

Multi-Objective Evolutionary Algorithm Determined Radar Phase Codes

Jeremy Stringer, Gary Lamont, and Geoffrey Akers

Department of Electrical and Computer Engineering

Air Force Institute of Technology

Dayton, Ohio 45433-7765

Email: jeremy.stringer@afit.edu

Abstract—Solving complex real world Multi-Objective Optimization problems is the forte of Multi-Objective Evolutionary Algorithms (MOEA). Such algorithms have been part of many scientific and engineering endeavors. This study applies the NSGA-II and SPEA2 MOEAs to the radar phase coded waveform design problem. The MOEAs are used to generate a series of radar waveform phase codes that have excellent range resolution and Doppler resolution capabilities. The study compares the ability of NSGA-II and SPEA2 to continually evolve (phase code) solutions on the Pareto front for the problem while maintaining a diversity of solutions (phase codes). Results demonstrate that for the radar phase code problem NSGA-II provides a more diverse population of acceptable solutions and therefore a greater number of different viable phase codes when compared to the solutions provided by SPEA2

I. INTRODUCTION

There are many areas of scientific endeavor that require a solution to a problem with many competing objectives. Traditional solution methods have a difficult time coming to grips with such complex problems [1]. It was for just such problems that Multi-Objective Evolutionary Algorithms (MOEA) were designed. MOEAs attempt to stochastically find solutions that satisfy all objectives, and any imposed constraints at the same time. For example, MOEA can generate a series of solutions where solutions that increase one objective value over that of a given solution, require a decreased value, with respect to the same solution, for a different competing objective (maximization problem). Such points are referred to as Pareto optimal solutions, and are the goal of MOEAs.

This study applies a MOEA to the Multi-Objective Problem (MOP) of designing phase coded radar waveforms that have both good range resolution and Doppler resolution, while maintaining a high Probability of Detection (P_d). Good range resolution requires high bandwidth, indicating short pulse waveforms. P_d , however, is dependent upon the energy received per pulse, and thus requires a long pulse length. Good Doppler resolution also requires a long pulse length. Given the above dependencies, some trade-offs are required in radar waveform design. By applying multiple MOEAs to the problem, it is hoped that the trade-offs required are made in an “optimal” way, and a series of good radar phase codes generated.

Section two of this work describes the radar phase code waveform problem in more detail. Section three then describes

two MOEA that are used to solve the phase code MOP. Section four details the experimental design used to test the MOEA in solving the problem. Section five presents details of the experimental results, followed by a statistical analysis of the results in section six. The paper concludes with a few brief remarks and direction for future work in the area.

II. MOP PROBLEM DOMAIN DESCRIPTION

Radar range resolution is inherently a function of radar waveform bandwidth. Short pulsed waveforms have higher bandwidths than longer pulse waveforms. However, these short waveforms can limit the probability of detection for a given transmit power. Furthermore, short pulse waveforms can limit the accuracy of Doppler measurements [2]. To overcome these limitations, radar waveform designers employ phase codings in the transmitted signals to increase the bandwidth of long duration pulses [3]. In past literature, phase coded waveforms with desirable properties have been found through exhaustive search. Some more common codes, such as Barker codes, have been achieved that have desirable properties, however, these codes are limited in length. More recent studies have applied convex optimizers, Particle Swarm Algorithms, and greedy techniques to attempt to create good radar phase codes [4], [5], [6]. The problem at hand is to develop new phase codings for radar waveforms, that are of sufficient length to allow for high P_d values and doppler accuracies, with bandwidths sufficient for accurate range determination. The initial portion of the following derivation follows that of [4].

To develop the mathematical formulation of the Multi-Objective Problem (MOP), consider a transmitted radar signal consisting of a coherent burst of pulses:

$$s(t) = a_t u(t) e^{j(2\pi f_0 t + \phi)}, \quad (1)$$

where a_t is the transmit signal amplitude, $j = \sqrt{-1}$ and

$$u(t) = \sum_{i=0}^{N-1} a(i) p(t - iT_r), \quad (2)$$

is the signal’s complex envelope. Also in the above, $p(t)$ is the envelope of a single pulse, T_r is the pulse repetition interval (PRI), $[a(0), a(1), \dots, a(N-1)] \in \mathbb{C}^N$ is the unit norm phase code, f_0 is the carrier frequency, and ϕ is a random phase. The

backscattered signal received from a target with two way time delay τ is:

$$r(t) = \alpha_r e^{j2\pi(f_0+f_d)(t-\tau)} u(t-\tau) + n(t), \quad (3)$$

where α_r is the complex echo amplitude, accounting for all channel effects, f_d is the target Doppler frequency, and $n(t)$ is an additive disturbance i.e. clutter and thermal noise.

After down-conversion, and application of a matched filter, the received signal is sampled at $t_k = \tau + kT_r, k = 0, \dots, N-1$ providing the observables:

$$v(t_k) = \alpha_r a(k) e^{j2\pi k f_d T_r} + w(t_k), \quad k = 0, \dots, N-1 \quad (4)$$

where $w(t)$ is the down converted and filtered disturbance component. Then, denoting by $\mathbf{c} = [a(0), a(1), \dots, a(N-1)]^T$ (T being the vector transpose) the N -dimensional column vector containing the phase code, $\mathbf{p} = [1, e^{j2\pi f_d T_r}, \dots, e^{j2\pi(N-1)f_d T_r}]^T$ the temporal steering vector, $\mathbf{v} = [v(t_0), v(t_1), \dots, v(t_{N-1})]^T$, and $\mathbf{w} = [w(t_0), w(t_1), \dots, w(t_{N-1})]^T$, the backscattered signal received at the radar receiver is given by:

$$\mathbf{v} = \alpha_r \mathbf{c} \odot \mathbf{p} + \mathbf{w}; \quad (5)$$

where \odot is the Hadamard element-wise product.

Target detection in radar systems utilizes statistical hypothesis testing on the received radar data. When a target is present, the observables are given as (5). Then, if there is no target present, the observed data is $\mathbf{v} = \mathbf{w}$, i.e. noise only. When the interference vector \mathbf{w} is modeled as complex, zero mean, circular White Gaussian noise, the complex echo amplitude is modeled as Gaussian, and using a Neyman-Pearson test [7], the detection probability is:

$$P_d \triangleq \exp \left(\frac{\ln P_{fa}}{1 + \sigma_a^2 (\mathbf{c} \odot \mathbf{p})^H \mathbf{M}^{-1} (\mathbf{c} \odot \mathbf{p})} \right), \quad (6)$$

where $\sigma_a^2 = E\{|\alpha|^2\}$, P_{fa} is the desired probability of false alarm, and $\mathbf{M} = \mathbf{E}\{\mathbf{w}\mathbf{w}^H\}$ is the correlation matrix of the interference. All locations of H represent the Hermetian transpose operator [4].

The Doppler accuracy of a waveform is governed by the Cramer-Rao Lower Bound (CRLB), given by [4]:

$$\Delta_{CR}(f_d) \triangleq \frac{\Psi}{2 \frac{\partial \mathbf{h}^H}{\partial f_d} \mathbf{M}^{-1} \frac{\partial \mathbf{h}}{\partial f_d}}, \quad (7)$$

where $\mathbf{h} = \mathbf{c} \odot \mathbf{p}$ and $\Psi = \frac{1}{E\{|\alpha|^2\}}$.

If $\mathbf{u} \triangleq [0, j2\pi, \dots, j2\pi(N-1)]^T$ then:

$$\frac{\partial \mathbf{h}}{\partial f_d} = T_r \mathbf{c} \odot \mathbf{p} \odot \mathbf{u}. \quad (8)$$

The above allows the CRLB to be written as:

$$\Delta_{CR}(f_d) \triangleq \frac{\Psi}{T_r^2 (\mathbf{c} \odot \mathbf{p} \odot \mathbf{u})^H \mathbf{M}^{-1} (\mathbf{c} \odot \mathbf{p} \odot \mathbf{u})}. \quad (9)$$

To meet all radar waveform requirements, (6) needs to be maximized, and (9) needs to be minimized. If the following definitions are made:

$$(\mathbf{c} \odot \mathbf{p})^H \mathbf{M}^{-1} (\mathbf{c} \odot \mathbf{p}) \triangleq \mathbf{c}^H \mathbf{R} \mathbf{c} \quad (10)$$

and

$$(\mathbf{c} \odot \mathbf{p} \odot \mathbf{u})^H \mathbf{M}^{-1} (\mathbf{c} \odot \mathbf{p} \odot \mathbf{u}) \triangleq \mathbf{c}^H \mathbf{R}_1 \mathbf{c}, \quad (11)$$

where $\mathbf{R} = \mathbf{M}^{-1} \odot (\mathbf{p}\mathbf{p}^H)^*$ and $\mathbf{R}_1 = \mathbf{M}^{-1} \odot (\mathbf{p}\mathbf{p}^H)^* \odot (\mathbf{u}\mathbf{u}^H)^*$ are positive semidefinite [4] and the symbol $*$ is the complex conjugation operator, the *optimization problem* can then be rewritten as:

$$\max_{\mathbf{c}} (\mathbf{c}^H \mathbf{R} \mathbf{c}, \mathbf{c}^H \mathbf{R}_1 \mathbf{c}); \quad (12)$$

where the notation denotes solving the multi-objective problem over the variable \mathbf{c} .

Solving the above optimization problem, derives phase coded waveforms that are Pareto Optimal with respect to P_d and Δ_{CR} . Such functions, however, may have bad autocorrelation functions and thus given poor range resolution or have undesired range ambiguities. To ensure phase codes with good ambiguity functions, the *constraints* of low integrated sidelobe ratio (ISLR) and peak sidelobe ratio (PSLR) are added to the radar phase code MOP. ISLR is given by:

$$ISLR = \frac{1}{|y_0|^2} \sum_{n \neq 0} |y_n|^2, \quad (13)$$

where y_0 is the voltage level of the autocorrelation peak, and the $y_n, n \neq 0$ are the values of the autocorrelation voltage at all other locations. The measure relates the energy in the autocorrelation peak to the energy in all of the sidelobes. The PSLR is given by:

$$PSLR = \frac{|(y_n)_{peak \text{ sidelobe}}|^2}{|y_0|^2}, \quad (14)$$

and is the ratio of the energy in the main autocorrelation peak to the energy in the largest autocorrelation sidelobe. The power level of the phase codes are also constrained by ensuring each code is of unit modulus. All three constraints, (13), (14), and $||\mathbf{c}|| = 1$, are handled by applying a penalty function for their violation, such that the value of the objective functions are increased when the constraints are violated. There are numerous methods for dealing with constraints. Penalty functions are chosen as even infeasible solutions, in terms of the power constraint, may produce solutions with good ISLR and PSLR value and constraint methods such as repairing infeasible solutions may destroy the “good” information present in these solutions.

III. MOEA COMPUTATION DOMAIN DESCRIPTION

Evolutionary Algorithms (EA) attempt to solve complex problems by mimicking the evolutionary process of nature, as theorized by Darwin [8]. EAs operate on a population of solutions, each referred to as a chromosome. Each chromosome represents an individual, i.e., a complete solution to the given optimization problem. Each chromosome of instantiated decision variable values is assigned a fitness value with regard to the optimization criteria. Some method is used to choose the best solutions for reproduction. The selected chromosomes are then subjected to a crossover operator, that attempts to take good building blocks of genetic material from each of the

good solutions to form a new solution. The new solutions are then mutated according to a given mutation rate. This MOEA operation approach is then executed for a selected number of *generations* or until convergence to the true Pareto Front (*PF*), discussed below, is achieved.

Pareto optimal solutions are solutions that are non-dominated. That is a given solution \mathbf{x} is dominated if for all other solutions, $\mathbf{x} x_i \leq y_i$ and $\exists! x_j \in \mathbf{x} | x_j < y_j \forall y_i \in \mathbf{y}$. For a full discussion on Pareto optimality see [9]. In solving a single objective problem the goal is to find a single solution that is the global optimum for that particular optimization problem. When dealing with MOEAs, however, there may be no global optimum solution that is optimal for all objectives. The goal is then to find a set of Pareto optimal solutions that form a PF in objective space. MOEAs attempt to find an uniform diversity of points along the true PF for the optimization problem. At each point in time, the MOEA has a set of non-dominated solutions called $PF_{current}$. As the algorithm progresses it should store the overall non-dominated solutions from $PF_{current}$ in PF_{known} , the best set of non-dominated vectors generated by the MOEA thus far. The final solution used from the algorithm is then chosen by the decision maker from the PF_{known} [9].

Two MOEAs that are used by many throughout the literature are the Non-dominated Sorting Genetic Algorithm (NSGA-II), and the Strength Pareto Evolutionary Algorithm (SPEA2). These two algorithms have been chosen in the past, and are chosen for this study, because of their ability to find a diversity of non-dominated points close to PF_{true} , over a wide variety of test Multi-Objective Problems (MOP). These two MOEA also have demonstrated success in solving real world problems. It must be mentioned, that the past success of any MOEA, does not *a-priori* guarantee good performance on any future MOP. NSGA-II and SPEA2 are chosen, not because they are guaranteed to find a good set of phase codes for radar waveforms, but because the algorithms possess the properties that allow them to achieve the three goals:

- 1) Preserve non-dominated points (elitism) with $PF_{current} \rightarrow PF_{known}$
- 2) Progress or guide PF_{known} towards PF_{true}
- 3) Generate and maintain a diversity of points on the PF_{known} (phenotype) and/or Pareto optimal solutions P_{known} (genotype)

These characteristics are elitist preservation of non-dominated solutions in PF_{known} , and niching/sharing mechanisms to keep a diversity of points in both $PF_{current}$ and PF_{known} . Both algorithms also possess a mechanism by which they retain solutions near the edge of the Pareto front. This is important as algorithms, especially NSGA-II, have a tendency to prefer solutions on the interior of PF_{known} [10].

With regard to any MOEA, selection of the proper solution representation of decision variables has a large impact on selection of other algorithm constructs. Equation (7) represents the objective function for the MOP. The values for c are complex. However, computers deal only with real representations. A proper representation for the radar phase code problem, is

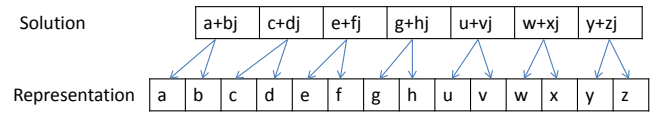


Fig. 1. Solution to Genotype Mapping

then a simple mapping from the complex domain to the real domain. The mapping that makes sense is to map the real and imaginary parts of the complex solutions to a single string of real values, shown in Fig 1. The resulting string is twice as long as the number of decision variables, as each decision variable is complex, and must be represented by “real” and “imaginary” vectors, combined into a single solution vector. Real number vector representations allow for a number of different crossover and mutation operators. For simplicity, and because there are no *a-priori* known “best” mutation and crossover operators, intermediate crossover and uniform random mutation are used in both NSGA-II and SPEA2. Further, in both NSGA-II and SPEA2, binary tournament selection is used by many researchers [9], and is used for this study.

The population size, to a large degree, determines the computational loading of a MOEA. This is because the majority of computation time is spent calculating the objective function values [8], [9]. For the radar phase code problem, the objective evaluation requires two matrix multiplies. The complexity of the objective function is then $O(MN^3)$, where N is the phase code length, and M is the population size. Doubling the population size, then doubles the computation time required.

Thanks to the open source software community, there is a panoply of MOEA software available for solving MOP. The decision of what software package to use for a particular MOP, then, is based upon the implementer’s familiarity with the software, and the software’s ability to deal with the objective function evaluation. It is the objective function evaluation, that is the biggest software decision factor, with regard to designing radar phase codes. The radar phase code problem requires solution of a matrix algebra problem, where the size of the matrices is dependent upon the size of the phase code desired. Evidently a software package able to handle large sized matrix operations is required for the radar phase code MOP to be useful.

While the Java and C++ libraries contain built in mathematical packages, the Mathworks software MATLAB excels in matrix operations, and is therefore chosen for this study. There is a large MATLAB community that has designed software for many applications, including MOEAs [11]. From this community effort, the *A NSGA-II Program in MATLAB (NPGM)* and *MOEA-SPEA2-MATLAB* implementations are chosen for solving the poly-phase coded radar waveform MOP.

IV. DESIGN OF EXPERIMENTS

The goal of the current effort is not to design new MOEA, but to use MOEAs to create radar phase coded waveforms.

Specifically, the goal is to find a variety of phase coded waveforms, of given length, suitable for use in modern radar systems. For a given length, there may or may not be a global optimal phase code. The existence of such codes are not known *a-priori*, and so this work assumes that globally optimum codes for multiple objectives do not exist. The MOEA, then, must find a set of Pareto optimal radar phase codes. Any experiment performed should evaluate the given MOEA's ability to generate a variety of good radar phase code waveforms.

What constitutes a good radar waveform, is an entire subject unto itself, not to be taken up in the current study. For this study, a good waveform, consists of a radar phase code that has a high autocorrelation mainlobe, and low autocorrelation sidelobes, as determined by the ISLR and PSLR [3]. The study focuses on a MOEA's ability to find a variety of different good radar phase codes, in a reasonable amount of time. After the algorithm completes execution, radar waveform quality will be assessed by examining the autocorrelation properties of a sample of waveforms from each algorithm's Pareto front.

There are a bevy of MOEA metrics and indicators available for evaluating MOEA performance. The literature identifies a set of good metrics that measure a MOEA's ability to meet the three main objectives of an MOEA [9]. Unfortunately, most of these indicators require that PF_{true} be known for metric evaluation. As is the case for most real world problems, the PF_{true} for the optimal radar phase code, is unknown, and thus many of the widely used metrics cannot be used. There are two metrics, however, that determine how well the algorithm maintains a diversity of points along the Pareto front, that do not require PF_{true} to be known. If the algorithm can be shown to create radar phase codes of good quality, then these metrics, Spacing and Overall Non-dominated Vector Generation (ONVG), are sufficient to indicate MOEA performance for the radar phase code problem.

As MOEA's are stochastic algorithms, statistical analyses are required for performance evaluation. For true statistical analysis, an infinite number of data runs is required. This is obviously not achievable, and researchers in general, accept any number of trials greater than 10 as being statistically significant. A more modern acceptable number of runs, due in part to the improved capabilities of computational hardware, is 30 [12]. This study, therefore, uses 30 experimental runs for each algorithm used on each problem.

Deciding what size problem to use for experimentation is the final step in the experimental design. To show true usefulness, the algorithms must determine phase codes of sufficient length to be usable in realized radar systems. Codes should also be of a length comparable to other studies, so that comparisons between the techniques can be made. To meet these criteria, this study uses the NSGA-II and SPEA2 algorithms to generate 64-chip length codes.

V. EXPERIMENTAL RESULTS

All experiments are performed on an Apple Computer Mac Pro, running two dual core Intel Xeon processors at 2.6 GHz,

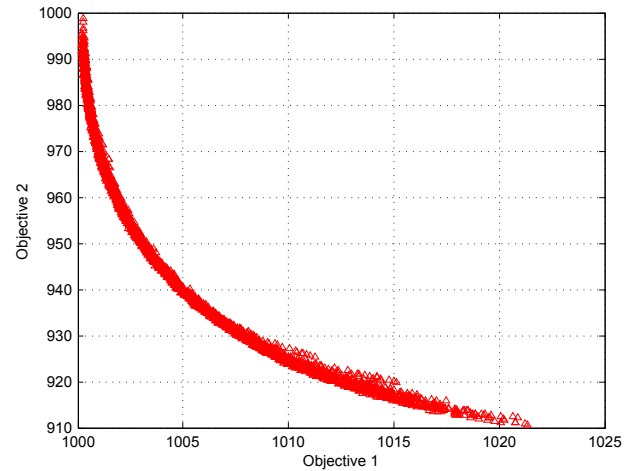


Fig. 2. Pareto Front Points Found by NSGA-II in all 30 Trials

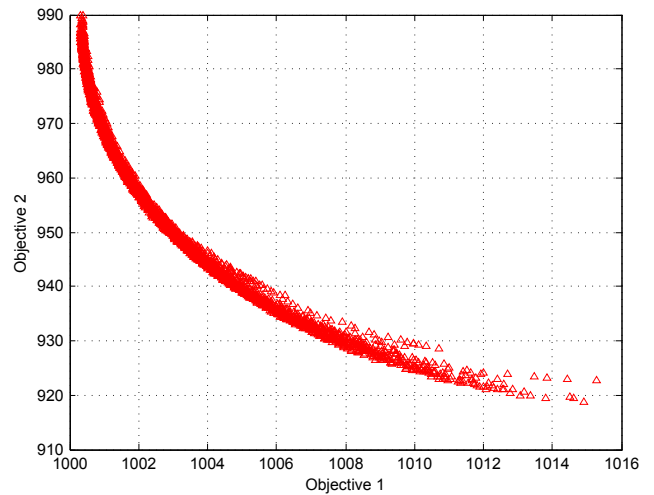


Fig. 3. Pareto Front Points Found by SPEA2 in all 30 Trials

and possessing 13 GB of RAM. The operating system is Mac OS10.6 Snow Leopard 64-bit. Computational software is MATLAB Release 2010b 64-bit. As MATLAB binaries are platform independent, it is expected that computational hardware with similar performance specifications should perform similarly, independent of operating system. Parallel processing was not specifically used, however, MATLAB was configured to use multi-threading where possible. The effects of activating or deactivating this MATLAB feature were not part of the study.

All studies are based upon Pareto front generation. Fig. 2 gives all solutions, in objective space, found by NSGA-II. Objective 1 is the probability of detection and Objective 2 is the CRLB for Doppler measurements, both given in (12). Fig. 3 gives the objective space solutions for SPEA2.

In both Figures, the MOEA's demonstrate their ability to find a diversity of points along the Pareto front. SPEA2 appears to

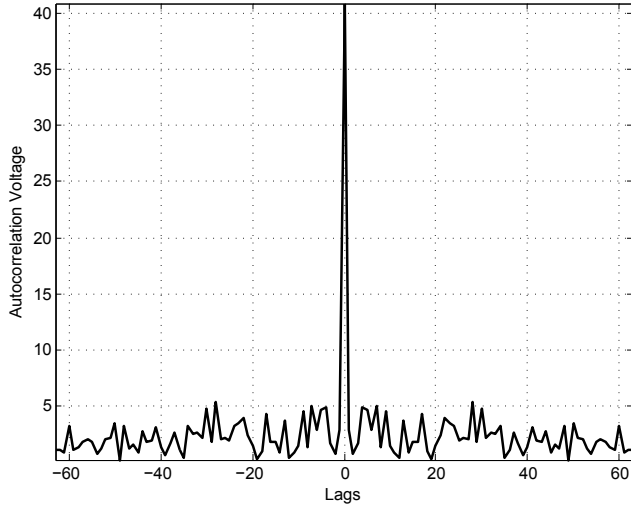


Fig. 4. Autocorrelation Function of a NSGA-II Determined Phase Code

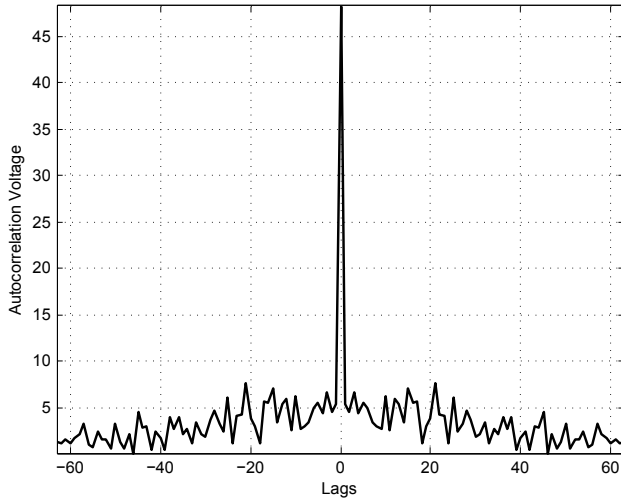


Fig. 5. Autocorrelation Function of a SPEA2 Determined Phase Code

have a more equal spacing of points along the Pareto front, which indicates a wider variety of phase codes to choose from. Given the similar Pareto fronts determined by both algorithms, however, both MOEAs should be able to determine phase codes with the desired properties. For both MOEAs, the algorithm ran for 500 iterations. This parameter was chosen to ensure both algorithms converged toward PF_{true} . Looking at the results, Fig. 3 has a tighter grouping of PF_{known} points than does Fig. 2. Accordingly, SPEA2 may require fewer iterations to converge toward the true PF than does NSGA-II. This comparison is left for future study.

Fig. 4 is the autocorrelation function of a representative phase code found by NSGA-II and Fig. 5 is the autocorrelation function of a phase code found by SPEA2. Both figures show a desirable autocorrelation in that the plot is of one main

TABLE I
SPACING AND ONVG METRICS FOR NSGA-II AND SPEA2

NSGA-II	mean	mode	std dev
Spacing	290.4584	243.8294	91.9716
ONVG	100	100	0.0
SPEA2	mean	mode	std dev
Spacing	312.0041	0.0	86.4211
ONVG	100	100	0.0

central spike with sidelobes well below the height of the main peak.

VI. ANALYSIS

One of the reasons that NSGA-II and SPEA2 are the MOEAs chosen, is that they both employ some sort of sharing/niching to ensure a diversity of points along the Pareto front. Given this feature, a small spacing value is expected for both algorithms. The question, as to whether, in general, NSGA-II or SPEA2 have better spacing values is not easily answered. From [10], results in the literature claim SPEA-2 is the better algorithm, while other published results indicate that NSGA-II is the better algorithm. Zitler, in the initial paper on SPEA2 [13], claimed that SPEA2 “provides a better distribution of points,” but does not clarify exactly what he means. Evidently, the performance of NSGA-II and SPEA2 are close enough, that their ranking is dependent upon the particular problem. Both algorithms are expected to have an ONVG close to the population size used. This is related to the spacing metric in that the algorithms should find a diversity of points along PF_{known} on each algorithm run, and therefore the points should be close to the Pareto front.

Table I contains a statistical analysis of the spacing and ONVG metrics for both NSGA-II and SPEA2 MOEA.

As expected, with respect to the ONVG metric, both MOEAs are statistically equivalent. Just considering the mean value and standard deviation for the spacing metric, would indicate similar performance of both algorithms. The mode, however, tells a different story. For NSGA-II, the mode is within one standard deviation of the mean, while for SPEA2 the mode is zero. This indicates that the majority of the time SPEA2 has equally spaced points on PF_{known} giving the spacing value of zero. This is as expected from the design of the algorithms [10].

Of more interest than MOEA metric performance to radar waveform designers, is the quality of the phase codes generated by the algorithms. As the MOP of (12) has been demonstrated in [4] to achieve high P_d and accurate Doppler measurements, this study focuses on the MOEAs ability to solve the MOP under the additional waveform constraints. Table II gives a statistical comparison of the ISLR and PSRLR for the phase codes determined by both NSGA-II and SPEA2. What is of interest, is that NSGA-II appears to provide better radar codes in terms of ISLR and PSRLR. The best codes determined by NSGA-II are better than those generated by SPEA2. Furthermore, the average values of both ISLR and

TABLE II
ISLR AND PSLR VALUES

SPEA2	max	min	mean	std-dev
PSLR	-15.4088	-20.0934	-17.9215	1.2047
ISLR	-2.2961	-4.8619	-3.6305	0.7154
NSGA-II	max	min	mean	std-dev
PSLR	-17.1787	-21.4020	-19.0789	1.2667
ISLR	-3.4906	-6.3799	-4.5400	1.0190

PSLR of all codes generated by NSGA-II were better than the codes generated by SPEA2.

The MOEA generated phase codes also compare favorably to codes generated through other search methods. The study done in [6], used a greedy local search technique to improve known phase codes. The [6] generated codes had a best case performance of -18.7 dB PSLR and -4.9 dB ISLR, when optimized over a single objective. This value is slightly better than the average of the SPEA2 generated codes, but below the average of the NSGA-II generated codes for both values. When the codes are optimized over both ISLR and PSLR, [6] gives a best ISLR value of -6.4 dB and a best PSLR of -30.2 dB. The MOEA generated waveforms have ISLR values equivalent to the best greedy derived phase codes, but have around 8 dB poorer performance in terms of PSLR.

It should be noted, that the MOEA derived waveforms have also been optimized for P_d , as well as Doppler accuracy. The waveforms generated in [6], were optimized strictly in terms of ISLR and PSLR. The MOEA generated waveforms have similar performance in terms of ISLR and PSLR, while having good baseline radar performance; other single objective optimization methods such as the greedy technique have no such performance guarantee. Furthermore, due to their inherently parallel nature MOEAs generate many phase codes with good overall performance as demonstrated in Table II. The greedy technique can only optimize a single waveform at a time.

VII. CONCLUSIONS AND FUTURE WORK

This work presented a technique, using Multiple-Objective Evolutionary Algorithms (MOEA), to create radar phase code waveforms with desirable autocorrelation function properties. Two MOEAs, the NSGA-II and SPEA2 algorithms, were used to develop 64-bit radar phase codes. Each MOEA was run 30 times, and a statistical analysis of the performance of each algorithm performed. Algorithm performance was characterized on the algorithm's ability to produce a diversity of points along the Pareto front, as determined by the spacing metric. For the length 64 phase code, the NSGA-II algorithm was determined to outperform the SPEA2 algorithm in terms of ISLR and PSLR values. Graphs of the Pareto curves for both algorithms demonstrated both to be capable of producing a diversity of points on PF_{known} that progress towards PF_{true} . Furthermore, it was demonstrated that the autocorrelation functions of the generated waveforms realized better ISLR and PSLR values than other code generating methods described in the literature.

Due to the limitation of available MATLAB MOEA implementations, only NSGA-II and SPEA2 were selected for analysis. Future work should consider at a minimum MOEA/D [14], as it is designed specifically to spread solutions along a Pareto front and should produce good spacing values. Another future research effort should include varying the parameters of the NSGA-II and SPEA2 algorithms to determine if better performance, vis-a-vie creating radar phase codes, can be accomplished through algorithm tuning. The number of generations each MOEA is allowed to run should be addressed as this criteria could produce a "best" MOEA for determining radar phase codes.

ACKNOWLEDGMENT

The authors would like to thank the Air Force Research Laboratory RYDR office for financial support as well as laboratory and equipment use.

The views expressed in this paper are those of the authors and do not reflect the official policy or position of the United States Air Force, the Department of Defense, or the United States Government

REFERENCES

- [1] Z. Michalewicz and D. B. Fogel, *How to Solve It: Modern Heuristics*, 2nd ed. Springer, 2004.
- [2] M. I. Skolnik, *Introduction to Radar Systems*, 3rd ed. McGraw Hill, 2001.
- [3] N. Levanon and E. Mozeson, *Radar Signals*. Wiley Interscience, 2004.
- [4] A. D. M. an dM. Piezzo, A. Farina, and M. Wicks, "Pareto-optimal radar waveform design," *IET Radar, Sonar and Navigation*, 2010.
- [5] X. Zeng, Y. Zhang, and Y. Guo, "Polyphase coded signal design for mimo radar using mo-micpso," *Journal of Systems Engineering and Electronics*, vol. 22, no. 3, June 2011.
- [6] J. Jakabosky, P. Anglin, M. R. Cook, S. D. Blunt, and J. Stiles, "Non-linear fm waveform design using marginal fisher's information within the cpm framework," *IEEE Radar Conference*, 2011.
- [7] S. M. Kay, *Fundamentals of Statistical Signal Processing - Estimation Theory*. Prentice Hall, 1993, vol. 1.
- [8] E.-G. Talbi, *Metaheuristics From Design to Implementation*. Wiley and Sons, 2009.
- [9] Coello, Lamont, and V. Veldhuizen, *Evolutionary Algorithms for Solving Multiobjective Problems*. Springer, 2007.
- [10] G. B. Lamont and M. Kleeman, "A comparison of nsga-ii and spea-2," Air Force Institute of Technology, Tech. Rep., 2008.
- [11] various, "Matlab central," internet.
- [12] A. G. S. of Engineering Department of Electrical and C. Engineering, "Performance evaluation of computational experiments," Air Force Institute of Technology, Tech. Rep., May 2010.
- [13] E. Zitler, M. Laumanns, and L. Thiele, "Spea2: Improving the strength pareto evolutionary algorithm," Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland, Tech. Rep. Technical Report 103, May 2001.
- [14] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712-731, December 2007.