

# An Evolutionary Approach to Designing Complex Spreading Codes for DS-CDMA

Balasubramaniam Natarajan, *Member, IEEE*, Sanjoy Das, and Daniel Stevens, *Member, IEEE*

**Abstract**—This paper proposes a novel evolutionary approach to spreading code design in direct sequence code division multiple access (DS-CDMA). Specifically, a multiobjective evolutionary algorithm (EA) is used to generate complex spreading sequences that are optimized with respect to the average mean-square cross-and/or autocorrelation (CC and/or AC) properties. A theoretical model is developed in order to demonstrate the optimality of the generated codes. The proposed algorithm enables spreading code design with no constraints on the code length. Furthermore, it is possible to generate  $K \geq N$  codes of length  $N$  with very little cost in correlation properties. This results in significant capacity enhancement in DS-CDMA systems.

**Index Terms**—Direct sequence code division multiple access (DS-CDMA), evolutionary algorithm, mean-square correlation, multiobjective optimization, Pareto front, spreading sequences.

## I. INTRODUCTION

ONE of the most significant limitations in multiple access communication over a finite bandwidth channel is the interference introduced by competing users of the channel. This multiple access interference (MAI) reduces the practical channel capacity and leads to reduced performance as a result of increased bit-error rate (BER) for individual users. In an asynchronous direct sequence code division multiple access (DS-CDMA) system, the MAI is combated by employing spreading codes with good cross correlation (CC) properties [1]. The autocorrelation (AC) properties of a set of codes are important for spectrum spreading as well as for initial timing synchronization and tracking [1]. It has been noted in many publications that the AC properties of a set of spreading codes come at the expense of CC properties and vice versa [2].

Spreading code design has been a widely studied topic over many years ([3]–[5]). But the focus was primarily on real codes for DS-CDMA systems, while the code design work on complex codes was focused primarily on periodic correlation properties [6], [7]. Also, prior works on code design provide binary or complex spreading codes that have restrictions on the code length  $N$  (i.e.,  $N$  is limited to  $2n$  or  $2n \pm 1$ , where  $n$  is an integer, or  $N$  is prime). Furthermore, most of these methods provided a limited set of  $K$  codes, e.g.,  $K = N$  for Hadamard–Walsh (HW) codes [8];  $K = N + 2$  for Gold codes [8];  $K = N - 1$  for Oppermann codes [2].

Manuscript received December 19, 2003; revised April 26, 2004 and July 15, 2004; accepted September 27, 2004. The editor coordinating the review of this paper and approving it for publication is V. Lau.

The authors are with the Department of Electrical and Computer Engineering, Kansas State University, Manhattan, KS 66506-5204 USA (e-mail: bala@ksu.edu; sdas@ksu.edu; dan@ksu.edu).

Digital Object Identifier 10.1109/TWC.2005.853826

In this paper, we employ a multiobjective evolutionary approach to design complex spreading codes of any length  $N$  as well as to generate  $K > N$  codes. Multiobjective evolutionary algorithms (EAs) allow the search for solutions that satisfy a set of simultaneous objectives. We believe that a meaningful objective in DS-CDMA code design is one that considers the average BER of all users rather than the worst case BER associated with the peak interference values between individual users. The average BER in a DS-CDMA system is dependent on the average interference produced by the mean-square aperiodic CC (as pointed out in [2]). Similarly, the average mean-square aperiodic AC that combines the odd and even periodic correlations is a much more reasonable measure for the AC properties of a code set. Instead of defining complex spreading codes and evaluating the CC and AC properties (as in [2]), we use the multiobjective EA to search the phase space for a set of complex phase codes that simultaneously attempt to satisfy the objectives of minimizing the average mean-square CC and AC.

The proposed approach yields a large number of spreading code sets with a wide range of correlation properties. We compare the average mean-square CC and AC properties of the EA-based code sets with other code sets such as HW, Gold, EOE-Gold, Frank–Zadoff–Chu (FZC), Oppermann, and the recently introduced carrier interferometry (CI) codes [9]. The EA-based codes demonstrate better CC and AC properties relative to these code sets while offering flexibility in code length. Furthermore, we demonstrate that the EA-based code sets can be expanded to provide up to 100% increases in capacity with little cost in correlation properties.

## II. CORRELATION MEASURE

Pursley [10] showed that the average signal-to-noise ratio (SNR) for the  $y^{th}$  user in an asynchronous DS-CDMA employing binary phase-shift keying (BPSK) is a function of the average interference parameter (AIP) and the additive noise power. Specifically, the SNR degradation of the  $y^{th}$  user strongly depends on the squared absolute value of the aperiodic CC of the  $y^{th}$  user's code with every other user's code in the system.

Therefore, in this paper, the average mean-square value of the aperiodic CC is used as the measure of code set performance (as in [2]). This measure corresponds to

$$R_{CC} = \frac{1}{K(K-1)} \frac{1}{N^2} \sum_{x=1}^K \sum_{y=1, y \neq x}^K \sum_{l=1-N}^{N-1} |C_{xy}(l)|^2 \quad (1)$$

where  $K$  is the number of codes in the code set and  $C_{xy}$  is the aperiodic correlation function between two codes in the set defined as

$$C_{xy}(l) = \begin{cases} \sum_{k=0}^{N-1-l} u_x(k)u_y^*(k+l), & 0 \leq l \leq N-1 \\ \sum_{k=0}^{N-1+l} u_x(k-l)u_y^*(k), & 1-N \leq l \leq 0 \\ 0, & \text{elsewhere} \end{cases} \quad (2)$$

where  $u_x$  and  $u_y$  are the spreading codes of user  $x$  and  $y$ , respectively. In (1),  $R_{CC}$  has been normalized by the code length in order to facilitate comparisons between codes of different lengths. A similar mean-square measure for  $R_{AC}$  is given by

$$R_{AC} = \frac{1}{K} \frac{1}{N^2} \sum_{x=1}^K \sum_{l=1-N, l \neq 0}^{N-1} |C_{xx}(l)|^2. \quad (3)$$

The relationship between  $R_{CC}$  and  $R_{AC}$  is given by the following theorem.

*Theorem 1:* For any set of  $K$  codes of length  $N$  satisfying  $C_{uu}(0) = N$  for all  $u$ 's in the code set,  $R_{CC}(K-1) + R_{AC} > (K-1)$ .

*Proof:* See Appendix.  $\blacksquare$

Ideally, it is desirable to minimize both  $R_{CC}$  and  $R_{AC}$ . In reality, the  $R_{CC}$  and  $R_{AC}$  measures have to be traded off for each other (as demonstrated by Theorem 1). It is observed that the spectra of complex codes with very low  $R_{CC}$  are narrowband, and using these code sets in DS-CDMA results in a system that has frequency division multiple access (FDMA) characteristics (see [2]). Code sets with very low  $R_{AC}$  are wideband but possess high CC values that significantly degrade the BER performance. Therefore, it is desirable to avoid the extreme values of  $R_{CC}$  and  $R_{AC}$  in the DS-CDMA spreading code design. The multiobjective EA-based approach discussed next offers the freedom to do that.

### III. MULTIOBJECTIVE EA

EAs have emerged as one of the most popular approaches for the complex optimization problems in engineering [11]. They draw upon Darwinian paradigms of evolution to search through the solution space (the set of all possible solutions). Each solution, which in this case is a set of codes, is represented as a chromosome and comprises of smaller units called genes. Therefore, we shall use the terms solutions and chromosomes interchangeably. Starting with a set (or population) of chromosomes, in each generation of the algorithm, new populations are created from older ones. The creation of new chromosomes is accomplished by means of two operators, recombination and mutation. Mutation is carried out by imparting a small usually random perturbation to the chromosome. Early experiments with code design problems suggest that recombination, the other standard operator, does not improve the performance of an EA. Hence, in this paper, we have excluded that from our discussion. In a manner similar to the Darwinian paradigm of survival of the fittest, only "good" solutions are allowed to remain in a population with the degree of optimality ("goodness") being assessed through a measure called fitness.

When dealing with optimization problems with multiple objectives, the conventional concept of optimality does not hold good [12]. The simplistic approach of aggregating multiple objectives into a single one often fails to produce good results. It produces only a single solution. In multiobjective optimization, the concepts of dominance and Pareto-optimality are applied and an entire set of solutions along the Pareto front is extracted from the solution space. The concept of Pareto-optimality is briefly explained below.

Assume that the optimization problem involves the minimization of  $M$  objectives  $c_n(\cdot)$ ,  $n = 1, \dots, M$ . A solution  $p$  is said to dominate over another solution  $q$  if and only if  $c_n(p) < c_n(q)$ ,  $\forall n \in \{1, \dots, M\}$ , i.e., for every objective,  $p$  is better than  $q$ . This relationship is represented as  $p \succ q$ . In a population of solutions, the set of all nondominating solutions is called the Pareto front. In other words, if  $\mathbf{P}$  is the population, the Pareto front  $\Gamma$  corresponds to

$$\Gamma = \{p \in \mathbf{P} | \forall q \in \mathbf{P}, q \not\succ p\}. \quad (4)$$

In recent years, many EAs for multiobjective optimization have been proposed [13], [14].

In the proposed approach, each chromosome of the population is a set of  $K$  codes each of length  $N$ . The phase angles of the spreading codes are arranged as a  $K$  by  $N$  matrix,  $\Phi = [\phi_x]$ ,  $x = 1, \dots, K$ , where  $\phi_x$  represents the angles corresponding to row  $x$  and is equal to  $[\phi_x(1), \phi_x(2), \dots, \phi_x(N)]$ . Every element of  $\phi_x$  takes on a value between 0 and  $2\pi$ , i.e.,  $\phi_x(i) \in [0, 2\pi]$ ,  $i = 1, \dots, N$ . The actual spreading code corresponding to row  $x$  is  $[e^{j\phi_x(1)}, e^{j\phi_x(2)}, \dots, e^{j\phi_x(N)}]$ . Each code (row of the matrix) is considered to be a single gene.

The size of the population is kept constant at a value  $P$ . The fitness of each code set is a two-dimensional (2-D) vector whose elements are  $R_{AC}$  and  $R_{CC}$ . The fitness of the  $p^{th}$  chromosome is normalized and scaled appropriately

$$\mathbf{F}^{(p)} = \left[ \frac{\max(R_{AC}^{(q)}) - R_{AC}^{(p)}}{\max(R_{AC}^{(q)}) - \min(R_{AC}^{(p)})} \times \frac{\max(R_{CC}^{(q)}) - R_{CC}^{(p)}}{\max(R_{CC}^{(q)}) - \min(R_{CC}^{(p)})} \right]. \quad (5)$$

Here,  $R_{AC}^{(p)}$  and  $R_{CC}^{(p)}$  are the AC and CC of the  $p^{th}$  chromosome and the  $\max(\cdot)$  and  $\min(\cdot)$  operations are carried out over all chromosomes in the population. This linear transformation was done so that each individual has a fitness  $\mathbf{F}$  whose entries lie between zero and unity, with the worst correlation getting assigned a zero fitness and the best, a fitness of unity, in the corresponding place.

Maintaining a uniform spatial distribution of chromosomes in the evolving population has been a core issue in multiobjective evolutionary optimization. Most current approaches tend to explicitly enforce population diversity by biasing the search towards those regions in the Pareto front where the chromosomes are sparsely populated. One of the common

techniques to do so is to reduce the fitness values of the individual chromosomes in such a manner that if the population contains chromosomes that are very closely spaced together, their fitness values are reduced appropriately. In this manner, chromosomes occupying isolated regions in the search space have a better chance of getting selected by the algorithm for the next generation. This technique is called fitness sharing [12]. Fitness sharing is introduced into the algorithm by scaling down the raw fitness values according to the equation

$$\mathbf{F}^{(p)} = \frac{\mathbf{F}^{(p)}}{\sum_{\forall q} \text{sh}^{(p,q)}} \quad (6)$$

where  $p$  and  $q$  are two chromosomes and  $\text{sh}^{(p,q)}$  corresponds to the sharing function between them. The sharing function is defined as

$$\text{sh}^{(p,q)} = \begin{cases} 1 - \left[ \frac{\|\mathbf{F}^{(p)} - \mathbf{F}^{(q)}\|}{\sigma_{\text{sh}}} \right]^{\kappa}, & \text{when } \|\mathbf{F}^{(p)} - \mathbf{F}^{(q)}\| < \sigma_{\text{sh}}. \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

In the above equation,  $\|\mathbf{F}^{(p)} - \mathbf{F}^{(q)}\|$  is the Euclidean distance between the raw fitnesses of two chromosomes  $p$  and  $q$ ;  $\kappa$  and  $\sigma_{\text{sh}}$  are parameters associated with fitness sharing [12].  $\sigma_{\text{sh}}$  is typically a small-valued parameter. Unless there are other chromosomes present whose fitnesses are within a distance  $\sigma_{\text{sh}}$  of any given chromosome, the summation in the denominator of (6) adds up to unity for the latter since  $\|\mathbf{F}^{(p)} - \mathbf{F}^{(p)}\| = 0$ . Hence, its scaled fitness becomes identical to the raw fitness. Otherwise, for each chromosome whose fitness is within the same distance from the given chromosome, the sharing function evaluates to a value higher than unity. The closer the fitness values are, the higher the sharing function becomes. Therefore, the fitness values of chromosomes having others with similar values in its vicinity get lowered.

In every generation of the proposed approach, a new population is created from an existing population in the following manner. First, the Pareto front of the existing population of code sets is determined and reinserted into the offspring population. This helps preserve the best solutions in each iteration. When the total number of nondominated solutions is less than a quantity  $\alpha_{\text{max}}$ , the entire front of size  $\alpha$  is included in the subsequent population. However, when the Pareto front is large, a total of  $\alpha_{\text{max}}$  solutions, picked at random, is used. Early experiments suggested that randomly picking the solutions produced results that were not different from selecting them through other techniques suggested in the literature [15]. If  $P$  is the population size, the remaining  $P - \alpha$  solutions are selected on the basis of fitness. The fitness vector of each solution is computed and the new population is initialized to an empty set. Solutions are copied into the new population one at a time from the parents. The probability of selecting a solution  $\Phi^{(p)}$  from the parent population to be copied is given by

$$\text{Prob}(\Phi^{(p)}) = [\lambda \mathbf{1} - \lambda]^T \mathbf{F}^{(p)} \quad (8)$$

where  $\lambda$  is chosen randomly from  $[0,1]$  with uniform probability. A total of  $Q = P - \alpha$  solutions are chosen in this manner.

With the exception of the  $\alpha$  Pareto-optimal chromosomes, the other code sets are subject to mutation, with a probability  $\mu$ , called the mutation rate in the algorithm. During mutation, each solution receives a small amount of perturbation. Mutation is carried out to make the search exploratory. A code  $\phi_x$  is picked at random from the solution and to it is added a random perturbation vector whose elements follow a uniform distribution in  $[-\pi, \pi]$ .

At the end of each iteration, the new population replaces the previous one. A predetermined maximum number of iterations  $\tau_{\text{max}}$  of the multiobjective EA is carried out before the best solutions are extracted as the final set of codes.

#### IV. RESULTS

The algorithm is fine tuned for fast convergence. The probability of a code set undergoing mutation, the mutation rate, is 0.5. Up to a maximum of  $\alpha_{\text{max}} = 100$  chromosomes in the Pareto front are taken from the parent population to the offspring population in each iteration. The quantities  $\sigma_{\text{sh}}$  and  $\kappa$  associated with fitness sharing are 0.1 and 4. The algorithm is run for a population size of 200 and for 5000 iterations. Simulations reveal that the optimization is completed in a small fraction of this time.

The EA algorithm is first used to generate  $K = N - 1$  codes, each with length  $N = 31$  (these values of  $K$  and  $N$  are chosen in order to provide a fair comparison with other complex codes such as FZC, Oppermann, and CI). Table I compares the  $R_{\text{CC}}$  and  $R_{\text{AC}}$  properties of the multiobjective EA-based code set with other code sets ( $N = 31$ ). It is evident that the EA-based approach not only provides codes that have “good” correlation properties but also yields a wide range of codes with intermediate  $R_{\text{CC}}$  and  $R_{\text{AC}}$  values. These intermediate codes tradeoff  $R_{\text{CC}}$  properties for  $R_{\text{AC}}$  and are vital for DS-CDMA system design. It can be observed that the intermediate Oppermann codes are comparable to the intermediate codes of the multiobjective EA, but the Oppermann codes cannot be designed for any length  $N$  (while maintaining the  $K = N - 1$  capacity limit). Table II demonstrates the ability of the multiobjective EA-based approach to generate codes of different lengths ( $N = 9, 16, 20$ ) each with a wide range of correlation properties. It is important to remember that since the EA-based code sets are not parametric (like the FZC, CI, and Oppermann), it is necessary to allocate memory to store the code set in use.

Fig. 1 shows the large number of code sets, with code length 8 and a wide range of correlation properties, generated using the EA and archived at various time instants. The solid lines represent the theoretical optimum from Theorem 1. Each point on the  $R_{\text{CC}} - R_{\text{AC}}$  plane represents a generated code set. It is clear from this figure that the multiobjective EA-based approach yields code sets that fall along the optimum theoretical line. Furthermore, the multiobjective EA-based approach also enables us to increase  $K$  while keeping the length of the code constant. This is a significant benefit of using the evolutionary approach as none of the well-known code sets are flexible

TABLE I  
COMPARISON OF EA-BASED CODES AND OTHER CODES

Code Set	$R_{CC}$	$R_{AC}$
GA-Based Code Set	0.334	19.604
	0.402	17.839
	0.594	12.358
	0.753	7.440
	0.999	0.214
CI Code	0.355	19.677
Gold Codes	0.970	0.900
EOE-Gold	0.950	0.952
FZC	1.000	0.344
Oppermann	0.400	18.200
	0.500	19.670
	1.000	0.620

TABLE II  
EA-BASED CODES FOR DIFFERENT  $N$

EA-Based Code Set	$R_{CC}$	$R_{AC}$
$N = 9$	0.370	5.037
	0.447	3.670
	0.665	2.795
	1.000	0.068
$N = 16$	0.356	9.679
	0.441	8.505
	0.627	5.796
	1.000	0.147
$N = 20$	0.350	11.532
	0.456	10.647
	0.556	8.702
	0.999	0.217

enough to support  $K \gg N$  codes of length  $N$ . Fig. 1 demonstrates that even with this increase in the number of codes, the genetic algorithm successfully determines the best possible code sets that lie along the Pareto front. Fig. 2 shows code sets that are obtained using the proposed method for  $N = 16$  and for three different values of  $K$ . As before, the codes lie along the theoretical Pareto front of Theorem 1. This again demonstrates the efficacy of the algorithm in deriving optimal code sets.

In Figs. 1 and 2, we observe that the best  $R_{CC}$  value achieved by our simple multiobjective EA increases with increasing  $K$ . This is not a surprise if one considers the fact that when  $K > N$ , the orthogonality among codes (for lag 0) is lost. Another observation from Fig. 2 is that the minimization of  $R_{CC}$  ceases when  $R_{AC}$  reaches a certain maximum value. This is once again easily explained by considering the spectrum of a code with high  $R_{AC}$ . The code sets with maximum  $R_{AC}$

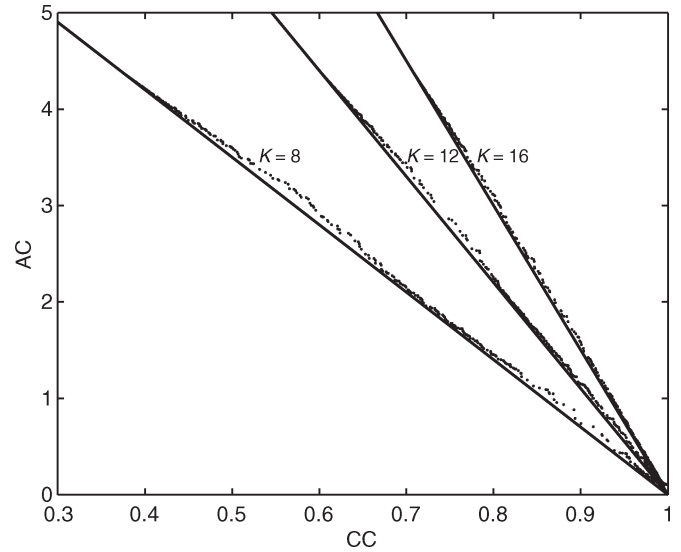


Fig. 1. Correlation properties of EA-based code sets with  $N = 8$ .

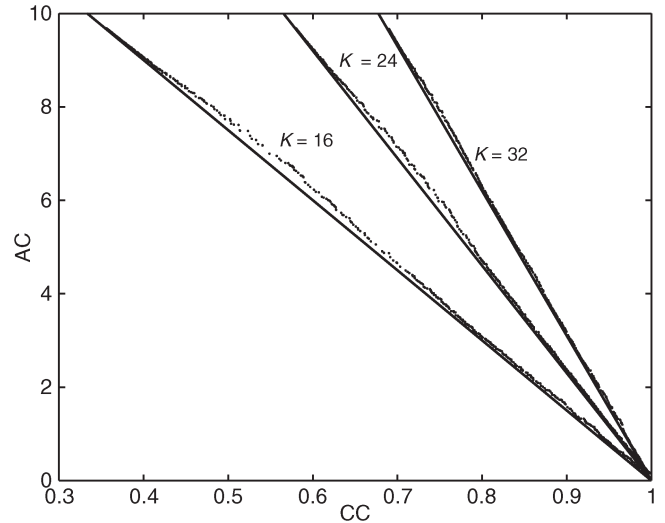


Fig. 2. Correlation properties of EA-based code sets with  $N = 16$ .

start behaving like an FDMA scheme and hence are not useful in a wideband DS-CDMA system. Therefore, the algorithm saturates near the code set that possesses FDMA-like characteristics. The multiobjective EA provides a large family of codes that lie along the middle of the theoretical line (see Figs. 1 and 2), and these are the code sets that are most useful for DS-CDMA design engineers.

The EA-based approach discussed above not only applies to code design with continuous phase angles but also to code design with discrete phase angles, e.g., multiples of  $2\pi/N$ . In this case, the initial population has code sets with discrete phases and discrete mutation is performed in each iteration. Having discrete phase angles significantly lowers the implementation complexity and also reduces the susceptibility to phase jitters. Since it also results in a finite set of possible code sequences, it is possible to save memory (required to store large codes) by storing the index of the sequence instead of the entire sequence. Fig. 3 illustrates the ability to design EA-based code sets with discrete phases. Here, the algorithm is used to generate code

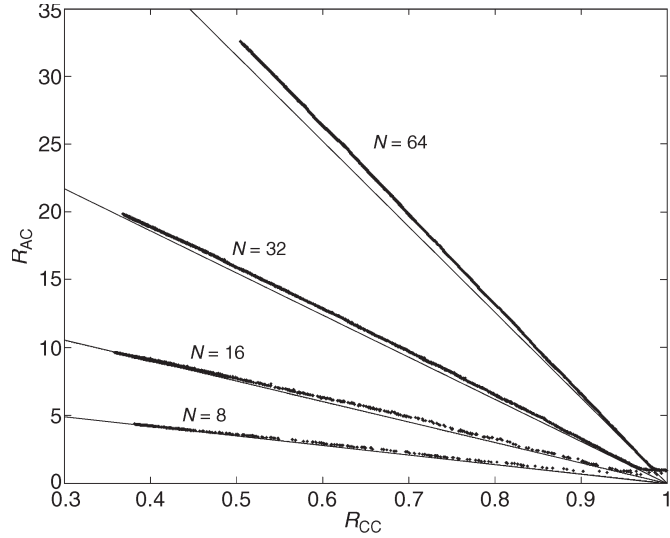


Fig. 3. Correlation properties of EA-based code sets with  $N = 8, 16, 32$  and  $64$  and discrete phase angles.

sets with length  $N = 8, 16, 32,$  and  $64$  and  $K = N$ . For each  $N$ , the discrete phases used are  $(2\pi/N)k, k = 0, 1, \dots, N - 1$ .

The complexity of the EA is a function of the desired  $R_{CC}$  for the DS-CDMA system and the length of code  $N$ . For practically useful  $R_{CC}$  values (between  $0.65$  and  $0.85$  representing the mid part of the theoretical line), the algorithm converges in a few hundred iterations (with discrete phases). The number of iterations increases linearly with increasing  $N$ , with  $N = 64$  requiring a few thousand iterations to converge. As expected, the convergence speed is slower with continuous phases than with discrete phases. It is important to note that if the end goal is to obtain a code set with a target  $R_{CC}$ , the objective function can be appropriately modified to increase the convergence speed for large values of  $N$ .

## V. CONCLUSION

In this paper, complex spreading codes for DS-CDMA were designed using a multiobjective evolutionary algorithm (EA). This approach offers flexibility in code design that cannot be achieved with any other technique. The benefits include: 1) the ability to design spreading sequences with no constraints on length; 2) the ability to design spreading code sets with a wide range of correlation properties (while achieving the theoretical optimum); and 3) increasing the capacity of a DS-CDMA system by enhancing the number of codes that can be designed for a given length  $N$ .

## APPENDIX

We can rewrite (1) as

$$R_{CC}K(K-1)N^2 = \sum_{x=1}^K \sum_{y=1}^K \sum_{l=1-N}^{N-1} |C_{xy}(l)|^2 - \sum_{x=1}^K \sum_{l=1-N}^{1-N} |C_x(l)|^2. \quad (9)$$

We can rewrite (3) as

$$\begin{aligned} R_{AC}KN^2 &= \sum_{x=1}^K \sum_{l=1-N}^{N-1} |C_x(l)|^2 - \sum_{x=1}^K |C_x(0)|^2 \\ &= \sum_{x=1}^K \sum_{l=1-N}^{N-1} |C_x(l)|^2 - KN^2 \\ \sum_{x=1}^K \sum_{l=1-N}^{N-1} |C_x(l)|^2 &= R_{AC}KN^2 + KN^2. \end{aligned} \quad (10)$$

Substituting (10) in (9), we get

$$R_{CC}K(K-1)N^2 = \sum_{x=1}^K \sum_{y=1}^K \sum_{l=1-N}^{N-1} |C_{xy}(l)|^2 - R_{AC}KN^2 - KN^2. \quad (11)$$

The first term on the right hand side can be further simplified (see [3]) as

$$\begin{aligned} \sum_{x=1}^K \sum_{y=1}^K \sum_{l=1-N}^{N-1} |C_{xy}(l)|^2 &= \sum_{x=1}^K \sum_{y=1}^K \sum_{l=1-N}^{N-1} C_x(l)C_y^*(l) \\ &= \sum_{l=1-N}^{N-1} \left| \sum_{x=1}^K C_x(l) \right|^2 \\ &= \sum_{l=1-N, l \neq 0}^{N-1} \left| \sum_{x=1}^K C_x(l) \right|^2 + K^2N^2 \\ &= K^2N^2 + 2 \sum_{l=1}^{N-1} \left| \sum_{x=1}^K C_x(l) \right|^2. \end{aligned} \quad (12)$$

The term on the right side has a lower bound of  $K^2N^2$ . Therefore, we can conclude that

$$\sum_{x=1}^K \sum_{y=1}^K \sum_{l=1-N}^{N-1} |C_{xy}(l)|^2 > K^2N^2. \quad (13)$$

Introducing this result in (12) and rearranging, we get

$$R_{CC}K(K-1)N^2 + R_{AC}KN^2 + KN^2 > K^2N^2 \\ R_{CC}(K-1) + R_{AC} > (K-1). \quad (14)$$

## REFERENCES

- [1] A. J. Viterbi, *CDMA: Principles of Spread Spectrum Communication*, 1st ed. Reading, MA: Addison-Wesley, 1995.
- [2] I. Oppermann and B. Vucetic, "Complex spreading sequences with a wide range of correlation properties," *IEEE Trans. Commun.*, vol. 45, no. 3, pp. 365–375, Mar. 1997.
- [3] D. V. Sarwate, "Bounds on crosscorrelation and autocorrelation of sequences," *IEEE Trans. Inf. Theory*, vol. IT-25, no. 6, pp. 720–724, Nov. 1984.

- [4] H. Fukumasa, R. Kohno, and H. Imai, "Design of pseudonoise sequences with good odd and even correlation properties for DS-CDMA," *IEEE J. Sel. Areas Commun.*, vol. 12, no. 5, pp. 828–836, Jun. 1994.
- [5] K. Yang, Y. Kim, and P. V. Kumar, "Quasi-orthogonal sequences for code division multiple-access systems," *IEEE Trans. Inf. Theory*, vol. 46, no. 3, pp. 982–993, May 2000.
- [6] W. O. Alltop, "Complex sequences with low periodic correlations," *IEEE Trans. Inf. Theory*, vol. IT-26, no. 3, pp. 350–354, May 1980.
- [7] P. V. Kumar and O. Moreno, "Polyphase sequences with periodic correlation properties better than binary sequences," *IEEE Trans. Inf. Theory*, vol. 37, no. 3, pp. 603–616, May 1991.
- [8] J. G. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.
- [9] B. Natarajan, C. R. Nassar, S. Shattil, M. Michelini, and Z. Wu, "High performance MC-CDMA via carrier interferometry codes," *IEEE Trans. Veh. Technol.*, vol. 50, no. 6, pp. 1344–1354, Nov. 2001.
- [10] M. B. Pursley, "Performance evaluation of phase coded spread spectrum multiple access communication—Part 1: System Analysis," *IEEE Trans. Commun.*, vol. COM-25, no. 8, pp. 784–790, Aug. 1977.
- [11] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, 1st ed. Reading, MA: Addison-Wesley, May 1989.
- [12] C. Fonseca and P. Fleming, "An overview of evolutionary algorithms in multiobjective optimization," *Evol. Comput.*, vol. 3, no. 1, pp. 1–16, 1995.
- [13] L. Haiming and G. Yen, "Rank-density-based multiobjective genetic algorithm and benchmark test function study," *IEEE Trans. Evol. Comput.*, vol. 7, no. 4, pp. 325–343, Aug. 2003.
- [14] J. Knowles and D. Corne, "Properties of an adaptive archiving algorithm for storing nondominated vectors," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 100–116, Apr. 2003.
- [15] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999.