fully exploit the given quasi-basin, at least the local minimum resides in the quasi-basin should be found, to guarantee a good local search and global search coordination. This inference is consistent with the empirical results of studies which concluded that longer local search lengths are favored in memetic algorithms [2, 8, 9].

In the subthreshold seeker, the exhaustive local search certainly is the optimal local search algorithm that can achieve fully exploitation of a quasi-basin. As QBC defines only the number of subthreshold points and the number of quasi-basins, except that a subthreshold seeker is likely to find the neighboring subthreshold points, there is no relation among subthreshold points. In other words, in a quasi-basin, the value of each point could be arbitrarily random. In this perspective, the local search part of the subthreshold seeker can be considered as a random walker without repetition in the subthreshold sub-space. Thus, the subreshold seeker effectively employs random search of different kinds in different sub-spaces, the superthreshold search space and the subthreshold search space, as global search and local search. The global search randomly samples the search space to find a subthreshold point in a quasi-basin, while the local search randomly samples in the subthreshod space which consists distinct, separate quasi-basins to find the global minimum. In this description, the number of subthreshold points and the number of basins are certainly two key factors to the MA performance.

In our theoretical model for the subthreshold seeker, as the local search is essentially a random search that samples the subthreshold search space, regardless of other parameters, it is expected to sample half of the subthreshold search space to find the global minimum. On the other hand, the expected total global search time is related to several parameters: the size of the search space, the number of subthreshold points, the number of quasi-basins, and the local search step size. The former two parameters are related to the time for the global search to find a subthreshold point, and the other two parameters, how many quasi-basins are required to be exploited and how well a quasi-basin will be exploited, are related to the number of global search runs should be used to find quasi-basins until the global minimum is found. The best threshold derived from the proposed model is roughly $\sqrt{bcsN}$ with the best evaluation time $\sqrt{bcsN}$ which has equal shares of the total global search time and the total local search time. Because the global search time drops quickly as m increases and the local search time increases linearly as $m/2$, it is reasonable that the optimal search occurs at a rather small m relative to the search space size.

Generally, we can assume that the average global search time to enter a local search zone is inversely proportional to the size ratio between local search zones and the search space, while the average total local search time is proportional to the size ratio. Putting these two terms together, we can obtain the "V-shaped" curve which resembles those derived from Equation (5). This V-shaped curve implies that a good collaboration between global search and local search should guarantee a short average global search time to hit local zones and sufficiently small sizes of local zones for the local search to exploit. Regarding the influence of the size of local search zones on the average global search time to find local search zones and the average time for local search to find the optimal point, memetic algorithms which have small sized local search zones will perform better. This observation is consistent with the use of elitism in local search candidate selection and the infrequent local search principle in quite a number of research works [1, 2, 3, 4, 5]. It is also notable that several studies adopt a local search/global search ratio which is consistent with our theoretical model [29].

Despite the aforementioned consistency between the proposed model and the elitism based strategy in local search candidate selection, infrequent local search, long local search length, and local search/global search ratio, some previous studies also show a strong connection to our model. In an investigation on the balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling [8], the authors examined 132 combinations of 11 values of k, which is the maximum number of examined neighbors of the current solution, and 12 values of p_{LS}, which is the local search probability applied to the tournament selected individuals. The former factor k connects to the degree of how well a feasible sub-region can be exploited, and the second factor p_{LS} connects to the threshold that triggers local search. The authors found that the combination of the maximum k value and the minimum nonzero p_{LS} value achieved the best performance, the lowest cost of flowshop scheduling, in their experiments. The V-shaped curve of cost along the axis of the maximum k with respect to p_{LS} in their Fig. 13 resembles our V-shaped curve of the evaluation time in Fig. 8. Because the stop criterion of their experiments is the evaluation of a fixed number of points, the factor combinations that requires less evaluation time to find the global minimum will have better solution quality, i.e., lower cost. This agreement implies that the proposed model may be adopted to give a theoretical explanation to the internal working of their multi-objective memetic algorithms.

Another set of intriguing empirical results is presented in the study of parameterizing local search [10]. In that study, the authors applied a hybrid approach to the memory cost minimization problem (MCMP) with various local search parameter settings. The local search parameter refers to the intensity of the local search method, a tractable algorithm called code size dynamic programming post optimization (CDPPD), applied to every individual in the population. The authors depicted in Fig. 13 in their paper that when a fixed run time is used, the number of generations completed decreases rapidly as the local parameter increases. Because the local search is applied to every individual, the global search time is proportional to the number of generations. It implies that the average global search time is also proportional to the number of generations. Note that this curve, indicating the average global search time, resembles the expected T_θ in our Fig. 2. As the average global search time is illustrated and the expected local search time will be proportional to the intensity of local search, summing up the expected global search time and the expected local search time, a V-shaped curve of the expected evaluation time with respect to the intensity of local search will be obtained. Fig. 12 in their study illustrates the attained solution quality, lower cost preferred, versus the setting of local search intensity. As previously discussed, lower attained cost in a given fixed time leads to shorter expected time to find the optimal solution. This figure also resembles the V-shaped curve of our theoretical model which confirms that the proposed model is quite applicable to their conclusions.

Although in these two studies, practical memetic algorithms, instead of the subthreshold seeker, are employed and investigated, the trend of their evaluation time resembles the proposed model developed based on the subthreshold seeker. It indicates that our theoretical model is indeed representative of memetic algorithms as we previously inferred. Another interesting study is the optimal bounds on finding fixed points of contraction mappings proved by [37]. In this investigation, the authors presumed that the expected lower bound of a randomized algorithm to find the fixed point of a contraction mapping $f : M \leftarrow M$ on a finite metric space (M,d) is $\Omega(\sqrt{|M|})$ and proved this bound is valid. In this fixed point problem, given any point $x \in M$ with the $d(x, f(x))$ the k-th largest, one can find the fixed point with k steps via a valid deterministic algorithm. Consider the set exploited by the deterministic algorithm as the subthreshold sub-space which consists only one quasi-basin and the random sampling process to find a starting point for the deterministic algorithm as global search, according to Equation (5), the best size of the subthreshold sub-space should be $\Omega(\sqrt{|M|})$ resulting in an expected optimal evaluation time of $\Omega(\sqrt{|M|})$. Thus, our theoretical model can also provide a reasonable, theoretical interpretation to this presumed value $\Omega(\sqrt{|M|})$.

Although the proposed model is developed based on the subthreshold seeker on QBCs, it is applicable to common, practical memetic algorithms. Despite the aforementioned related studies and observations, as both global search and local search of the subthreshold seeker resembles random search, the exploration and exploitation behavior switched by a threshold is valid to be used to model the internal workings of memetic algorithms. Implanting the quasi-basin factor to the coordination mechanism further enhances the generality of the proposed model. Our model delineates an expected coordination behavior over classes of functions with the distribution of local search zones on their landscapes. Because every problem with a corresponding memetic algorithm can be categorized as one specific QBC, the proposed model is therefore applicable to a broad range of problems as well as memetic algorithms.

6 Summary and Conclusions

In this paper, based on the concept of local search zones, we proposed the Quasi-Basin Class (QBC) and took the subthreshold seeker as a representative archetype of memetic algorithms to develop a theoretical model. The developed model not only depicts well how the collaboration between local search and global search in memetic algorithms on various problems is related to the expected evaluation time but also consists with the observations made by previous studies in the literature. Such a consistency implies that the proposed model is representative and may be applicable to the future analysis of memetic algorithms.

In the present framework, the subthreshold seeker is simple but definite as the expected global search time, expected local search time, and numbers of local search runs and global search runs are easy to assess. To follow the approach to analyze more complicated memetic algorithms on various problems, one needs to assess the following quantities with respect to the distribution of local search zones: the expected global search time to enter local search zones and the expected local search time to find the optimal point. The efficiency of the local search algorithm and the number of local search zones also much influences the MA performance and the optimal size of local search zones. Utilizing an objective function value as the threshold to create local search zones in the search space may be a good choice to coordinate global search and local search. In this way, the expected time for a global search algorithm to encounter a local search zone and the expected time for a local search algorithm to find the optimal point in local search zones may be assessable. With the technique in hands and certain understandings of the underlying problem, one may design good memetic algorithms according to the proposed model.

Acknowledgments

The work was supported in part by the National Science Council of Taiwan under Grant NSC 98-2221-E-009-072. The authors are grateful to the National Center for High-performance Computing for computer time and facilities.

References

[1] W. E. Hart, "Adaptive global optimization with local search," Ph.D. dissertation, University of California, 1994.

[2] M. W. S. Land, "Evolutionary algorithms with local search for combinatorial optimization," Ph.D. dissertation, University of California, 1998.

[3] K.-H. Liang, X. Yao, and C. Newton, "Evolutionary search of approximated n-dimensional landscapes," *International Journal of Knowledge Based Intelligent Engineering Systems*, vol. 4, no. 3, pp. 172–183, 2000.

[4] M. Tang and X. Yao, "A memetic algorithm for VLSI floorplanning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 37, no. 1, pp. 62–69, 2007.

[5] Y.-H. Liu, "A hybrid scatter search for the probabilistic traveling salesman problem," *Computers & Operations Research*, vol. 34, no. 10, pp. 2949–2963, 2007.

[6] M. Lozano, F. Herrera, N. Krasnogor, and D. Molina, "Real-coded memetic algorithms with crossover hill-climbing," *Evolutionary Computation*, vol. 12, no. 3, pp. 273–302, 2004.

[7] S. Garcia, J. R. Cano, and F. Herrera, "A memetic algorithm for evolutionary prototype selection: A scaling up approach," *Pattern Recognition*, vol. 41, no. 8, pp. 2693–2709, 2008.

[8] H. Ishibuchi, T. Yoshida, and T. Murata, "Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 204–223, april 2003.

[9] Z. Zhu, Y.-S. Ong, and M. Dash, "Wrapper-filter feature selection algorithm using a memetic framework," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 37, no. 1, pp. 70–76, 2007.

[10] N. Bambha, S. Bhattacharyya, J. Teich, and E. Zitzler, "Systematic integration of parameterized local search into evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 2, pp. 137–155, april 2004.

[11] N. Krasnogor, "Studies on the theory and design space of memetic algorithms," Ph.D. dissertation, University of the West of England, 2002.

[12] Y. S. Ong and A. J. Keane, "Meta-Lamarckian learning in memetic algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 2, pp. 99–110, 2004.

[13] J. Smith, "Coevolving memetic algorithms: A review and progress report," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 37, no. 1, pp. 6–17, feb. 2007.

[14] Q. H. Nguyen, Y.-S. Ong, and M. H. Lim, "A probabilistic memetic framework," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 3, pp. 604–623, 2009.

[15] G. Rudolph, "Convergence properties of evolutionary algorithms," Ph.D. dissertation, Verlag Dr. Kovač, 1997.

[16] H.-G. Beyer, *The Theory of Evolution Strategies*, ser. Natural Computing Series. Springer, 2001, ISBN: 978-3-540-67297-5.

[17] M. Jiang, Y. Luo, and S. Yang, "Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm," *Information Processing Letters*, vol. 102, no. 1, pp. 8–16, 2007.

[18] H. Mühlenbein, "How genetic algorithms really work: I. mutation and hillclimbing," in *Proceedings of the Second Conference on Parallel Problem Solving from Nature (PPSN II)*, 1992, pp. 15–25.

[19] S. Droste, T. Jansen, and I. Wegener, "A rigorous complexity analysis of the (1 + 1) evolutionary algorithm for separable functions with boolean inputs," *Evolutionary Computation*, vol. 6, no. 2, pp. 185–196, 1998.

[20] J. Garnier, L. Kallel, and M. Schoenauer, "Rigorous hitting times for binary mutations," *Evolutionary Computation*, vol. 7, no. 2, pp. 173–203, 1999.

[21] S. Droste, T. Jansen, and I. Wegener, "On the analysis of the (1+1) evolutionary algorithm," *Theoretical Computer Science*, vol. 276, no. 1-2, pp. 51–81, 2002.

[22] J. He and X. Yao, "Towards an analytic framework for analysing the computation time of evolutionary algorithms," *Artificial Intelligence*, vol. 145, no. 1-2, pp. 59–97, 2003.

[23] T. Jansen and I. Wegener, "The analysis of evolutionary algorithms – a proof that crossover really can help," *Algorithmica*, vol. 34, no. 1, pp. 47–66, 2008.

[24] D. Sudholt, "On the analysis of the (1+1) memetic algorithm," in *Proceedings of ACM SIGEVO Genetic and Evolutionary Computation Conference 2006 (GECCO-2006)*, 2006, pp. 493–500.

[25] ——, "The impact of parametrization in memetic evolutionary algorithms," *Theoretical Computer Science*, vol. 410, no. 26, pp. 2511–2528, 2009.

[26] A. Sinha, Y.-p. Chen, and D. E. Goldberg, "Designing efficient genetic and evolutionary algorithm hybrids," in *Recent Advances in Memetic Algorithms*, ser. Studies in Fuzziness and Soft Computing. Physica-Verlag, 2004, vol. 166, pp. 259–288.

[27] S. Christensen and F. Oppacher, "What can we learn from no free lunch? a first attempt to characterize the concept of a searchable function," in *Proceedings of the Genetic and Evolutionary Computation Conference 2001 (GECCO-2001)*, 2001, pp. 1219–1226.

[28] D. Whitley and J. Rowe, "Subthreshold-seeking local search," *Theoretical Computer Science*, vol. 361, no. 1, pp. 2–17, 2006.

[29] D. Molina, M. Lozano, C. Garcia-Martinez, and F. Herrera, "Memetic algorithms for continuous optimisation based on local search chains," *Evolutionary Computation*, vol. 18, no. 1, pp. 27–63, 2010.

[30] F. Neri, J. Toivanen, G. L. Cascella, and Y.-S. Ong, "An adaptive multimeme algorithm for designing HIV multidrug therapies," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 4, no. 2, pp. 264–278, 2007.

[31] J. Tang, M. H. Lim, and Y. S. Ong, "Diversity-adaptive parallel memetic algorithm for solving large scale combinatorial optimization problems," *Soft Computing*, vol. 11, no. 9, pp. 873–888, 2007.

[32] P. Merz and B. Freisleben, "Fitness landscapes and memetic algorithm design," in *New Ideas in Optimization.* McGraw-Hill Ltd., 1999, pp. 245–260.

[33] C. Witt, “Runtime analysis of the (μ + 1) EA on simple pseudo-boolean functions,” *Evolutionary Computation*, vol. 14, no. 1, pp. 65–86, 2006.

[34] B. Doerr, N. Hebbinghaus, and F. Neumann, “Speeding up evolutionary algorithms through asymmetric mutation operators,” *Evolutionary Computation*, vol. 15, no. 4, pp. 401–410, 2007.

[35] A. H. Wright and J. N. Richter, “Strong recombination, weak selection, and mutation,” in *Proceedings of ACM SIGEVO Genetic and Evolutionary Computation Conference 2006 (GECCO-2006)*, 2006, pp. 1369–1376.

[36] L. Kallel, B. Naudts, and C. R. Reeves, “Properties of fitness functions and search landscapes,” in *Theoretical aspects of evolutionary computing.* Springer-Verlag, 2001, pp. 175–206, ISBN: 3-540-67396-2.

[37] C.-L. Chang and Y.-D. Lyuu, “Optimal bounds on finding fixed points of contraction mappings,” *Theoretical Computer Science*, vol. 411, no. 16-18, pp. 1742–1749, 2010.