A Useful Metaheuristic for Dynamic Channel Assignment in Mobile Cellular Systems

Deepak Kumar Singh, K. Srinivas and D. Bhagwan Das
Department of Electrical Engineering, Faculty of Engineering,
Dayalbagh Educational Institute

Abstract **—The prime objective of a Channel Assignment Problem (CAP) is to assign appropriate number of required channels to each cell in a way to achieve both efficient frequency spectrum utilization and minimization of interference effects (by satisfying a number of channel reuse constraints). Dynamic Channel Assignment (DCA) assigns the channels to the cells dynamically according to traffic demand, and hence, can provide higher capacity (or lower call blocking probability), fidelity and quality of service than the fixed assignment schemes. Channel assignment algorithms are formulated as combinatorial optimization problems and are NP-hard. Devising a DCA, that is practical, efficient, and which can generate high quality assignments, is challenging. Though Metaheuristic Search techniques like Evolutionary Algorithms, Differential Evolution, Particle Swarm Optimization prove effective in the solution of Fixed Channel Assignment (FCA) problems but they still require high computational time and therefore may be inefficient for DCA. A number of approaches have been proposed for the solution of DCA problem but the high complexity of these proposed approaches makes them unsuitable/less efficient for practical use. Therefore, this paper presents an effective and efficient Hybrid Discrete Binary Differential Evolution Algorithm (HDB-DE) for the solution of DCA Problem.**

Keywords **—Dynamic Channel Assignment, Mobile Networks, Optimization and Soft Computing.**

I. INTRODUCTION

It is important to efficiently utilize the scarce radio spectrum using a proper channel assignment scheme [1]. The aim of a channel assignment algorithm is to determine a spectrum efficient assignment of channels to the cells such that the traffic demand can be met as far as possible while satisfying the channel reuse constraints: co-channel constraints, channels separation constraints, and co-site constraints. A channel can be reused is that the same channel can be assigned to multiple cells simultaneously due to the radio propagation path loss. Channel assignment algorithms can be classified as static or dynamic. In a static approach, which is commonly called Fixed Channel Assignment (FCA) the channels are allocated and prefixed to each cell during the setup according to the traffic intensity estimated by the designer in the cell. FCA is still in use because it requires a moderate amount of BS radio installer and equipment and a simple monitoring procedure/algorithm. However, high efficiency of total channel usage over the whole service area if the traffic varies dynamically from cell to cell cannot be attained by FCA. To solve this problem, DCA [2] has been in trend since last twenty years. In DCA, channels are assigned dynamically over the cells in accordance with traffic load. FCA being a static technique can afford to spend more time to generate a better solution and is also easier to implement FCA in practice. However, from a resource utilization point of view, DCA is more preferable over FCA as DCA is designed to adjust resource assignment according to traffic demand, and hence, can support a higher capacity (or lower call blocking probability). The advantages of DCA are that it has a lower blocking rate than FCA at low traffic intensity and a lower forced call termination rate than FCA when blocking rates are equal [3].

DCA Problem is NP-hard and its effectiveness depends on its algorithm's efficiency in determining fast solution, good quality solution and its ease of implementation. Thus, devising a DCA, that is practical, efficient, and which can generate high quality assignments, is a challenging problem [4]. Metaheuristic Search techniques prove effective in the solution of DCA problem. A number of approaches [5-13,17,19,20,21,22] have been proposed for the solution of Channel Assignment Problem. The initial efforts for the solution of FCA were based on deterministic methods but as the problem is NP-hard these methods proved ineffective and inefficient for practical implementation for the next generation of mobile systems in which higher traffic demand was expected. To overcome the difficulties associated with the deterministic methods other heuristic methods such as Simulated Annealing [5], Tabu Search [6], Neural Networks [7,8] and Genetic Algorithms [9,11,12] were used for the solution of FCA problem. Later, Feedforward Neural Networks [19], Hopfield Neural Networks [20,21], Genetic Algorithms [10,13], Combinatorial Evolution Strategy [22], and Particle Swarm Optimization [17] have been used for the solution of DCA problem. However, the ever increasing number of mobile cellular users and the increasing demand for bandwidth call for more and more efficient Dynamic Channel Assignment strategies. Therefore, this paper presents an effective and efficient Hybrid Discrete Binary Differential Evolution (HDB-DE) for the solution of DCA Problem.

HDB-DE is a discrete binary version of Differential Evolution [23] which is an effective stochastic parallel search evolutionary algorithm for global optimization. The problem formulation and the implementation of HDB-DE take care of the soft constraints as well as hard constraints and hence focuses search only in the feasible regions of the search space

DOI: 10.9781/ijimai.2012.161

resulting in fast convergence. The features of HDB-DE also help in the reduction of the population size which consequently reduces the computation time, and also results in better global/ near global solution.

Section II briefly describes the constraints in CAP, assumptions pertaining to cellular model and call arrival and then the problem formulation of DCA. Section III briefly describes Hybrid Discrete Binary Differential Evolution Algorithm (HDB-DE) for the solution of DCA Problem**.** The implementation details of HDB-DE algorithm are given in Section IV. The performance of HDB-DE is evaluated on standard benchmark problems. The details of these benchmark problems, simulations and the obtained results are presented in Section V. Conclusions are presented in Section VI.

II. Problem Formulation For DCA

A. Constraints in Channel Assignment Problem

In any cellular network, whenever two cells use the same channel or when two cells use channels adjacent to each other on the spectrum or when two channels are assigned to the same cell, interference occurs; these types of interference are called Co-channel Interference, Adjacent Channel Interference and Co-site Interference respectively [14].

They lower the signal-to-noise ratio at the receiving end, leading to the deterioration of system performance. Though the computation of the actual level of interference is tough, primarily owing to its dependence on the topology of the real environment, experiments show that the effect of interference is reasonably low if the following three constraints are satisfied:

Co-Channel Constraint (CCC): The same channel cannot be simultaneously allocated to a pair of cells unless there is a minimum geographical separation between them.

Adjacent Channel Constraint (ACC): Adjacent channels cannot be assigned to a pair of cells unless there is a minimum distance between them.

Co-Site Constraint (CSC): A pair of channels can be employed in the same cell only if there is a minimum separation in frequency between them. These constraints are called Electromagnetic Compatibility Constraints, which together with the traffic demand constraint, are known as *hard constraints*.

Apart from the *hard constraints*, another set of constraints called *soft constraints* is also considered, which may be described as follows:

The *packing condition* requires that a channel, in use in one cell, should be reused in another cell as close as possible (but obviously not interfering with the former) so that the number of channels used by the network is minimal, thereby lowering the probability of future call blocking in other cells.

The *resonance condition* tries to ensure that same channels are assigned to cells belonging to the same *reuse scheme* [4], as far as possible.

Another soft constraint is that, when a call arrives, minimum number of channel reassignment operations should be performed because excessive reassignment in a cell may lead to increase in blocking probability.

A solution to the CAP must satisfy the hard constraints whereas a soft constraint may be violated; the latter only helps maximize the utilization of resources and/or improve the quality of service.

Apart from the traffic demand constraint, the only other hard constraint that we have taken into account is Co-channel Constraint; other sources of interference are assumed to be absent, as reported in literature Vidyarthi et al [15], Battiti et al [16], and Chakraborty et al [17].

B. Assumption Pertaining to Cellular Model and Call Arrival

1) The topological model is a group of hexagonal cells that form a parallelogram shape (equal number of cells along x-axis and y-axis) as shown in figure 1(adapted from [15], figure 1]).

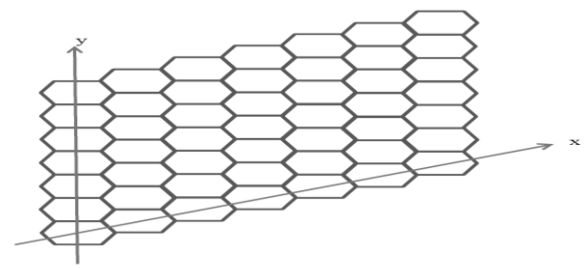

Fig.1. Cellular network model.

2) Cells are grouped in cluster of size 7 cells.
3) Each channel may serve only one call (i.e., multiplexing techniques are ignored). In DCA, all channels are put in central pool. A channel is assigned to an incoming call by a central controller that supervises the whole cellular network.
4) The selection of a channel is only subject to co-channel interference shown in figure 2. Other sources of interference are ignored.

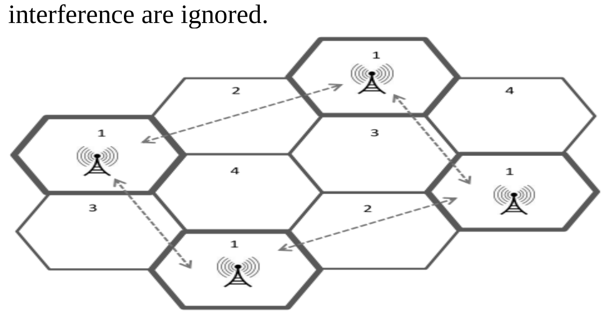

Fig.2. Co-channel interference

5) We consider not the start-up situation but the situation at a certain intermediate time-instant t when a certain number of calls is already being served by the network.
6) At time t, only one new call arrives at only one cell, called the host cell, all other conditions in the entire network remaining unaltered.
7) We set a minimum "reuse distance", which represents the minimum allowable normalized distance between two cells which may use the same channel at the same time as shown in figure 3.

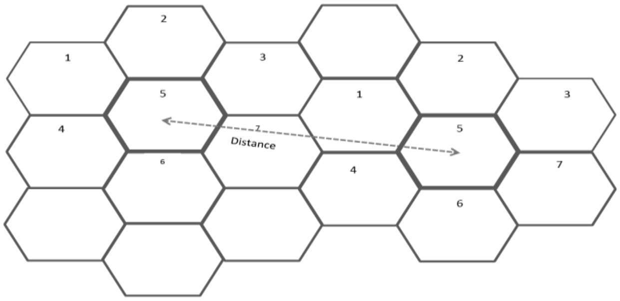

Fig.3. Frequency reuse distance.

8) A call is blocked if the entire set of channels in the network is in use in the cell involved in call arrival and its neighborhood that is there is no channel that satisfies the co-channel interference.
9) Existing calls in a cell involved in a new call arrival may be rearranged.

C. Formulation of Fitness Function

The equations 1, 2, 3 & 4 given below correspond to the different conditions i.e. co-channel interference, packing conditions, resonance conditions, and discouraging excessive rearrangement condition. These equations are combined to constitute a quadratic energy function (equation 5) whose minimization leads to an optimal solution for DCA [13].

$$f_1(X) = \sum_{j=1}^{CH} \sum_{\substack{i=1 \\ i \neq k}}^{CE} V_{k,j} * A_{i,j} * interf(i,k) \quad (1)$$

Equation (1) expresses the hard condition. Where V_k is output vector for cell k, with dimension channel (CH). $V_{k,j}=1$ if channel j is assigned to cell k, otherwise $V_{k,j}=0$. Here k signifies in which cell, call arrives. The energy function increases in case a channel j which is assigned in cell i is selected by cell k that interferes with i. It thus ensures that solutions with no interference give *better* (smaller) fitness values. $A_{i,j}$ is the ij^{th} element of the assignment table A, which is 1 if channel j is assigned to cell i, and 0 otherwise.

$$f_2(X) = \sum_{j=1}^{CH} \sum_{\substack{i=1 \\ i \neq k}}^{CE} V_{k,j} * A_{i,j} * \frac{(1 - interf(i,k))}{dist(i,k)} \quad (2)$$

This term expresses the *packing condition*. The energy decreases if channel j assigned to cell k is also selected by cell and interf (i, k) =0. Energy reduction depends on the distance between i and k. The packing condition requires that a channel, in use in one cell, should be reused in another cell as close as possible (but obviously not interfering with the former) so that the number of channels used by the network is minimal, thereby lowering the probability of future call blocking in other cells.

$$f_3(X) = \sum_{j=1}^{CH} \sum_{\substack{i=1 \\ i \neq k}}^{CE} V_{k,j} * A_{i,j} * \left(1 - res(i,k)\right) \quad (3)$$

Where res(i, k) is function whose value is 1 if cells i and k belongs to the same reuse scheme, otherwise 0. Equation (3) symbolize the *resonance condition*. The resonance condition tries to ensure that same channels are assigned to cells belonging to the same reuse scheme, as far as possible.

$$f_4(X) = \sum_{j=1}^{CH} V_{k,j} * A_{k,j} \quad (4)$$

Which subtracts 1 whenever a channel already being used by cell *k*, before the arrival of the new call, is considered in the candidate solution (i.e. in the new configuration) so that a mobile terminal being served need not change its channel too often.

Finally, the fitness function *F*(*X*) is given by,

$$F(X) = W_1 * f_1(X) - W_2 * f_2(X) + W_3 * f_3(X) - W_4 * f_4(X) \quad (5)$$

Where W_1, W_2, W_3, W_4 are *weights* that determine the importance of various terms. Clearly, $f_1(X)$ accounts for the *hard constraint* which should be given a high weightage over the other terms those are associated with the *soft constraints*. The coefficient values as used in Battiti et al [16] have been used. Where W_1=7000, W_2=1.2625, W_3=0.01, W_4= 4.17625.

III. HDB-DE

HDB-DE is a discrete binary version of Differential Evolution which is an effective stochastic parallel search evolutionary algorithm for global optimization. Unlike DE here the individuals are initialized as binary strings. HDB-DE algorithm consists of three major operations – mutation, crossover and selection, which are carried out for each member of the population (called as target vector). Mutation on each target vector of the population generates a new mutant vector uniquely associated with it. The resultant mutant vector is no longer binary because of the difference operator and the control parameter. Therefore, the discretization process from a real continuous space to a binary space is done. Then the crossover operation generates a new trial vector using the mutant vector and the target vector itself. In selection phase the fitness of the trial vector is compared with the target vector and the vector with higher fitness replaces the target vector in the population for the next iteration.

A. Pseudo-Code of HDB-DE

1) Initialize parameters *t*=0, *NP* (NP is number of individuals in population), CR and F. Where $CR \in (0,1)$ is crossover constant, $F \in (0,2)$ is mutant constant.
2) Initialize target population X^t.
3) Evaluate each individual *i* in the population using the objective function.
4) Obtain the mutant population (a mutant individual, $V_i^t = [v_{i1}^t, v_{i2}^t, ..., v_{in}^t]$ is determined such that

$$V_i^t = X_{ai}^t + F * (rand * X_{bi}^t - rand * X_{ci}^t)$$

Where X_{ai}^t, X_{bi}^t *and* X_{ci}^t are three randomly chosen individuals from the population such that $ai \neq bi \neq ci$.

5) Discretization process from a real continuous space to binary space is done according to the following equation

$$V_i^t = \begin{cases} 1\ if\ \ rand(0,1) \le sigmoid(V_i^t) \\ 0\ if\ \ rand(0,1) > sigmoid(V_i^t) \end{cases}$$

Where *sigmoid( )* is a sigmoid limiting transformation function.

6) Obtain the trial population (For each mutant individual, $V_i^t = [v_{i1}^t, v_{i2}^t, ..., v_{in}^t]$ an integer random number between 1 and *n*, i.e., $D_i \in (1,2,...,n)$, is chosen, and a trial individual, $U_i^t = [U_1^t, U_2^t, ..., U_{NP}^t]$ is generated such that:

$$U_i^t = \begin{cases} V_{ji}^t, if\ (rand(0,1) \le CR)\ or\ (j = I_{rand}) \\ X_{ji}^t, if\ (rand(0,1) > CR)\ and\ (j \ne I_{rand}) \end{cases}$$

Where I_{rand} refers to a randomly chosen dimension (*j=1,2,..,n*))

7) Evaluate trial population
8) Selection (The selection is based on the survival of the fittest among the trial population and target population such that:

$$X_i^{t+1} = \begin{cases} U_i^{t+1}, & if \quad f(U_i^{t+1}) \ge f(X_i^t) \\ X_i^t, & otherwise \end{cases}$$

9) Repeat steps 2 to 8 While Termination condition not reached.
10) Output best solution.

IV. Implementation Details of HDB-DE For DCA

The Dynamic Channel Assignment problem is specified in the literature in terms of the number of cells in the network (N_{ce}), the number of channels in the pool (N_{ch}) and a *demand vector* ***D*** which is a vector whose i^{th} element denotes the traffic demand in cell *i*, *i*= 1,2,…, N_{ce}.

We assume that the new call demand is placed at cell *k* which is already serving demand (*k*) calls where demand (*k*) denotes the total traffic load (ongoing) in cell *k* at time *t*; and no ongoing call is terminated in the entire network. Our problem is to assign an available channel to the incoming call with possible reassignment of channels to the calls in progress in cell *k*.

A. Solution Representation

The candidate solution to the problem is represented as a binary string X which is the representation of V_k mentioned earlier where k signifies the cell in which call arrives. The size of the vector X is equal to the number of available channels (N_{ch}) and $X_{k,j}$=1 if channel j is assigned to cell k, otherwise $X_{k,j}$=0. The number of 1s in each solution vector is equal to demand(*k*)+1 i.e. the total ongoing calls plus the call that arrives at the concerned time instant.

B. Mutation

A new mutation operator has been designed for HDB-DE which is more effective than the one which has been used earlier [17] for mutation of solutions with binary representation in DE. The new mutation operator is as given below:

$$V_i^t = X_{ai}^t + F * (rand * X_{bi}^t - rand * X_{ci}^t)$$

The effectiveness of the operator can be seen from the fact that if both X_{bi}^t and X_{ci}^t assume a value 1 then in the earlier mutation operation the difference becomes 0 and therefore does not result in any change in the value of X_{ai}^t whereas in the newly designed mutation operation it does not happen so and leads to the generation of better mutants and thereby faster convergence.

V. Computational Results

HDB-DE algorithm was implemented in Matlab and the following benchmark problems were used for its evaluation:

1) The first benchmark problem CSys 49 [13] consists of 49 hexagonal cells that are arranged to form a parallelogram structure with 70 channels available to the system.
2) The second and third benchmark problems i.e. HEX 1 and HEX 3 [18] are based on a 21-cell system.
3) The last four benchmark problems i.e. EX 1, EX 2, KUNZ 1 and KUNZ 2 [18] are based on a 4, 5, 10 and 15-cell system respectively.

The details of the benchmark problems and the used demand vectors are summarized in Table I.

TABLE I.
Details of Benchmark Problems

Problem	N_{ce}	N_{ch}	Demand Vector (demand)
CSys 49	49	70	4,2,6,4,4,6,2,4,2,4,2,2,4,4,4,4,2,2,6,2,6,6,2,4,8,2,2,2,4, 2,2,4,2,4,6,2,4,6,4,2,6,4,4,2,2,6,4,4,4
HEX 1	21	37	2,6,2,2,2,4,4,13,19,7,4,4,7,4,9,14,7,2,2,4,2
HEX 3	21	21	1,1,1,2,3,6,7,6,10,10,11,5,7,6,4,4,7,5,5,5,6
EX 1	4	11	1,1,1,2
EX 2	5	17	2,2,2,3,3
KUNZ 1	10	30	10,11,9,5,9,4,4,7,4,8
KUNZ 2	15	44	10,11,9,5,9,4,4,7,4,8,8,9,10,7,7

For each of the considered problems it has been assumed that all cells are arranged in the form of a parallelogram, the given N_{ce} of each problem is expressed in the form $r \times c$, where *r*, *c* are integers, and hence determine the configuration of the cellular network by setting the number of rows to *r* and the number of all columns to *c* [17]. A cell is arbitrarily selected and then it is assumed that, just before a call demand arrives in this cell at time *t*, *demand(i)* calls were already in progress in the i^{th} cell, *i* =1 ,2,…, N_{ce}, and *demand(k)* calls are ongoing in the k^{th} cell. Accordingly, N_{ce} X N_{ch} assignment matrix, avoiding co-channel interference has been manually determined which describes the status of ongoing calls in each cell before the new call arrival and thus represents the initial condition. The assignment table used for HEX3 problem is given in Table II.

Whenever a simulation results in a solution, which violates *Co-Channel Constraint* (CCC) the call is rejected. C*all Rejection Probability* (CRP) as given in [17] is used as a parameter for determining the effectiveness of the proposed method.

$$CRP = N_{rejected} / N_{total}$$

Where $N_{rejected}$ = number of simulations in which the incoming call is rejected in the host cell considered; N_{total} = total number of simulations.

TABLE II.
INITIAL ASSIGNMENT MATRIX FOR HEX3

Problem	Initial Assignment Matrix
HEX3	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
	0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
	0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
	1 0 1 0 0 0 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0
	0 1 0 0 0 1 0 1 0 0 0 1 0 0 1 0 0 0 0 0 0
	0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 1 0 1 0 0 0
	1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 0 0 0 0
	0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 0 0 0
	1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 0 0 0 0
	0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 1 0 1 0
	0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 0 0 0 0 0
	1 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	0 0 1 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0
	0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 1 0
	0 1 0 1 0 0 0 0 0 1 0 0 1 0 1 0 0 0 0 0 0
	0 0 1 0 1 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0
	1 0 0 0 0 1 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0
	0 0 0 1 0 0 1 0 0 0 0 0 0 1 0 0 0 1 0 0 0
	0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0

Thus, CRP is the cumulative proportion of simulations, for which the call is rejected, in the long run. This parameter CRP is based on but different from the *call blocking probability* used in the [13,14]. The former characterizes a particular cell under a given initial condition while the latter characterizes the cellular network as a whole.

TABLE III.
SIMULATION RESULTS

Problem	Host Cell	Number of simulations	Number of times, call is rejected				CRP			
			Method							
			PSO [17]	SGA	MGA	HDBDE	PSO [17]	SGA	MGA	HDBDE
CSys 49	6	250	-	14	0	**0**	-	0.06	0	**0**
HEX1	6	104	21	8	0	**0**	0.20	0.08	0	**0**
HEX3	5	105	32	12	0	**0**	0.31	0.11	0	**0**
EX1	4	112	32	6	0	**0**	0.28	0.05	0	**0**
EX2	4	100	28	9	0	**0**	0.28	0.09	0	**0**
KUNZ1	7	110	39	11	0	**0**	0.35	0.10	0	**0**
KUNZ2	7	100	20	11	0	**0**	0.20	0.11	0	**0**

The parameters used in HDB-DE are as follows:
Population size=30,
Mutation factor F=1.2,
Crossover constant CR= 0.6 and
Maximum number of generations=50.

The simulation results obtained by HDB-DE and those obtained by Simple Genetic Algorithm (SGA), Modified Genetic Algorithm (MGA) [24] and PSO [17] for the different benchmark problems are shown in Table III.

The results shown in table III indicate that the performance of HDB-DE and MGA is good compared to PSO. The comparison of the convergence curves and the average number of Evaluation expended to yield the best solution over a good number of runs of the algorithms will further throw light on the efficiency, efficacy and consistency of the algorithms.

Figure 4, 5, 6, 7, 8, 9, and 10 shows the convergence curve of HDB-DE, MGA and SGA for the seven benchmark problems CSys 49, HEX 1, HEX 3, EX 1, EX 2, KUNZ 1 and KUNZ 2 respectively.

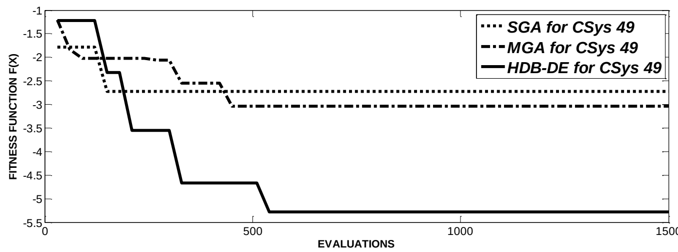

Fig.4. Convergence curve for CSys 49 consists of 49 hexagonal cells with 70 channels.

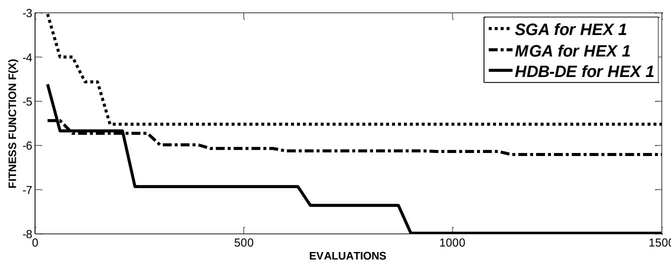

Fig.5. Convergence curve for HEX 1 consists of 21 hexagonal cells with 37 channels

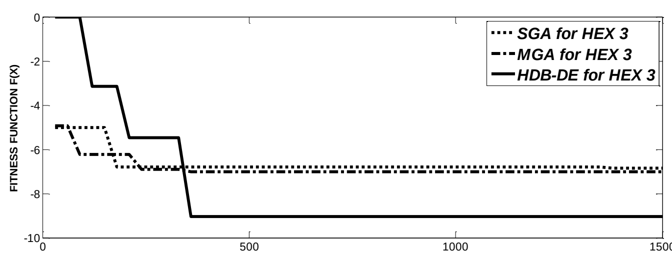

Fig.6. Convergence curve for HEX 3 consists of 21 hexagonal cells with 21 channels.

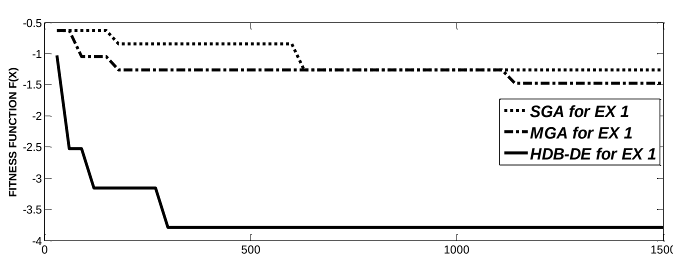

Fig.7. Convergence curve for EX 1 consists of 4 hexagonal cells with 11 channels.

TABLE IV.
SIMULATION RESULTS OBTAINED BY HDB-DE AND GA FOR ALL BENCHMARK PROBLEMS

Benchmark	**CSys 49**			**HEX 1**			**HEX 3**			**EX 1**			**EX 2**			**KUNZ 1**			**KUNZ 2**		
Algorithm	*HDBDE*	*MGA [24]*	*SGA*	*HDBDE*	*MGA [24]*	*SGA*	*HDBDE*	*MGA [24]*	*SGA*	*HDBDE*	*MGA [24]*	*SGA*	*HDBDE*	*MGA [24]*	*SGA*	*HDBDE*	*MGA [24]*	*SGA*	*HDBDE*	*MGA [24]*	*SGA*
Best # of Evaluations	**90**	200	450	**60**	150	380	**60**	100	250	**28**	65	210	**60**	85	250	**60**	160	230	**90**	150	230
Worst # of Evaluations	**300**	410	900	**170**	360	750	**130**	270	600	**65**	130	410	**95**	240	550	**160**	410	550	**180**	350	650
Average # of Fitness Evaluations	**163**	301	681	**98**	249	512	**93**	180	416	**47**	97	312	**79**	156	394	**99**	255	368	**123**	236	387

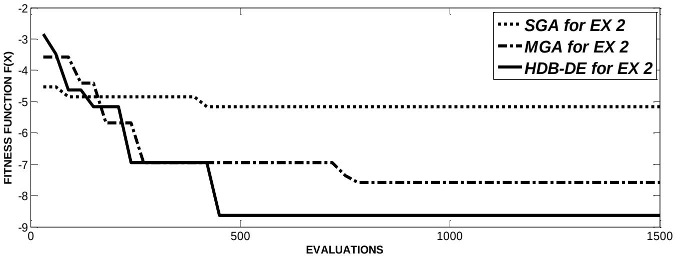

Fig.8. Convergence curve for EX 2 consists of 5 hexagonal cells with 17 channels.

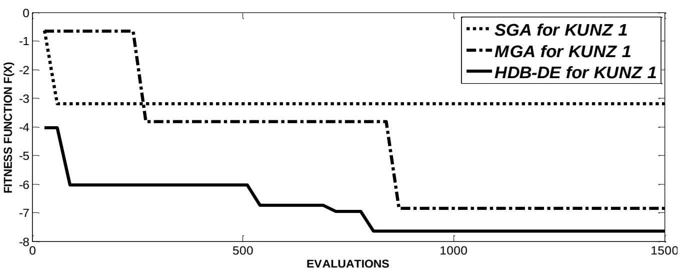

Fig.9. Convergence curve for KUNZ 1 consists of 10 hexagonal cells with 30 channels.

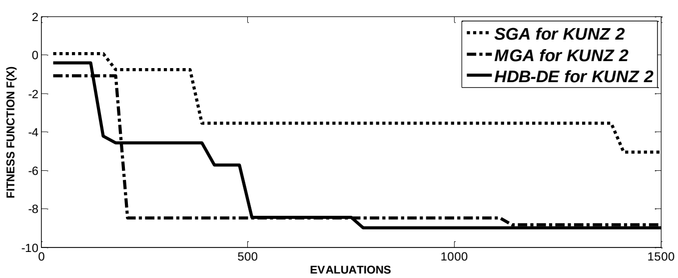

Fig.10. Convergence curve for KUNZ 2 consists of 15 hexagonal cells with 44 channels.

Table IV presents the results in terms of Best number of evaluations, Worst Number of evaluations and average number of evaluations expended by HDB-DE, MGA and SGA to give the final best solution in 30 independent runs on each benchmark problem.

The convergence curves shown in figures 4 to 10, and the results presented in Tables III and IV clearly indicate the superior and consistent performance of HDB-DE over the other methods reported in the literature in terms of quality of solutions and speed of convergence. The results are promising and indicate the suitability of HDB-DE for the solution of even bigger instances of real-world DCA problem.

VI. CONCLUSION

This paper presents an effective and efficient Hybrid Discrete Binary Differential Evolution (HDB-DE) Algorithm for the solution of Dynamic Channel Assignment Problem in cellular radio networks. The features of HDB-DE help in achieving fast convergence and good solution. The problem formulation and the implementation of HDB-DE for DCA takes care of the soft constraints as well as hard constraints and hence focuses search only in the feasible regions of the search space resulting in fast convergence. HDB-DE algorithm has been evaluated on benchmark problems for DCA. The obtained results and their comparison with those obtained by the other methods establish the superiority of HDB-DE over other methods and indicates its suitability for the solution of even bigger instances of dynamic channel assignment in cellular networks resulting in lower call rejection probability, higher capacity utilization and a good quality of service.

REFERENCES

[1] A. Mehrotra, Cellular Radio. Analog and Digital Systems. Norwood, MA: Artech House, 1994.

[2] W. C. Jakes, Microwave Mobile Communication. New York: Wiley, 1974.

[3] W. C. Y. Lee, Mobile Cellular Telecommunications Systems. New York: McGraw-Hill, 1989.

[4] R. C. V Macario, Cellular Radio: Principles and Design, 2nd ed. New York: Macmillan, 1997.

[5] M. Duque-Anton, D. Kunz, and B. Ruber, "Channel assignment for cellular radio using simulated annealing". IEEE Trans. Vehicular Technology., vol. 42, no. 1, pp. 14–21, Feb. 1993

[6] D. J. Castelino, S. Hurley, and N. M. Stephens, "A tabu search algorithm for frequency assignment". Ann. Oper. Research., vol. 63, pp. 301–319. Sept. 19, 2005.

[7] D. Kunz, "Channel assignment for cellular radio using neural networks". IEEE Trans. Vehicular Technology., vol. 40, no. 1 ,pp. 188–193. Feb. 1991

[8] N. Funabiki and Y. Takefuji, "A neural network parallel algorithm for channel assignment problems in cellular radio networks".IEEE Trans. Vehicular Technology., vol. 41,no. 4,pp. 430–437. Nov. 1992

[9] M. A. C. Lima , A. F. R. Araujo and A. C. Cesar, "Dynamic channel assignment in mobile communications based on genetic algorithms", In 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications., no. 5, pp. 2204-2208. Sep. 2002.

[10] M. Cuppini , "A genetic algorithm for channel assignment problems", European Trans. Telecom. Rel. Technology., vol. 5, pp. 285–294, Mar 1994.

[11] K. Lai and G. G. Coghill , "Channel assignment through evolutionary optimization", IEEE Trans. Vehicular Technology., vol. 45, no. 1,pp. 91–96. Feb. 1996.

[12] R. Dorne, and J. K. Hao, "Constraint handling in evolutionary search: A case study of the frequency assignment, problem solving from nature—PPSN IV.Voigt, H. M., Ebeling, Rechenberg, W., I., and Schwefel, H.-P., Springer, vol. 1141, pp. 801–810, 1996.

[13] H. G. Sandalidis, P. P. Stavroulakis, and J. Rodriguez-Tellez, "Implementation of genetic algorithms to a channel assignment problem in cellular communications", In Proceedings of the 6th Int. Conf. Advances in Communications and Control (COMCON 6), pp. 453–460. Jun. 1997.

[14] K. Okada and F. Kubota, "On dynamic channel assignment in cellular mobile radio systems", IEEE International Symposium on Circuits and Systems. vol. 2, pp. 938-941. June 11-14, 1991.

[15] G. Vidyarthi, A. Ngom, and I. Stojmenovic´, "A hybrid channel assignment approach using an efficient evolutionary strategy in wireless mobile networks", IEEE Trans. Vehicular Technology, vol. 54, no. 5 , pp. 1887–1895. Sept. 2005.

[16] R. Battiti, A. A. Bertossi, and M. Brunato, "Cellular channel assignment: A new localized and distributed strategy, mobile networks and applications", Kluwer., no. 6, pp. 493–500. Jan 10, 1998.

[17] M. Chakraborty, R. Chowdhury, J. Basu, R. Janarthanan and A. Konar, "A particle swarm optimization-based approach towards the Solution of the dynamic channel assignment problem in mobile cellular networks", IEEE Region 10 Conference (Hyderabad, India,), pp. 1-6, Nov. 19-21, 2008.

[18] K. Smith, and M. Palaniswami, "Static and dynamic channel assignment using neural network".IEEE Journal on Selected Area in Communications, vol. 15, no. 2, pp. 238-249, Feb. 1997

[19] P. T. H. Chan, M. Palaniswami, and D. Everitt, "Neural network-based dynamic channel assignment for cellular mobile communication systems," IEEE Trans. Vehicular Technology, vol. 43, pp. 279–288. May 1994

[20] K. Nakano, M. Yokono, M. Sengoku, Y. Yamaguchi, S. Shinoda, S. Motooka and T. Abe, "An application of dynamic channel assignment to a part of a service area of a cellular mobile communication system," IEICE Trans. Fundamentals, no. 75, pp. 369–379. Mar. 1992

[21] E. Del Re, R. Fantacci, and L. Ronga, "A dynamic channel allocation technique based on hopfield neural networks," IEEE Trans. Vehicular Technology, vol. 45, pp. 26–32. Feb. 1996

[22] H. G. Sandalidis, P. P. Stavroulakis, and J. Rodriguez-Tellez, "An efficient evolutionary algorithm for channel resource management in cellular mobile systems". IEEE Trans Evolutionary Computation, vol. 2, no. 4, pp. 125–137, Nov. 1998.

[23] C. Peng, L. Jian and L. Zhiming. "Solving 0-1 knapsack problems by a discrete binary version of differential evolution", in Proc. Second International Symposium on Intelligent Information Technology Application, vol. 2, pp. 513-516, Dec. 20-22, 2008.

[24] D. K. Singh, K. Srinivas and D. B. Das, "A dynamic channel Assignment in GSM telecommunication network using modified genetic algorithm", in proc. 6th Euro-American Conference on Telematics and Informat5ion System, IEEE, pp. no. 1-5, May 23-25, 2012.

Deepak Kumar Singh received his B. Tech. (Electronics & Communication) from G.L.A.I.T.M, Mathura, India in 2005 and M. Tech. (Engineering Systems) from Dayalbagh Educational Institute, Agra, India in 2008. Presently, he is pursuing his Ph.D. from Department of Electrical Engineering, Dayalbagh Educational Institute, Agra, India. His research area includes Mobile Telecommunication Systems, Applications of Soft Computing techniques in Mobile Telecommunication Systems, and Optimization techniques etc.

K. Srinivas received B.E. in Computer Science & Technology, M.Tech. in Engineering Systems and Ph.D. in Evolutionary Algorithms. He is currently working as Assistant Professor in Electrical Engineering Department, Faculty of Engineering, Dayalbagh Educational Institute, Agra, India. He teaches Information Technology, Computer Networks, Software Engineering, Software Testing, Systems Optimization Using Evolutionary Algorithms, Java, Network Programming and Internet Technologies. His research interests include Soft Computing, Optimization using Metaheuristic Search Techniques, Search Based Software Engineering, Mobile Telecommunication Networks, Systems Engineering, and E-learning Technologies. He is an active researcher, guiding research, published papers in journals of national and international repute, and is also involved in R&D projects. He is a member of IEEE and a life member of Systems Society of India (SSI). He is also the Publication Manager of Literary Paritantra (Systems)- An International Journal on Literature and Theory.

D. Bhagwan Das obtained B.Sc.(Engineering) in Electrical from DEI, Dayalbagh in 1990, M.Tech. (Power Systems) from IIT, Delhi in 1992 and Ph.D. in Electrical Engineering (Soft Computing Applications in Power System Optimization) from DEI, Dayalbagh in 1999. He is presently working as Associate Professor in the Department of Electrical Engineering in DEI, Dayalbagh. His research interests include Embedded Systems, FPGAs/SOC applications in designing various controllers for Solar Electric Power Plants, Power System Protection, Wide Area Measurement Systems, hardware realization of Fuzzy Controllers, ANN and Evolutionary Algorithms on FPGA chip for real life, real-time problems, application of Evolutionary Algorithms for Power System Optimization, Modeling, Simulation and optimization of VCR Diesel Engines for different blends of Bio-diesel. He is involved in the design and commissioning of 515kWp Distributed Solar Electric Power Plants in Dayalbagh Educational Institute. He is an active researcher and carried out several Research and development projects funded by AICTE, UGC and MHRD. He also has a collaborative project with BHEL, Bangalore on the design of a comprehensive control strategy and its implementation on FPGA based system for Solar Electric Power Plants. He has also designed and developed a solar powered Van. He has received AK Sinha Memorial Prize for securing highest CGPA in the Department of Electrical Engineering, IIT, Delhi in 1992, ISCA Young Scientist Award in Engineering Sciences in 2002 and several Best Paper Awards in Conferences. He has widely published in a number of International and National Journals.