World Congress on Nature and Biologically Inspired Computing
NaBIC 2009
Coimbatore, India
December 9-11, 2009

Efficient Design of Pulse Compression Codes Using Multiobjective Genetic Algorithm

Ajit Kumar Sahoo, Ganapati Panda and Pyari Mohan Pradhan
Department of Electronics and Communication
National Institute of Technology, Rourkela, Orissa, India
Email: ajitsahoo1@gmail.com, ganapati.panda@gmail.com, pyarimohan.pradhan@gmail.com

Efficient Design of Pulse Compression Codes Using Multiobjective Genetic Algorithm

Ajit Kumar Sahoo, Ganapati Panda and Pyari Mohan Pradhan
Department of Electronics and Communication
National Institute of Technology, Rourkela, Orissa, India
Email: ajitsahoo1@gmail.com, ganapati.panda@gmail.com, pyarimohan.pradhan@gmail.com

Abstract

Binary sequences having good aperiodic autocorrelation functions (ACFs) with low range sidelobes are required for many communication applications. However, few such good binary sequences are available having large sequence length. In this paper nondominated sorting genetic algorithm-II (NSGA-II) is used to generate biphase pulse compression codes from length 49 to 100. Pulse compression technique avoids the transmission of a signal having small pulse width and high peak power for better range resolution. The efficiency of these codes depends upon the energy content in the range sidelobes of their autocorrelation functions. Peak sidelobe level (PSL) and integrated side lobe level (ISL) are the two performance measures for pulse compression codes. In this paper the multi objective problem is designed by taking PSL and ISL as the objective functions.

1. Introduction

In a pulse radar system the transmitted pulse width should be as long as possible to increase the sensitivity of the system and as small as possible at the receiver for better range resolution. Range resolution is the ability of the radar receiver to discriminate nearby targets. The performance of range resolution radar would be optimal, if the coded waveform has impulsive autocorrelation. In pulse compression technique a long coded pulse is transmitted and the received echo is processed to obtain a relatively narrow pulse. Thus increased detection capability of a long pulse radar system is achieved while retaining the range resolution capability of a narrow pulse system. The range resolution is determined by bandwidth of the signal. Wide bandwidth is necessary for good range resolution. The signal bandwidth is obtained by modulating phase or frequency of the signal, while maintaining constant pulse amplitude. Mostly biphase pulse compression is used in radar system in which the phase of the transmitted signal is 0 degree relative to a local reference for a ’+1’ in the binary code and 180 degree for a’-1’. Phase coding can be used to reduce radio frequency interference between adjacent radar and make the waveform more robust to interference. Phase coded (PC) waveforms support better range-resolution compared with linear frequency modulated (LFM) pulses because the windowing functions used with LFM pulses to lower time sidelobes cause a broadening in the mainlobe response and a deterioration in range resolution, which is not present with phase coding. But the autocorrelation of phase coded waveforms contain range sidelobes, which have a negative influence on the detection performance of radar. These sidelobe can mask useful information and may give undesired gain to the signal from the unwanted target. A desirable property of the compressed pulse is that it should have low sidelobes in order to prevent a weak target from being masked in the time sidelobes of a nearby stronger target. The lower the sidelobes, relative to the main lobe peak, the better the main peak can be distinguished and, therefore the better is the corresponding code. The compressed pulse should not significantly degraded when the return signal has been doppler shifted due to target motion. Most of the modern radars incorporate pulse compression waveforms to avoid transmission of a pulse having a large peak power. The selection of a pulse compression codes depends upon the application and the environmental conditions. If the application is radar designed for a scenario dominated by distributed clutter then integrated side lobe level (ISL) is very important. On the other hand if the application requires detection of targets in the presence of large discrete clutter then the peak side lobe level (PSL) is more important. If the desired ISL or PSL performance is not achieved with a matched filter, some signal to noise ratio (SNR) gain may be sacrificed and a mismatch filter may be used to achieve the desired side lobe level.

Sequences having low off-peak signal value at the output of the matched filter have found extensive applications in radar and communication systems. Binary pulse compression codes such as the Barker code [1] or maximal-length sequences [2] are extensively used in many radar systems. The digital matched filter for the binary code is much less complex than the polyphase code, and the clutter rejection performance of the binary code is fair while that of the chirp method is poor. Barker sequences are the only known binary sequences with the lowest PSL of unity. The longest available barker sequence is of length 13.Many practical applications require longer codes to achieve higher signal to noise ratio. By exhaustive computer search program, Lindner [3] found all binary sequences up to length 40 with minimum PSL. Cohen $et\ al.$ [4] extended those results to length 48. Using a neural network approach, Hu $et\ al.$ [5] obtained useful binary sequences for lengths up to 100.Kerdock $et\ al.$ [6] found the maximum length of a code for a given peak sidelobe level. Boehmer[7] has developed an analytical technique for generating good binary pulse compression codes. Rao and Reddy [8] has obtained larger length binary sequences with minimum sidelobes in their aperiodic autocorrelation pattern by cyclically shifting the Legender sequences.Levanon [9] used mismatch processing to optimize integrated or peak sidelobe levels of long binary signals. Baden and Cohen [10] has explained a method for generating a filter that minimizes the energy in the side lobes after multiplying the sidelobes by a given weighing function. Therefore the pulse compression problem of radar may be viewed as a multiobjective problem in the sense that the PSL and ISL need simultaneous minimization. This burning issue has been studied in [5] using artificial neural network (ANN) technique but not as a multiobjective problem.

The organization of the paper is as follows. In section 2, the basis of generating new aperiodic codes is outlined as a multiobjective optimization problem in which the PSL and ISL are simultaneously optimized. The multiobjective tool used as to optimize these two objective functions is discussed in section 3. The new sequences generated by the proposed method are obtained by NSGA-II through simulation study using MATLAB and are listed in section 4. Finally in section 5 the conclusion of the investigation is highlighted.

2. Multiobjective Formulation

Let an L length binary sequence is given by

$$S = \{s_1, s_2, s_3, \cdots, s_L\} \tag{1}$$

where each element of S has value is either +1 or -1.

The output of the matched filter that is the aperiodic autocorrelation function for positive delays is given as

$$C_k(S) = \sum_{i=1}^{L-k} s_i s_{i+k} \tag{2}$$

where $k = 0, 1, 2, \cdots, L-1$

A significant problem inherent in biphase pulse compression is that the autocorrelation function does not yield a perfect impulse, that means it does not produce $C_k(S) = 0$ for $k \neq 0$.Any non zero value of $C_k(S)$ for $k \neq 0$ is referred to as sidelobe where as the zero-offset correlation value $C_0(S)$ is called the mainlobe. The difference between a pulse compression waveform and a simple pulse waveform is in the existence and magnitude of these sidelobes. The amplitude of the main peak of an autocorrelation is a measure of signal-to-noise ratio improvement for a given code. The sidelobes limit the usefulness of a code regardless of the strength of the mainlobe. This is because sidelobes signify ”self noise” which may be much greater than the system noise. Codes are chosen for a given application based on their length and sidelobe levels.

There are two frequently used sidelobe measures. The first one is the peak sidelobe level(PSL) which is the largest sidelobe in the autocorrelation function of the code and is defined as

$$PSL = Max\left|(C_k(S))\right|, k \neq 0 \tag{3}$$

The second is the integrated sidelobe level (ISL) defined as the total energy contained in all the sidelobes. As the signal is real valued the autocorrelation is real and symmetric about the zero delay. The ISL is represented as

$$ISL = 2\sum_{k=1}^{L-1} C_k^2(S) \tag{4}$$

The PSL and ISL is represented as the terms defined in (3) and (4) are simultaneously minimized to achieve desired aperiodic codes.

3. Nondominated Sorting Genetic Algorithm-II

In 1967, Rosenberg suggested a genetic search to the simulation of genetics and the chemistry of a population of single-celled organisms with multiple properties or objectives [11]. Genetic algorithm (GA) was originally proposed by J. Holland [12] which imitates Nature’s robust way of evolving successful organisms. The first practical algorithm, called vector evaluated genetic algorithm (VEGA), developed by Schaffer [13]. One of the constraints of VEGA is its bias towards some Pareto-optimal solutions. Subsequently Deb and Srinivas [14] proposed the nondominated sorting genetic algorithm(NSGA) which is a popular nondomination based genetic algorithm for multi-objective optimization problem. It is a very effective algorithm but has been generally not preferred for its heavy computational complexity, lack of elitism and for choosing the optimal parameter values. A modified version, NSGA-II was developed by Deb et al. [15] as a better sorting algorithm which incorporates elitism and without using a sharing parameter. A brief outline of NSGA-II algorithm is given below.

1) **Population Initialization**: The population contains a set of chromosomes. Each chromosome is initialised randomly with binary bits having length same as the code length.
2) **Non-Dominated sort**: The initialized population is sorted based on non-domination
   - for each individual p in main population P perform the following

    - Initialize the set of individuals dominated by p $S_p = \phi$.
    - Initialize the number of individuals that dominate p i.e. $n_p = 0$.
    - for each individual q in P
        - $*$ if p dominates q then
            - $\cdot$ $S_p = S_p \cup \{q\}$
        - $*$ else if q dominates p then
            - $\cdot$ $n_p = n_p + 1$
    - if $n_p = 0$ then p belongs to the first front and rank of individual p i.e. $p_{rank} = 1$. Update the first front F_1 by adding p to front one i.e. $F_1 = F_1 \cup \{p\}$
- This is carried out for all the individuals in main population P.
- Initialize the front counter $i = 1$
- perform the following if i^{th} front is nonempty i.e. $F_i \neq \phi$
    - $Q = \phi$. The set Q is used to store the members of the next front.
    - for each individual p in front F_i
        - $*$ for each individual q in S_p
            - $\cdot$ $n_q = n_q - 1$,decreament the domination count for individual q
            - $\cdot$ if $n_q = 0$, set $q_{rank} = i + 1$. Update q i.e. $Q = Q \cup q$.
    - $i = i + 1$.
    - set Q is the next front and hence $F_i = Q$.

3) **Crowding Distance Assignment**: Once the non- dominated sorting is complete the crowding distance is assigned. Since the individuals are selected based on rank and crowding distance, all the individuals in the population are assigned a crowding distance value front wise. The crowding distance is calculated as
    - For each front F_i, n is the number of individuals.
        - Initialize the distance for all individuals to be zero i.e. $F_i(d_j) = 0$, where j corresponds to an individual in front F_i.
        - for each objective function m
            - $*$ Sort the individuals in front F_i based on objective function m i.e $I = sort(F_i, m)$.
            - $*$ Assign infinite distance to boundry values for each individual in F_i i.e $I(d_1) = \infty$ and $I(d_n) = \infty$
            - $*$ for $k = 2$ to $(n - 1)$
                - $\cdot$ $I(d_k) = I(d_k) + \frac{I(k+1)\cdot m - I(k-1)\cdot m}{f_m^{max} - f_m^{min}}$
                - $\cdot$ where $I(k) \cdot m$ is the value of the m^{th} objective function of k^{th} individual in I

4) **Selection**: The selection is carried out using a crowded-comparison-operator $(\prec_n)$. The comparison is carried out as below
    a) non-domination rank p_{rank} i.e. individuals in front F_i will have their rank as $p_{rank} = i$.
    b) crowding distance $F_i(d_j)$
    - $p \prec_n q$ if
        - $p_{rank} < q_{rank}$
        - or if p and q belong to the same front F_i then $F_i(d_p) > F_i(d_q)$ i.e the crowding distance of p should be more q.

    The individuals are selected using a binary tournament selection with crowded comparison operator.
5) **Genetic Operators**: Single point crossover and mutation operations as used in GA[16] are also employed in NSGA-II.
6) **Recombination and Selection**: The offspring population is combined with the current generation population and the total population is sorted based on nondomination. The new generation is filled by chromosomes from each front subsequently until the population size exceeds the current population size N. If by adding all the individuals in front F_j the population exceeds N then individuals in front F_j are selected based on their crowding distance sorted in the descending order until the population size is N.

4. Simulation results

The NSGA-II algorithm dealt in section 3 is used to obtain the desired non periodic binary strings yield lowest possible PSL and ISL. PSL and ISL are the two cost functions are minimized using NSGA-II algorithm to generate binary strings varying the lengths from 49 to 100. The PSL and ISL values for each chromosome are determined according to (3) and (4) respectively. The population is sorted based on non-domination. Each individual in first front are given a rank value of 1 and the same in second is assigned a rank value of 2 and so on. Parents are selected from the population by using binary tournament selection based on the rank and crowding distance. An individual is selected if its rank is less than the other or if its crowding distance is greater than the other. The selected population generate offspring using crossover and mutation operators. The crossover and mutation probability used are 0.8 and 0.1 respectively. The total population including the current population and current off springs is sorted again based on non-domination and only the best N individuals are selected. The selection is based on rank and crowding distance on the last front. The population size N is taken as 200 and the number of generations varied from 150 to 200.As the sequence length increases generations increased to get better result. In multi objective algorithms a pareto front contains more than one sequence for each length. The sequences found by the proposed method for length 49-100 which are better than [5] are listed in Table 1. The PSL and ISL values of [5] are also presented in Table 1 for the purpose of comparison.

Table 1: Sequences

Seq Len	[5]		NSGA II		**Sequence**
	PSL	ISL	PSL	ISL	
49	5	688	4	560	0011111000111111011000000100101100111001101010101
49	5	688	5	472	0011111100111111011000000100101100110100101010101
50	5	674	5	658	00010010011100111100100101000110111010101111111010
51	5	618	5	594	110000010000000110100110011100010001011110010101101
52	5	708	5	604	0000110011001100001111000000000101000101101010101101101
53	5	700	5	604	00100101100100011001111001110111111001010100001111101
54	5	750	5	702	111001110111010110011110100110011111000010101101111110
55	5	902	5	830	1000111100010001011111001000110101011111110101101001001
56	6	1000	5	920	00111000100011011100011010001000000010011011010000111101
56	6	1000	6	776	00111010100011011100011010001000000010111011010000100101
57	6	920	6	784	110110010111110101110110101110001100010010000111111011000
58	6	1130	5	890	1001010001110111000101011111000011101001110110010000000010
58	6	1130	6	698	1001010001110011000101011011000011101001110110010000000000
59	5	1050	5	994	00000100101100001111111100010101100011110100011001001110101
60	6	1212	6	836	110000001010100111000001101011011011101001001100110001111011
61	6	1068	6	780	1010100101110001111000010011111101110111100100001100110100100
62	6	1094	6	998	10000001101101010010101110100000000100000111100111001100101 1011
63	5	934	5	902	100110101101111000010000000111010000010111001010001101010110011
64	6	1336	6	1024	0111011101010100001010011101001001011011110000100110000000110000
65	6	1328	6	1160	11111001101001011101000101110011000111001110110101001011111011110
66	6	1410	6	1010	011000110001010001010101101100110110111111100101000011111000110111
67	6	1218	6	1098	1111100000111101111111010110001010010100101100101100110010001100101
68	6	1252	6	1180	10111100110001010101100111000000110100000001001100101101011111 0010100
69	6	1500	6	1236	101100100110010010100011101000010110001110010100111111100100001000001
70	6	1454	6	1422	0010110000110001110100110110000010101000010010001011101111011000100111
71	7	1860	7	1518	11011100010111001010110110100111101101010111111111000100100100110000111
72	7	1744	7	1568	010001010111111101010010001111001100101111011011010111011010001111100001
73	6	1504	6	1312	0100010101101110011001010001111100000111110010100001100010110111111110110
74	7	1786	7	1578	01110010110000110000111010001001001011110000000001000001101110011001110101 0
75	6	1250	6	1130	010111101010111100011011100111011001010101111101001001001100000110011110011
76	7	1972	7	1580	0101000010010000101000111001011101100011110010011010100010111111110010001011
77	6	1560	6	1396	11001011001110010110000110001000010001000101100010111100101010100100000111111
78	9	2150	7	1670	111111100000110100000000101000111000111001001010010111001000100011010110010000 1
79	7	2246	6	1526	1111101001101010001010110100101100010000101100110111111111000111101110110000011 1
79	7	2246	7	1454	0101101001101010001010110100101110010000101100110111111111000111101110110000011 1
80	8	2240	7	1584	10000110001001111000001010111100100110011010101001011101111111101100100101000111 1
81	7	1824	7	1720	001100111001000101010101000000110010110100111110110101001000000110110100011100001
82	7	2162	7	1826	0000001111000000111100101000000000000110101010011000110110101101110011001011001 10100

continued on the next page . . .

Sequences (*continued ...*)

Seq Len	[5]	[5]	NSGA-II	NSGA-II	Seq
	PSL	ISL	PSL	ISL	
83	7	2202	7	1986	010000010101001111010100000111100000010001100010100100110111101101100100111010110111
84	7	2636	7	2372	101100010111110100001110100011000011110010111011010100100000100101110011000101000100
85	7	2332	7	1916	1111110000011101101011001010010011011111011011100101100010101001110010100011001110000
86	7	2062	7	1902	01110011011101000100001000111100010100000001010000100011111010011011110010100110110001
87	8	2520	7	2046	001001000011001001111111011100010111100001001110101001010000000110001101001100000110010
87	8	2520	8	1982	001001000011001001011111011100010111100000001110101001010000010110001101000100000110010
88	7	2264	7	2224	1011011000010101111100100011001101001010101001100011111000011001100011111011111111110100
89	7	2128	7	2112	01111011010100000110001111110101010011110010000010110010001011100101101110011111100011001
90	8	2698	8	2346	010001100011110010111010010010000010111011101001000111011110100100011001000010101100010011
91	7	2226	7	2082	1110000000001100111111111100111011111001001110101100101101001011100010101010010101100111010
92	7	2004	7	1844	00110110010011010111001001010011100000011010011010100100111111111101010110001110111100011101
93	9	3316	7	2476	0111101111011001100110100000000010100110000111110011000001010101001111100000011000110101010010
94	8	3062	8	2630	1111001000110001110110001011000110100111000101110110101111101001101010001011111111101101101100
95	8	3374	8	2262	111010010010101000101111000010111101101000000010001000011001010001000001100011111001110010011001
96	10	3520	8	2416	0001000111111000010100001111011110101011001101011110010011010100010111111111011011011100001001100
97	9	3576	8	3440	110010111010101110010001000010110000001001110111100100011001011010111000000010000001101111010011110
97	9	3576	9	2696	01001010100010101001010100001011000000100111011110010001100101101111100000011000001110011010011110
98	8	2930	8	2594	01000000000011000110000111011101100111000010101101101011110000011001111010010111010011001001011101
99	8	2794	8	2386	101111000100100100010000000011011001101011011100011001011011010110101010001001111100011010001000111
100	8	3748	8	2508	001111110000110110111001110011011010111101100101100110101001011101100001000011101111111010101110001010

5. Conclusion

By using NSGA-II algorithm as given in section 3 a list of sequences of length 49-100 is obtained, as shown in Table 1 which also contains PSL and ISL value of the sequences.0's are used in place of -1's to conserve space. It is observed from the table that the codes having same PSL value have less ISL value compared to [5].Further it observed that at lengths 49,56,58,79,87 and 97 two different codes are found out of which one code optimal with respect to both PSL and ISL. The result reveals that this method performs better than the approach described in [5]. The search for optimum sequence depends on the selection of the initial population of parent sequences. As the sequence length increases the search procedure requires more time for obtaining a good solution. The quality of solution improves with increase in the number of generations.

References

[1] A.W. Rihaczek,and R.M.Golden, *Range sidelobe suppression for Barker codes*, IEEE Trans. Aerosp. Electron. Syst., Vol.7,No.6, pp. 1087-1092,Nov.1971.

[2] M. I. Skolnik, *Radar Handbook (2nd ed.)*. New York: McGraw-Hill, 1990.

[3] J. Lindner, *Binary sequences up to length 40 with best possible autocorrelation function*, Electron. Lett., vol. 11, no. 21, p. 507, Oct.1975.

[4] M. N. Cohen, M. R. Fox, and J. M. Baden, *Minimum peak sidelobe pulse compression codes*, in Proc. IEEE Int. Radar Conf., pp.633-639, 1990.

[5] F. Hu, P. Z. Fan, M. Darnell, and F. Jin, *Binary sequences with good aperiodic autocorrelation functions obtained by neural network search*, Electron. Lett., vol. 33, no. 8, pp. 688-689, Apr. 1997.

[6] A.M. Kerdock, R.Mayer and D. Bass *Longest binary pulse compression codes with given peak sidelobe levels*,Proc. of IEEE, Vol.74,No.2,p. 366,Feb. 1986.

[7] A.M. Boehmer, *Binary pulse compression codes*,IEEE Trans. on Information Theory,Vol. 13 No.2,pp.156-167,Apr 1967.

[8] K.V. Rao, and V.U. Reddy,*Biphase sequences with low sidelobe autocorrelation function*, IEEE Trans. Aerosp. Electon. Syst.,Vol.22,No.2 pp. 128-133,Mar.1986.

[9] N. Levanon, Cross-correlation of long binary signals with longer mismatched filters IEE Proc. Radar Sonar Navig., Vol. 152, No. 6, pp.377-382,Dec.2005.

[10] J.M. Baden, and M.N. Cohen, *Optimal peak sidelobe filters for biphase pulse compression* Proc. IEEE Int. Radar Conf., pp. 249-252,1990.

[11] R. Rosenberg, *Simulation of Genetic Populations with Biochemical Properties*, PhD thesis, 1967.

[12] J. H. Holland, *Adaptation in natural and artificial systems*, University of Michigan Press, Ann Arbor (1975).

[13] J. D. Schaffer, *Some experiments in machine learning using vector evaluated genetic algorithms*, PhD Thesis, Nashville, TN: Vanderbilt University.

[14] N. Srinivas and K. Deb, *Multiobjective function optimization using nondominated sorting genetic algorithms*, Evolutionary Computation , vol. 2, no. 3, pp. 221-248,1995

[15] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, *A fast and elitist multiobjective genetic algorithm: NSGA-II*, IEEE Tran. on Evolutionary Computation,Vol. 6,No.2,pp.182-197,Apr. 2002.

[16] G.Panda, BabitaMajhi, D.Mohanty, A.K.Sahoo, *A GA-Based pruning strategy and weight update algorithm for efficient nonlinear system identification*, Second International Conference Pattern Recognition and Machine Intelligence (PReMI), ISI Kolkata, pp.244-251, India,2007.