International Journal of Computational Intelligence and Applications
Vol. 7, No. 2 (2008) 1–14

ICP Imperial College Press
www.icpress.co.uk

# A HYBRID ESTIMATION OF DISTRIBUTION ALGORITHM FOR CDMA CELLULAR SYSTEM DESIGN

JIANYONG SUN

*School of Computer Science and Cercia*
*University of Birmingham*
*Edgbaston, Birmingham, B15 2TT, UK*

QINGFU ZHANG

*Department of Computer Science*
*University of Essex, Wivenhoe Park*
*Colchester, CO4 3SQ, UK*

JIN LI

*Safety and Environmental Assurance Centre*
*Unilever, Colworth Park, Sharnbrook*
*Bedford, MK44 1LQ, UK*

XIN YAO

*School of Computer Science and Cercia*
*University of Birmingham, Edgbaston*
*Birmingham, B15 2TT, UK*

Received
Revised

This paper proposes a hybrid estimation of distribution algorithm (HyEDA) to address the design problem of code division multiple access cellular system configuration. Given a service area, the problem is to find a set of optimal locations of base stations, associated with their corresponding powers and antenna heights in the area, in order to maximize call quality and service coverage, at the same time, to minimize the total cost of the system configuration. HyEDA is a two-stage hybrid approach which integrates an estimation of distribution algorithm, a K-means clustering method, and a simple local search algorithm. We have compared HyEDA with a simulated annealing method on a number of instances. Our simulation results have demonstrated that HyEDA outperforms the simulated annealing method in terms of the solution quality and computational cost.

*Keywords*: CDMA cellular system configuration design; hybrid evolutionary algorithm; estimation of distribution algorithms.

## 1. Introduction

In the last decade, code division multiple access (CDMA) has become a promising technology[1] for the mobile communication. The rapidly increasing need of the cellular communication raises several problems in optimization. In the design of a cellular communication network, the quality of service, the service coverage and the network cost are the three major concerns among many others that need to be optimized. These three factors are largely influenced by certain design parameter settings, such as the number of based stations (BSs), the locations of BSs, as well as the powers and antenna heights associated with each BS. In general, the more BSs are to be set up, the higher their powers and antenna heights are used, the larger the service area will be covered and the better the quality of service is, but the higher cost the network incurs.

The design of a cellular system can thus be treated as a multi-criteria optimization problem. Melachrinoudis and Rosyidi[2] transformed this problem into a single objective optimization problem by aggregating these three objectives into one, and then used a simulated annealing method for optimizing the single objective. A similar optimization problem, called the antenna placement problem (APP),[22,32] where only two parameters, i.e., the locations of BSs and antenna powers, need to be configured for cellular wireless services, has been proven to be an $\mathcal{NP}$-hard problem.[31] This fact implies that the problem considered here, which involves three parameters, is at least as hard as APP, thus should be regarded as an $\mathcal{NP}$-hard problem as well. This paper presents a hybrid estimation of distribution algorithm (HyEDA), which was developed to tackle the CDMA cellular system configuration based on the single objective problem converted by Melachrinoudis and Rosyidi.[2]

HyEDA is a two-stage hybrid method which integrates a K-means clustering method and a simple local search algorithm into an estimation of distribution algorithm. Its first stage aims to find optimal or near-optimal locations of BSs. Then its second stage is to find a optimal or near-optimal power and an antenna height for every BS by using the simple local search method. This two-stage optimization process is motivated by the fact that the allocation of BSs plays a more important role in achieving great performance of a cellular service network than the other two considerations, i.e., the power and antenna height.

The rest of the paper is organized as follows. Section 2 describes the cellular system configuration design problem proposed by Melachrinoudis and Rosyidi.[2] In Sec. 3, we discuss HyEDA in detail. Experimental results of HyEDA on several test problems, in comparison to that of the simulated annealing method, are given in Sec. 4. Section 6 concludes the paper.

## 2. Cellular System Configuration Design Problem

Given a service area, $A$, which is a two-dimensional geographical region, we could discretize $A$ into a lattice of grid points. Each BS will be located in a grid point. A grid point is denoted by its coordinate $(j, k)$, where $1 \leq j \leq M$ and $1 \leq k \leq K$. $M$ and $K$, set by service engineers, are the maximum number of rows and columns of the lattice, indicating the resolution of the lattice. The larger those values are, the higher the resolution is. Taking a higher resolution in the design would probably make the optimization problem even harder. This is because the search space of solutions is enlarged due to increasing possible locations to which where BSs can be assigned.

We also assume that a fixed number of BSs, $n$, need to be located. Each BS defines a cell (i.e., an area it can cover) which can be identified by using the propagation model and the link budget model,[2] provided that the power and antenna height of the BS are given. Given the locations, antenna powers and heights of the BSs, a demand allocation algorithm is proposed in Ref. 2 to associate the grid points to the BSs. A grid point is associated to the BS which receives the strongest reverse signal from the mobile stations (MSs), e.g., hand phones, in that grid point. After the demand allocation, we obtain a set of disjoint cells where a cell is associated with a BS. It is worth pointing out that our concern here is on the overall performance of the whole service area, i.e., the performance of all cells, rather than an individual cell. Such a concern will be reflected in the following functional definitions of three objectives, i.e., call quality, service coverage, and the cost of the system.

The call quality is determined by the bit error rate (BER) at MSs in the process of demodulation. The smaller the BER, the clearer the voice, and the higher the call quality. The BER value of a MS depends on its location within a cell. The formula for computing BERs can be found in Melachrinoudis and Rosyidi's paper.[2] Let $t_1$ be a threshold for the BERs. The deterioration of the call quality of cell $S_i$ is defined as:

$$d_{1i}^{+} = \begin{cases} \max\limits_{(j,k)\in S_i} \mathrm{BER}_{jk} - t_1, & \text{if } \max\limits_{(j,k)\in S_i} \mathrm{BER}_{jk} \geq t_1; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Thus, the deterioration of the call quality of the whole area is measured by $\max_i d_{1i}^{+}$, which should be minimized.

For the service coverage objective, the providers want to maximize the service area covered. Melachrinoudis and Rosyidi[2] introduced the $UCS$ (the uncovered traffic demand) value. The $UCS$ value of a cell $S_i$, $UCS_i$, is determined by the cellular system configuration.[2] Let $t_2$ be a threshold value. The deviation $d_{2i}^{+}$ of uncovered service of cell $S_i$ from the threshold is defined as follows.

$$d_{2i}^{+} = \begin{cases} UCS_i - t_2, & \text{if } UCS_i \geq t_2; \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Therefore, a good system configuration should minimize $\max_i d_{2i}^{+}$.

The total cost of the system configuration includes the cost of the BSs construction, their costs of powers and antenna. It can be summarized as follows[2]:

$$TCsystem = \sum_{i=1}^{n} CBS_i(p_i) + \sum_{i=1}^{n} CS_i(x_i, y_i) + \sum_{i=1}^{n} Chb_i(hi); \tag{3}$$

where $CBS_i$ is the cost of the BS $i$ depending on the power $p_i$ assigned on the BS; $CS_i$ is the cost of location $(x_i, y_i)$ for the construction of the BS $i$; $Chb_i$ is the cost of antenna with height $h_i$. The objective is further normalized by the maximum cost of cellular system configuration $TCsystem\_max$:

$$TC = \frac{TCsystem}{TCsystem\_max}, \tag{4}$$

where $TCsystem\_max$ is defined as the sum of costs of all the BSs locating in the most expensive locations with the largest power settings and the highest antenna.

The objective function of the cellular system configuration design problem is formulated as follows:

$$Z = w_1 \max_i d_{1i}^+ + w_2 \max_i d_{2i}^+ + w_3 TC, \tag{5}$$

where $w_i, i = 1, 2, 3$ are weights. Several points are worth mentioning here. First, values of the three measures are totally determined by the three configuration parameters, i.e., the locations, the powers, and the heights. Second, for simplicity, the detail of how to compute three measurements is not given here. The reader is referred to the paper of Melachrinoudis and Rosyidi.[2] Finally, in practice, the power, $p_i$, and the antenna height $h_i$ of a BS $i$ can only take values from available discrete sets of $P = [p_{\min}, p_{\max}]$ and $H = [h_{\min}, h_{\max}]$, respectively, with the corresponding cardinalities of $|P|$ and $|H|$.

## 3. HyEDA

This section describes the proposed algorithm. Basically, HyEDA is a two-stage heuristic algorithm. The first stage applies a hybrid estimation of distribution algorithm (HEDA), with the aim of finding the optimal or near-optimal locations of BSs. HEDA is regarded as the main part of HyEDA (the readers should notice the difference between HyEDA and HEDA). Following the BSs' locations being fixed at the first stage, the second stage adopts a simple local search algorithm to assign the BSs' powers and antenna heights.

### 3.1. *Algorithm framework*

Estimation of distribution algorithms (EDAs)[11,12,15,18] are a type of evolutionary algorithms (EAs). EAs, such as genetic algorithms (GAs), usually generate new offspring by using genetic operators like crossover and mutation. Different from GAs, EDAs generate new offspring by sampling from a probabilistic distribution model. An EDA extracts global statistical information from solutions visited so far to build the probabilistic distribution model. However, an EDA alone cannot efficiently search for optimal or near-optimal solutions of difficult optimization problems.[15] To improve the efficiency of EDAs, hill climbing algorithms[17,29] and other techniques should be incorporated.[30] In a previous work, EDAs were hybridized with hill climbing algorithms and other techniques to solve some $\mathcal{NP}$-hard optimization problems, such as maximum clique problem,[28] the quadratic assignment problem[26]

**Initialization.** Initial the probability model $p_0(x)$, sample *popsize* feasible solutions from it, apply local search algorithm to each solution. The resultant solutions consists of the initial population $P(0)$; Set $t := 0$;
**While (stop criteria are not met), do**:
  **Selection**: Select *selsize* solutions as the parent set $Q(t)$;
  **Modelling**: Construct a probabilistic model $p_t(\nu)$ according to $Q(t)$;
  **Sampling**: Sample *cresize* offspring from $p_t(\nu)$. Apply local search algorithm to each sampled solution;
  **Replacement**: Replace partially the current population with the offspring to constitute a new population $P(t+1)$; set $t := t+1$;

Fig. 1. Framework of the hybrid estimation of distribution algorithms.

and the routing and wavelength assignment problem under shared-risk-link-group constraints[27] arisen in the telecommunication area, and continuous optimization problems.[25,16]

The general template of HyEDA is summarized in Fig. 1. The template consists of five major components, including fitness definition, selection operation, probability model, sampling operation, and replacement operation. In the evolutionary process of the proposed first-stage algorithm for the cellular system design problem, the $Z$ value of a configuration is defined as its fitness. The truncation selection[8] is adopted. At generation $t$, $N$ solutions with the smallest $Z$ values are selected to constitute the parent set $Q(t)$ (Step 1). We use a K-means clustering to help the construction of the probability model $p_t(\nu)$ (Step 2). Details of Step 2 will be described in Sec. 3.3. The sampling operation creates new offspring using the previously created probability model (Step 3). We take a "Accept–Reject" sampling method, which shall be described in Sec. 3.4. Notably, a local search algorithm is incorporated into the template, in order to tune each solution into a local optimum after sampling both in population initialization and in offspring generation. We shall describe the local search algorithm in Sec. 3.5. To produce $P(t+1)$, the *popsize* solutions with the smallest $Z$ values are selected from the union of the sampled solutions so far and the solutions in $P(t)$.

### 3.2. *Solution representation*

Given a set of $n$ BSs' locations, the BSs' powers and antenna heights, the objective function $Z$ can be calculated by the propagation and the link budget model.[2] A solution $\nu$ of the cellular system configuration design problem can be represented as a vector $[(x_1, y_1, p_1, h_1), \ldots, (x_n, y_n, p_n, h_n)]$, where $(x_i, y_i)$ represents the location of BS $i$, $p_i$ and $h_i$ are the power and antenna height of BS $i$, respectively. $(x_i, y_i)$

take integer values, where $1 \leq x_i \leq M$ and $1 \leq y_i \leq K$, whilst $(p_i, h_i)$ take discrete integer values from set $P$ and set $H$, respectively.

Generally, $M, K \gg |P|, |H|$. The search space of the BS locations with cardinality $C^n_{M\times N}$, is much larger than that of power and antenna height $((|P| \times |H|)^n)$. For example, in the test problem with $|P| = |H| = 3, M = 13$, and $K = 20$ (used in Ref. 2), the ratio of the cardinalities of the two search spaces $C^n_{M\times N}/(|P| \times |H|)^n = C^{11}_{260}/9^{11} \approx 946305050$. Note that there are problems in which the variables may have small cardinalities, but big impacts in solving the problem. For the cellular system design problem of interest here, we argue that the BSs' locations play more important roles in achieving the good performance of the cellular system configuration than the other two parameters. Given a nonoptimized location assignment, regardless of how good the configuration of BSs' powers and antenna heights is, the overall performance of the cellular system would not be good enough. Therefore, the proposed algorithm is designed to have two stages, with much of the effort on searching for the optimal locations of the BSs. After the optimal locations have been found, a simple local search algorithm is applied for searching the optimal powers and antenna heights of these BSs in the second stage. In the first stage, the solution in the algorithms is represented as $\nu = [(x_1, y_1), \ldots, (x_n, y_n)]$, while the solution is $\nu = [(p_1, h_1), \ldots, (p_n, h_n)]$ in the second stage. To calculate the fitness value of a solution in the first stage, a set of power and antenna height settings for the BSs is randomly selected from $P$ and $H$ and applied to all solutions occurred in the evolution procedure. In the second stage, given the locations found in the first stage for the BSs, we use a simple local search algorithm to optimize $(p_i, h_i)$ with respect to function $Z$ for each BS.

### 3.3. *Probability model construction*

At generation $t$, the probability model is built with the help of a K-means clustering algorithm from the parent set $Q(t)$. The K-means clustering algorithm[5] is widely used in pattern recognition, unsupervised learning of neural network, classification analysis, clustering analysis, and so on. The algorithm classifies a set of data into several groups/clusters by minimizing the sum of squares of distances between data and the corresponding cluster centroid (center of the clustered data). We take Tchebycheff distance in HyEDA. The K-means clustering algorithm has also been used to construct a Gaussian distribution model for continuous optimization problem in Ref. 14 and 20. Here, the K-means is used to cluster a set of grid points with discrete values.

To construct $p_t(\nu)$, we first cluster the totally $N \times n$ points distributed in the service area into $n$ groups. Then we assign a probability $p_{ij}$ on each grid point $(i, j)$ according to the cluster results. The value of $p_{ij}$ indicates the probability that a BS is likely to be placed in $(i, j)$.

Suppose that the coordinates of the cluster centroid are $\{(x^*_k, y^*_k), 1 \leq k \leq n\}$ after clustering, and the corresponding standard deviation of distances between points to the centroid of cluster is $\{\sigma_k, 1 \le k \le n\}$. Generally speaking, a larger $\sigma_k$ means a bigger uncertainty to locate BS $k$ at $(x_k^*, y_k^*)$. Whereas a smaller $\sigma_k$ means a smaller uncertainty, i.e., we have more confidence to set BS $k$ at the centroid $(x_k^*, y_k^*)$. We can imagine that early on the evolution process, the $\sigma_k$'s would be relatively larger, but later on since the whole population tends to converge, the $\sigma_k$ values will converge to zero.

To set the probability value for each grid point $(i, j)$, we have to decide which cluster this grid point belongs to.[a] Let $d_l$ denotes the Tchebycheff distance between the grid point $(i, j)$ and the centroid $(x_l^*, y_l^*)$ for $1 \le l \le n$, and $k = \arg\min_l d_l$, then $p_{ij}$ is assigned as follows:

$$p_{ij} = \begin{cases} \epsilon + \mathcal{N}\left(\dfrac{d}{d_{\max}^k}; 0, \sigma_k^2\right) & \text{for } d_k \le d_{\max}^k \\ \epsilon, & \text{otherwise} \end{cases} \tag{6}$$

where $\epsilon$ is a positive value to guarantee that all grid point has a probability to be chosen; $\mathcal{N}$ is the normal probability density function (pdf); $d_k$ is the Tchebycheff distance between $(i, j)$ to its nearest centroid $(x_k, y_k)$; and $d_{\max}^k$ is the maximum Tchebycheff distance among the points in cluster $k$ to the centroid. From Eq. (6), it can be seen that if $\sigma_k$ is larger, the probabilities assigned on the grid points belonging to a certain cluster will be relatively flat to reflect our great uncertainty. If $\sigma_k$ is smaller, the probabilities will be set roughly to show our confidences on cluster $k$.

As an example, the probability assignment at a generation is shown in Fig. 2 with four BSs and different $\sigma$ values.

### 3.4. *Sampling method*

HyEDA employs the same sampling method for both offspring generation and population initialization. To sample for the initial population, the probability model $p_{ij}$ is set to $1/(M * K)$ for all $1 \le i \le M$ and $1 \le j \le K$. To sample new offspring at generation $t$, the probability model $p_{ij}$ produced in the modeling is applied.

The sampling process for a solution is as follows. To locate the $n$ grid points required for the solution, we select locations based on the probability model one by one. In each step, a location is picked from the available grid points proportionally to the probabilities associated with these grid points. For example, to pick the first location, the probability values for all the grid points can be re-arranged from $(1, 1)$ to $(M, K)$ to form a vector, we can then normalize the vector to proportionally select a location, which is the same as the famous roulette-wheel selection[8] in genetic algorithm.

[a] A grid point $g$ belongs to a cluster $k$ in the sense that the distance between $g$ and the centroid $c_k = (x_k^*, y_k^*)$ is the minimum among all distances of $g$ and $c_i, 1 \le i \le n$, i.e.,

$$k = \arg\min_i d(g, c_i),$$

where $d$ is taken as the Tchebycheff norm.

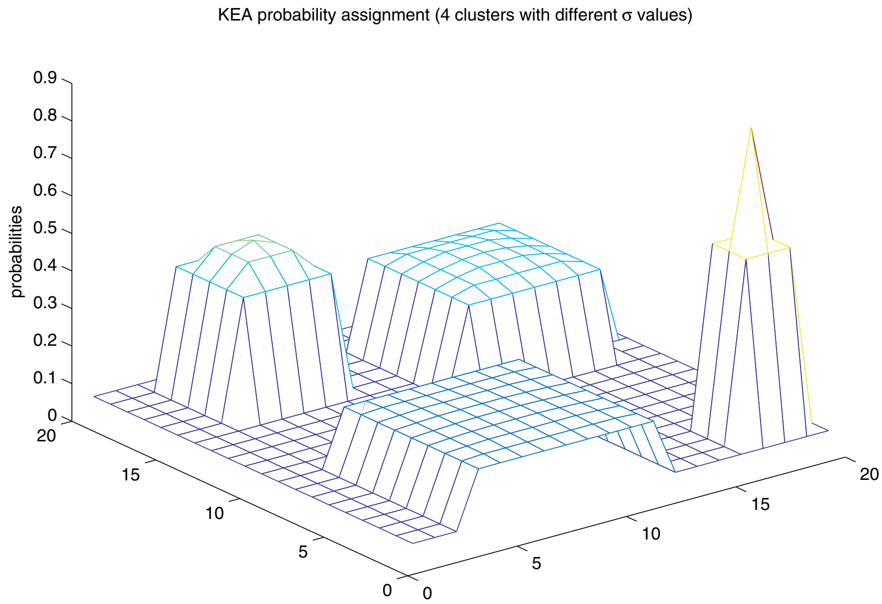


Fig. 2. The probability assignment used by the proposed probability model.

Once a grid point is picked as a BS location, its neighborhoods of grid points are not considered as the candidates for a next potential BS location any more (Here we make use of the specific knowledge of the CDMA cellular system design problem. That is, it is less likely that any two BSs in the system configuration would lie much close to each other. Since EDA can readily use this problem-specific knowldege, we adopt EDA rather than other evolutionary approaches, such as genetic algorithms). We achieve this through setting their probabilities to be zero. The neighborhood set of a grid point $g_i$ is defined as $\mathcal{D}(g_i) = \{z : d(z, g_i) \leq d_{\min}\}$, where $d_{\min}$ is a user-defined parameter and $d(z, g_i)$ is the Tchebycheff norm distance between $z$ and $g_i$. Such a process is called "Accept–Reject" process which has been used in the Monte Carlo simulation.[6] For each location picked, it is either accepted if it has enough distance between each of the picked BSs' locations, or rejected if its distance to a certain BS is less than a given distance. From the view of the probability model, the original probability model is changed after each pick of a BS location. Then the same roulette-wheel selection process can be applied again to pick a new location. The process continues until the required number of locations are all assigned.

### 3.5. *Local search*

As mentioned early, a local search algorithm has been used in the first stage, to tune each solution in initialization and after sampling process. In the second stage, HyEDA merely takes this local search method to find optimal or near-optimal values

1. Set $\nu^* := \nu = (s_1, s_2, \cdots, s_\ell)$ where $\ell = n$ for assigning BS locations, or $\ell = 2n$ for setting powers and antenna heights;
2. Repeat:
   for $i = 1 : \ell$
   1. Find the best $v' \in \mathcal{N}(s_i)$ and the corresponding $\nu' = (s_1, \cdots, s_{i-1}, v', s_{i+1}, \cdots, s_\ell)$, s.t. for any $v \in \mathcal{N}(s_i)$ and the corresponding $\nu = (s_1, \cdots, s_{i-1}, v, s_{i+1}, \cdots, s_\ell)$, $Z(\nu) \geq Z(\nu')$.
   2. Set $\nu^* := \nu'$.
3. Until no better solution is found.
4. Return $\nu^*$.

Fig. 3. Pseudo code of the local search algorithm applied in the algorithms.

of $(p, h)$ for each BS. The only difference of the local search method between the two stages lies in the definitions of neighborhood.

A solution $\nu = [(x_1, y_1), \ldots, (x_n, y_n)] = (g_1, \ldots, g_n)$ represents the BSs' locations assigned in the first stage. For each location $g_i = (x_i, y_i)$, its neighborhood $\mathcal{N}(g_i)$ is defined as the other eight grid points around $g_i$, i.e., $\mathcal{N}(g_i) = \{u : d(u, g_i) = 1.\}$. Whereas, for each $p_i(h_i)$, optimized in the second stage, its neighborhood $\mathcal{N}(p_i)$ $(\mathcal{N}(h_i))$ is defined as the other $|P| - 1$ $(|H| - 1)$ possible values.

The pseudo-code of the local search algorithm is summarized in Fig. 3. In each iteration of the local search, for each component of the solution ($g_i$, $p_i$ or $h_i$), the algorithm searches the component's neighborhood to find the best component value and replace the current component value. The search continues until no better solution can be found.

## 4. Experimental Results

### 4.1. *Test instances*

In Ref. 2, a case study for a cellular system configuration was conducted for the service area of the city of Cambridge and its vicinity in Eastern Massachusetts, and SA was applied to solve the problem. For the problem instance used in Ref. 2, the service area is divided into $13 \times 20$ grid points, with $n = 11$ BSs supposed to be located for the whole area, and the weights for the objectives set to be $w_i = 1/3$, $1 \leq i \leq 3$. We call such a problem as Problem 1.

To make the comparison between SA and HyEDA more rigourously, apart from the above Problem 1, additional four problems, which are newly derived by us based on Problem 1, have also been used to compare effectiveness of both algorithms. All five problems share the same settings of the service area and the number of BSs required to be located, but have a different scalable setting in resolutions of

Table 1. The characteristics of test problems.

| Problems | $M$ | $K$ | $\ell$(km) |
|---|---|---|---|
| 1 | 13 | 20 | 0.260 |
| 2 | 20 | 30 | 0.173 |
| 3 | 26 | 40 | 0.130 |
| 4 | 39 | 60 | 0.087 |
| 5 | 65 | 100 | 0.052 |

grid points, which determines how precisely 11 BSs can be located in the area. In Problem 2 to 5, the same area are divided into $20 \times 30$, $26 \times 40$, $39 \times 60$, and $65 \times 100$ grid points, respectively. As a result, finding optimal locations turns to be even harder progressively from Problem 1 to Problem 5, simply because of the enlarged search space of location choices.[b] The same as in Ref. 2, each grid point is in the center of a rectangular area with length $\ell$ in $km$. In the new created problems, the demand matrices and site cost matrices are randomly generated. To generate a demand matrix, its total number of demands is set to be the same as the total number of demands in the demand matrix in Ref. 2. The cardinalities of $P$ and $H$ are set to be 3 as in Ref. 2. Other parameters adopted for calculating the objective value are the same as those in Ref. 2. Characteristics of all the test problems are listed in Table 1.

### 4.2. *The effect of hybridization*

Local search is incorporated in the proposed algorithm. Can we obtain any benefit from the incorporation? To explain this, we compare the proposed algorithm with another algorithm, called sHyEDA, in which all the components are the same as the proposed, but no local search algorithm is incorporated. Table 2 shows the comparative results of the proposed algorithm and its counterpart on the test instances shown above.

In our experiment, the $d_{\min}$ values are set to 2 for the first two problems, 3 for problems 3 and 4, 4 for problem 5. The population size of the proposed algorithm is set to 10 for problems 1 and 2, 20 for the rest problems. The population size of sHyEDA is set to 100, and 200 for the rest problems. The algorithms will terminate if the algorithm cannot find a better solution in successively 10 generations after 30 generations, or a maximum 1,000,000 fitness evaluations are reached. This termination approach is adopted in the experiments discussed in the next section.

[b]To create a new test instance, one can enlarge the service area or refine the division. If the service area is enlarged, more BSs will be required, since the Walfisch/Ikegami model[13] is fit only for a cell of size 0.02–5 km, In such a case, the hardness of the optimization problem is not increased. To refine the division, one can obtain a more precise BS locations in the service area, and obviously hardness of the problem is increased. Therefore, we create new instances by using the later strategy, since it is more suitable in practice.

Table 2. Comparisons of HyEDA and sHyEDA on 5 design problems.

| Problem instance | sHyEDA | | | HyEDA | | | sig. |
|---|---|---|---|---|---|---|---|
| | Mean | Nfe. | std. | Mean | Nfe. | std. | |
| 1 | 0.3134 | 89,133 | 0.123 | 0.2301 | 59,140 | 0.031 | 0.012 |
| 2 | 0.3211 | 334,356 | 0.563 | 0.2310 | 209,053 | 0.058 | 0.000 |
| 3 | 0.5313 | 458,345 | 0.566 | 0.2415 | 380,304 | 0.075 | 0.000 |
| 4 | 0.5668 | 590,343 | 0.759 | 0.2663 | 482,288 | 0.121 | 0.000 |
| 5 | 0.6098 | 700,854 | 0.823 | 0.3058 | 529,205 | 0.098 | 0.000 |

Table 3. Comparisons of SA and HyEDA on 5 design problems.

| Problem instance | SA | | | | HyEDA | | | | sig. |
|---|---|---|---|---|---|---|---|---|---|
| | Mean | Nfe. | Time(s) | std. | Mean | Nfe. | Time(s) | std. | |
| 1 | 0.2414 | 157,893 | 150 | 0.023 | 0.2301 | 59,140 | 125 | 0.031 | 0.038 |
| 2 | 0.2441 | 451,120 | 460 | 0.359 | 0.2310 | 209,053 | 420 | 0.058 | 0.002 |
| 3 | 0.2491 | 618,278 | 800 | 0.256 | 0.2415 | 380,304 | 830 | 0.075 | 0.001 |
| 4 | 0.3167 | 1,000,000 | 1240 | 0.759 | 0.2663 | 482,288 | 1310 | 0.121 | 0.000 |
| 5 | 0.3618 | 1,000,000 | 1580 | 0.823 | 0.3058 | 529,205 | 1640 | 0.098 | 0.000 |

Both sHyEDA and HyEDA were carried out for 10 times for each problem. Listed in Table 3 are the average fitness objective value ("Mean" in the table), the average number of fitness evaluations ("Nfe" in the table), the standard deviation ("std." in the table) of the fitness objective values. One tailed $t$-test results ($p$ values under "sig." in the table) suggest that HyEDA outperforms sHyEDA at statistically significant levels for each problem in terms of the fitness objective values. Moreover, the number of fitness evaluations required by sHyEDA is larger than HyEDA for every problem. The comparison undertaken here demonstrates that the incorporation of local search leads to better solutions.

### 4.3. *The comparison of SA with the proposed algorithm*

We first briefly describe the SA used in Ref. 2. The neighborhood of a BS includes its eight directions around it, and in each direction, there are $|P|$ possible ways to select the power of the BS and $|H|$ possible ways to select the height of the antenna. The initial temperature $c$ is chosen to be

$$c_I = \Delta Z^{(+)}_{\min} + 0.1(\Delta Z^{(+)}_{\max} - \Delta Z^{(+)}_{\min}),$$

where $\Delta Z^{(+)}_{\max}$ and $\Delta Z^{(+)}_{\min}$ are the maximum and minimum of the positive changes from the initial solution to its neighborhood solutions.[2] The cooling schedule is based on search stage. A stage $k$ is finished after equilibrium has been obtained, and new temperature is lowered by a control parameter $\alpha$: $c_{k+1} = \alpha c_k$, where $\alpha = 0.9$ is suggested in Ref. 2. The equilibrium is obtained if the new configuration is always better than the current one, and the difference is less than the current temperature.

For each problem, we run both SA and HyEDA 10 times each. For each of 5 problems, the average fitness value produced by HyEDA is smaller than that achieved by SA, whilst the number of fitness evaluations required in HyEDA is less than that in SA. One tailed $t$-test results (under "sig." in the table are $p$ values $< 0.05$) confirm that HyEDA is superior to SA statistically significantly for all 5 problems in terms of solution quality. That is to say, HyEDA is more likely to find better solutions than SA while using a smaller number of fitness evaluations. However, HyEDA does incur slightly more computational time, due to extra time required to construct the probability model in solving a cellular CDMA system configuration design problem.

## 5. Conclusion

In this paper, we proposed a two-stage hybrid evolutionary approach, HyEDA, for solving the CDMA cellular system configuration design problem. In the first stage of HyEDA, a hybrid estimation of distribution algorithm was proposed, which resorts to the K-means clustering method for probability model construction, and a simple local search algorithm for solution quality improvement. HyEDA has been applied to tackle a problem given in Ref. 2, as well as some of its derived and more difficult cases. The experimental results have demonstrated that the proposed algorithm outperforms the simulated annealing approach used in Ref. 2, in terms of the quality of the solutions found and the number of function evaluations used, though at the price of long computational time.

In the future, HyEDA could be improved in several ways. First, we would like to carry out more experiments to thoroughly understand the effects of the components of HyEDA including the two-stage framework, the clustering method and the Tchebycheff distance metric, and improve HyEDA toward a single stage algorithm by incorporating the local search for the antenna' powers and heights into HEDA. Second, we will try to improve HyEDA itself for its convergence and robustness for solving other optimization problems. Third, we may enhance HyEDA by embedding multi-objective search engines,[9] thereby enabling itself to handle the inherently multi-objective optimization in the CDMA cellular system configuration design. Finally, we shall explore the principle of HyEDA further in solving similar other optimization problems, such as the terminal assignment,[23] the task assignment problem,[24] and other optimization problems raised in telecommunication area.

## References

1. K. I. Kim, *Handbook of CDMA System Design, Engineering and Optimization* (Prentice Hall PTR, 1999).
2. E. Melachrinoudis and B. Rosyidi, Optimizing the design of a CDMA cellular system configuration with multiple criteria, *Ann. Oper. Res.* **106** (2001) 307–329.

3. H. Müelenbein, T. Mahnning and A. O. Rodriguez, Schemeta, distributions, and graphical models in evolutionary optimization, *J. Heuristics* **5** (1999) 215–247.
4. P. Larrañaga and J. A. Lozano, *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation* (Kluwer Academic Publishers, 2002).
5. A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data* (Prentice Hall, 1988).
6. J. S. Liu, *Monte Carlo Strategies in Scientific Computing* (Springer, 2001).
7. D. B. Fogel and Z. Michalewicz, *How to Solve It: Modern Heuristics* (Springer, Heidelberg, 2000).
8. M. Gen and R. Cheng, *Genetic Algorithms and Engineering Optimization* (Wiley-IEEE, 2005).
9. K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms* (John Wiley & Sons, Chichester, UK, 2001).
10. C. A. C. Coello and G. B. Lamont, *Applications of Multi-Objective Evolutionary Algorithms* (World Scientific, Singapore, 2004).
11. S. Baluja, Population-based incremental learning: A method for integrating genetic search based function optimization and competitive learning, CMU-CS-94-163, Carnegie Mellon University, Pittersburgh, PA (1994).
12. H. Mühlenbein and G. Paaß, From recombination of gens to the estimation of distribution I: Binary parameters, in *Lecture Notes in Computer Science 1411: Parallel Problem Solving from Nature — PPSN IV* (1996), pp. 178–187.
13. V. K. Garg, K. Smolik and J. E. Wilkes, *Applications of CDMA in Wireless/Personal Communications* (Prentice-Hall, Upper Saddle River, NJ, 1997).
14. M. Pelikan and D. E. Goldberg, Genetic Algorithms, Clustering, and the Breaking of Symmetry, Illinois Genetic Algorithm Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL (2000).
15. J. Sun, Hybrid estimation of distribution algorithms for optimization problems, PhD thesis, University of Essex (2005).
16. J. Sun, Q. Zhang and E. P. K. Tsang, DE/EDA: A new evolutionary algorithm for global optimization, *Inf. Sci.* **169**(3) (2005) 249–262.
17. N. Krasnogor, W. Hart and J. Smith (eds.), *Recent Advances in Memetic Algorithms and Related Search Technologies* (Springer, 2003), to appear.
18. P. Larraanaga and J. A. Lozano, *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation* (Kluwer Academic Publishers, Norwell, MA, USA, 2001).
19. J. Li and S. Taiwo, Enhancing financial decision making using multi-objective financial genetic programming, to appear in *Proc. 2006 IEEE Congress on Evolutionary Computation (CEC'06), July 16–21, 2006* (Sheraton Vancouver Wall Centre, Vancouver, BC, Canada, 2006)
20. Q. Lu and X. Yao, Clustering and learning gaussian distribution for continuous optimisation, *IEEE Tran. Syst., Man, Cybern., Part C*: **35**(2) (2005) 195–204.
21. A. Molina, G. E. Athanasiadou and A. R. Nix, The automatic location of base-stations for optimised cellular coverage: A new combinatorial approach, in *Proc. IEEE VTC'99 Conf.* (1999), pp. 606–610.
22. L. Raisanen and R. M. Whitaker, Comparison and evaluation of multiple objective genetic algorithms for the antenna placement problem, *Mobile Networks Appl.* **10**(1–2) (2005) 79–88.
23. S. Salcedo-Sanz, Y. Xu and X. Yao, Hybrid meta-heuristic algorithms for task assignment in heterogeneous computing systems, *Comp. Oper. Res.* **33**(3) (2006) 820–835.

24. S. Salcedo-Sanz and X. Yao, A hybrid hopfield network — genetic algorithm approach for the terminal assignment problem, *IEEE Trans. Syst., Man Cybern.*, *Part B* **34**(6) (2004) 2343–2353.

25. Q. Zhang, J. Sun, E. P. K. Tsang and J. A. Ford, Hybrid estimation of distribution algorithm for global optimisation, *Eng. Comput.* **21**(1) (2003) 91–107.

26. Q. Zhang, J. Sun, E. P. K. Tsang and J. A. Ford, Combination of guided local search and estimation of distribution algorithm for solving quadratic assignment problem, in *Proc. Bird of a Feather Workshops, Genetic and Evolutionary Computation Conference* (2004), pp. 42–48.

27. Q. Zhang, J. Sun, G. Xiao and E. Tsang, Evolutionary algorithm refining a heuristic: A hybrid method for shared path protection in WDM networks under SRLG constraints, *IEEE Trans. Syst. Man Cybern., Part B* **37**(1) (2007) 51–61.

28. Q. Zhang, J. Sun and E. P. K. Tsang, Evolutionary algorithm with the guided mutation for the maximum clique problem, *IEEE Trans. Evolutionary Comput.* **9**(2) (2005) 192–200.

29. C. Blum and A. Roli, Metaheuristics in combinatorial optimization: Overview and conceptual comparison, *ACM Comput. Surv.* **35**(3) (2003) 268–308.

30. J. M. Peña, V. Robles, P. Larrañaga, M. S. Pérez and V. Herves, Towards a new evolutionary computation, in *Advances in Estimation of Distribution Algorithms*, J. A. Lozano, P. Larrañaga, I. Inza and E. Bengoetxea (Springer, 2006).

31. R. M. Mathar and T. Niessen, Optimum positioning of base stations for cellular radio network, *Wireless Networks* **6** (2000) 421–428.

32. C. Smith and C. Gervelis, *Celluar System Design and Optimization* (McGraw-Hill Professional Publishing, 1996).