COGNITIVE RADIOS WITH GENETIC ALGORITHMS: INTELLIGENT CONTROL OF SOFTWARE DEFINED RADIOS

Thomas W. Rondeau (Virginia Tech, Blacksburg, VA, USA; trondeau@vt.edu), Bin Le (Virginia Tech, Blacksburg, VA 24061, USA; binle@vt.edu), Christian J. Rieser (Johns Hopkins University Applied Physics Laboratory, Laurel, MD, 20723; christian.rieser@jhuapl.edu*), Charles W. Bostian (Virginia Tech, Blacksburg, VA, USA; bostian@vt.edu)

ABSTRACT

We can think of a cognitive radio as having three basic parts that make it cognitive: the ability to sense, including at a minimum sensing the RF spectrum, geographical surroundings, and the user's needs; the capacity to learn, ideally in both supervised and unsupervised modes; and finally, the capability to adapt within any layer of the radio communication system. At the Virginia Tech (VT) Center for Wireless Telecommunications (CWT), we have developed a cognitive radio engine to perform all of these tasks. This paper presents the adaptive component, which uses genetic algorithms (GAs) to evolve a radio defined by a chromosome. The chromosome's genes represent the adjustable parameters in a given radio, and by genetically manipulating the chromosomes, the GA can find a set of parameters that optimize the radio for the user's current needs. At the end of this paper, we present experimental results on both a hardware platform and software simulation.

1. INTRODUCTION

To take advantage of the flexibility afforded through adaptable radios, and in particular, software defined radios (SDR), we are developing a cognitive engine to provide intelligent control over an adaptable radio. An intelligent radio allows autonomous adaptation that can improve performance, enhance spectrum usage, and further advance wireless ubiquity. Instead of forcing a radio user to determine how the radio communicates, a cognitive radio will provide the user with a quality of service (QoS) without putting the burden directly upon the user.

The cognitive radio being developed at the CWT [1] [2] uses a sensing algorithm to read in information like propagation effects and presence of other radio users [3] [4] from the radio environment, learn the behavior of the radio in the different environments, and intelligently adapt the radio to new parameter settings efficiently to provide a user with the required QoS.

This paper concentrates solely on the adaptation mechanism, which is a method to adjust the radio parameters. The method uses a genetic algorithm (GA) to optimize the radio parameters on the physical (PHY) layer, and we are moving forward to include the link (or MAC), network, transport, and application layers. The genetic algorithm approach, called the Wireless System Genetic Algorithm (WSGA), is a powerful method to realize cross-layer optimization and a method of adaptive waveform control. This paper addresses the formulation of the WSGA and presents some experimental results.

In this paper, Section 2 covers the basic background of genetic algorithms and provides an overview of multi-objective genetic algorithms and their pertinence to the radio configuration problem. Section 3 describes the WSGA implementation. Section 4 presents experimental results using both a hardware platform and a simulation. Section 5 concludes the paper with some discussion of the future of cognitive radios.

2. MULTI-OBJECTIVE GENETIC ALGORITHMS

2.1. Genetic Algorithms – A Brief Review

In their original and most basic form, genetic algorithms (GAs) were designed as single-objective search and optimization algorithms. Common to all GAs is the chromosome definition—how the data are represented; the genetic operators of crossover and mutation; the selection mechanism for choosing the chromosomes that will survive from generation to generation; and the evaluation function used to determine the fitness of a chromosome. All of these operators are well described in [5]. Without reproducing the standard body of knowledge on GAs, we will present how we use these GA techniques in Section 3.

* Christian Rieser is currently employed at the Johns Hopkins University Applied Physics Laboratory. This work was performed as part of his Ph.D. research at Virginia Tech.

2.2. The Generic Multi-Objective Genetic Algorithm

Multi-objective problems analyze a number of, often-competing, objectives for optimization and decision making [6] – [10]. Zitzler [11] gives a brief but comprehensive overview of the concepts and literature of multi-objective problems and presents the basic formula for defining a multi-objective decision maker (MODM) as shown in (1).

$$\begin{aligned} &\min/\max\{\bar{y}\} = f(\bar{x}) = [f_1(\bar{x}), f_2(\bar{x}), \ldots, f_n(\bar{x})] \\ &\quad \textit{subject to}:\ \bar{x} = (x_1, x_2, \ldots, x_m) \in X \\ &\qquad\qquad \bar{y} = (y_1, y_2, \ldots, y_n) \in Y \end{aligned} \tag{1}$$

Where there are n dimensions to consider in the search space and $f_n(\bar{x})$ defines the mathematical function to evaluate dimension n. Both x, the set of input parameters, and y, the set of dimensions, may be constrained to some space, X and Y. The optimal solutions to MODMs lie on the *Pareto front*, which is the set of input parameters, x, that is non-dominated in any dimension, which is often a trade-off of goals [6] [10] [11].

GAs are well suited to multi-dimensional decision problems due to the parallel evaluation in many dimensions to find global optima as well as the ability to include constraints about the problem [6] [8] [11] [12]. A GA used in MODM problems is a multi-objective genetic algorithm (MOGA).

In effective wireless communications, the choice of the radio parameters on all layers affects the radio's behavior in many dimensions such as bit error rate (BER), bandwidth, power consumption, and network latency, to name just a few. Each of these dimensions has some relationship to the QoS, and these relationships change in their relative importance depending on the application being used. For example, a user transferring a large file would care most about BER and data rate, but a user holding a video conference would care more about network latency and jitter. Since these goals often compete with each other, as in minimizing a BER and minimizing power at the same time, the radio design problem is a MODM, which makes a MOGA a powerful algorithmic approach to autonomously adapting a radio.

With a radio, the user has some desirable operation that values certain goals more than others, such as the minimum latency requirement of a video conference. Although many different methods haven been proposed in the literature [11], we associate each optimization dimension in the radio with a weight to delineate the relative importance of the goals in the decision-making process. As the MOGA analyzes each dimension, optimization in the higher-weighted dimensions leads to a solution tailored to the user's preferences.

Different selection and evaluation methods have been proposed for MOGAs [8] [10] [13]. Many methods try to combine the evaluations along the different dimensions into a single metric [9]; this method breaks down in cases where the values of the dimensions can vary greatly in magnitude (BER of 10^{-6} versus data rate of 10^{6}), and normalizing each dimension requires a great deal of domain knowledge, which might be difficult to get [11] or may be changing. Other methods involve competition between population members and incrementing the fitness function of the winner for each objective dominated by the winning member [8] [13]. Horn [14] extends this idea to oppose two individuals against a larger pool of chromosomes from the population; the individual who wins the most competitions is deemed better fit and survives to the next generation.

3. WIRELESS SYSTEM GENETIC ALGORITHM

In order to communicate successfully, the radio must first be configured to fit the specific channel condition, such as a cellular fading channel or an interference-prone unlicensed channel; second, the radio must support user required service types like voice or data; third, sitting on top of everything the radio does are the regulatory requirements the radio must obey to operate legally in any band and geographic location. To combine all these issues effectively and provide the best performance trade-off, the radio needs to be aware of its environment; in other words, the radio needs a cognitive engine to analyze the physical link, user demands, and regulatory regimes, and it must balance multiple objectives and constraints. As stated above, genetic algorithms are well suited to solving multi-objective optimization and decision problems, which is why we have chosen to work with a MOGA in order to control the radio's adaptive process.

The wireless system genetic algorithm (WSGA) is a MOGA designed for the control of a radio by modeling the physical radio system as a biological organism and optimizing its performance through genetic and evolutionary processes. In the WSGA, radio *behavior* is interpreted as a set of PHY and MAC layer operation parameters defined by *traits* encapsulated in the *genes* of a *chromosome*. Other general radio functional parameters (such as payload size, antenna configuration, voice coding, encryption, equalization, retransmission requests, and spreading technique/code) are also identified as possible genes in the chromosome definition to allow for future growth through each layer of the radio communication's stack. The chromosome shown in Fig. 1 represents the PHY-layer traits currently of consequence to the WSGA due to current hardware limitations and the current state of the simulation.

Power	frequency	Pulse Shape	Symbol Rate	Modulation

Fig. 1. Representation of a chromosome for GA manipulation.

The WSGA analyzes the chromosome's fitness through a set of fitness functions defined by performance evaluations of the current radio channel. Each fitness function is weighted to represent the relative importance the user has associated with each objective. The Pareto front therefore moves so that the optimal solution provides the most efficient performance for the user's QoS requirements under combinatorial constraints. Here, efficiency and optimization mean providing a QoS without over-maximizing, which may waste radio resources such as spectrum and power. For example, a user sending email does not need a 100 Mbps link with a 30 dB carrier to noise ratio.

The fitness evaluation functions are designed to reflect the current link quality of both PHY and MAC layers, which currently include the mean transmitting power, data rate, BER, packet error rate (PER), spectral efficiency, bandwidth, interference avoidance, packet latency and packet jitter. One of the most powerful attributes of the WSGA is the dynamic fitness definition and evaluation, where not only is the weighting of each function adjustable, but any fitness function may or may not be used as required by the current link conditions and user requirements. All functions are dynamically linked from a database so that they can be dynamically added and weighted into the fitness evaluation for a specific link condition and performance objectives. Such dynamic adjustment of the fitness evaluation is directed by some higher-layer intelligence such as the learning machine in the cognitive engine which conducts the evaluation of the overall radio system and network performance.

In general, fitness functions associate to specific channel conditions. For example, the fitness function for determining a BER is channel specific. Likewise, the function weights associate to a specific user's desire for network performance, e.g. a desire to improve the BER while sacrificing the data rate. Although these associations hold for a general-purpose analysis of the WSGA, the user can influence the functions, such as the data rate objective, and the channel can also influence the weights for each function, such as increasing the BER weighting to compensate for a particularly poor channel.

We use a relative tournament selection method similar to [8], except that the fitness of the winner from a single comparison is scaled by the weight associated with that fitness function. After all the single comparisons in all dimensions, the winning member is the one with the highest fitness, and that one survives to the next population. While this does not guarantee that all winners are the best, or non-dominated, members of the population (only better relative to its combatant), it maintains species diversity within the population while still pushing towards the Pareto front. Diversity in the population allows different solutions to be tried and helps prevent the algorithm from getting stuck in a local optimum. We did not apply Horn's [14] method of fighting two individuals against a subset of the population because his method, while he claims it produces better results, calls for a larger population and more fitness comparisons. This becomes computationally intensive and is undesirable in a real-time optimization system.

Crossover and mutation are simple implementations of these mechanisms. Crossover is performed on a single point chosen as a uniform random number with a static probability of crossover occurring. Mutation is also a single point operation chosen from a uniform random number with a static probability of crossover occurring. Future enhancements to the WSGA call for an adaptive adjustment of crossover and mutation probabilities as well as the population size during the optimization process for higher convergence efficiency and accuracy.

The ability to apply constraints to the optimization problem as shown in (1) gives us the opportunity to incorporate regulatory and physical restrictions during chromosome evolution. If a trait determined by the chromosome exceeds the limits of the radio's capabilities, like finding a center frequency outside the tunable range of the radio, or breaks the law, like transmitting too much power in a specific band, then the WSGA penalizes those chromosomes. We have chosen to use a penalty approach similar to that outlined in [7] by setting the fitness evaluation of that chromosome to zero, basically nullifying its chance to survive to the next generation.

The final issue to realize the operation of the WSGA is the exchange of optimized chromosomes between radios wishing to communicate such that all networked radios evolve to have better traits. Provided a communications link already exists, all radios on the network running the WSGA will share the chromosomes and the most fit chromosome is elected as the winner (as though this were the last generation of the genetic algorithm). The radios will then all switch their parameters together. If no communications path is present, a control channel can be set up to allow temporary communication between the radios in order to exchange chromosomes and reconfigure themselves. We recognize that this is not the most complete or satisfactory solution for all situations, and we will need to adopt some protocol to establish the connections and exchange the adaptation information between all radios on the network.

4. EXPERIMENTAL RESULTS

4.1. Hardware Experiments

The experiments are done in both simulation and on a real hardware platform. For all experiments, both hardware and simulation, the GA parameters were set as listed in Table 1.

Table 1. WSGA Genetic Parameters

Parameter	Value
Crossover Rate	90 %
Mutation Rate	5 %
Population Size	30
Replacement Size	20
Max Generations	50

The radios we used were Proxim *Tsunami* radios, which have limited adaptability as shown in Table 2.

Table 2. Proxim *Tsunami* Adaptable Parameters

Parameter	Range
Frequency	5730 – 5820 MHz
Power	6 – 17 dBm
Modulation	QPSK, QAM8, QAM16
Coding	Rate 1/2, 2/3, 3/4
Time division duplex (TDD)	29.2% - 91% BSU to SU

These values are further restricted in that there are only up to six discrete 20.75 MHz wide channels, and the coding is only adjustable for QPSK modulation between rates 1/2 and 3/4. Because of the limited number of adaptable parameters in the Proxim radios, we limited the fitness evaluation to BER minimization, data rate maximization, and power minimization.

Even with this severely limited adaptable radio, we are able to see the benefits of the WSGA and hints of the power of the GA adaptation mechanism. The tests we ran were to see how well the radios could adapt to a user's needs and a known co-channel interferer by running a packet error rate (PER) test. To determine the ideal performance of the radios, we ran the PER test in a line of sight (LOS) path with no interference for each modulation (experiments LOS-1 – LOS-4).

The first test was to maximize the data rate for the user at the base station (experiment Data-5), which started with the radio set to its lowest data rate and ran the WSGA to find the maximum data rate with minimum power. The second test (experiment Int-6) used a bounce path off a building between the test base station unit (BSU) and subscriber unit (SU), where we used a second Proxim *Tsunami* BSU as an interferer on an overlapping channel as shown in Fig. 2. During the tests, we forced the radios always to use the same channels so that the interference avoidance could not be accomplished by simply switching channels (the easy solution). The radio configurations before the WSGA are summarized in Table 3 and all experiments used the test network RF channel in Fig. 2.

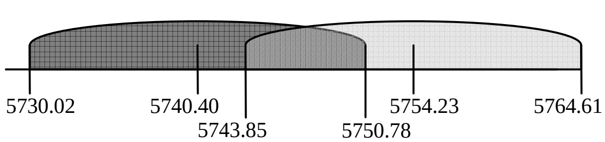

Fig. 2. Spectrum of test network (light gray) and interfering radio (dark gray). Frequency is in MHz.

Table 3. Experimental Configurations

Radio Settings	Experiment					
	LOS-1	LOS-2	LOS-3	LOS-4	Data-5	Int-6
Modulation	QAM16	QAM8	QPSK	QPSK	QPSK	QAM16
Power (dBm)	6	6	6	6	17	17
Coding	3/4	2/3	3/4	1/2	1/2	3/4
TDD (%)	50	50	50	50	50	50

The PER tests for experiments LOS-1 through LOS-4 and Int-6 are shown in Table 4. The PER tests were not applicable to experiment Data-5 where the WSGA was just optimizing the data rate and power because the same channel conditions and radio placements as in experiments LOS-1 through LOS-4. During experiment Data-5, we initialized the radio to the slowest bit rate, which is 10 Mbps for both up and down links, and the power was set to the maximum level, 17 dBm. When running the WSGA, both minimizing power and maximizing data rate were weighted with equal weights of 255, the maximum. The WSGA's solution to this problem is to set the modulation to QAM16 with a coding of 3/4 (there is no other choice), the TDD to 91%, and the power to 6 dBm. The WSGA successfully found the minimum possible power and the maximum possible data rate of the radios, 55 Mbps for the downlink and 5 Mbps for the uplink.

For experiment Int-6 with the interferer, we initially set the BER minimization fitness weight to 200, the power minimization function weight to 210, and the data rate maximization rate to 0. The results of the WSGA was a configuration of 7 dBm transmit power, QPSK with 1/2 rate coding, and a TDD scheme of 75%. Table 4 summarizes the data collected for experiment Int-6 along next to the results of experiments LOS-1 through LOS-4. As this table shows, the results after we ran the WSGA are better than before, but are still not that great. We then reset the radios to the previous configuration and re-ran the WSGA with different weights. This time, the BER minimization was set to 255 while the other fitness functions were set to 0. The WSGA produced a radio configuration of QPSK with coding rate ¾, 17 dBm of power, and 50% TDD. Table 4 shows greatly improved PER performance.

These experiments show that the WSGA works with real hardware to accomplish real goals. However, it is also obvious from these experiments and results that the hardware platform of the Proxim *Tsunamis* does not allow the WSGA's power to really show through because of the limited adaptability of the radios.

4.2. Simulation of an SDR PHY Layer

To test the full power of the WSGA in cognitive radio, we developed a simulation in MatLab. In this software radio testing bed, we can adjust the power, center frequency, modulation (type and order), pulse shape filtering characteristics (PSF roll-off factor and filter order), and symbol rate. These adjustable parameters and their ranges are shown in Table 5. WSGA uses them as the primary PHY-layer parameters from which a set of secondary PHY-layer parameters can be directly derived, such as channel bandwidth, bit rate and bit energy. We set them to mid-range values for illustration, but as an SDR platform model, these parameters can be set to any values according to specific radio environments.

Table 4. Experimental Results

Radio Settings		Experiment						
		LOS-1	**LOS-2**	**LOS-3**	**LOS-4**	**Int-6: Pre GA**	**Int-6: Post GA 1**	**Int-6: Post GA 2**
BSU–SU	Packets	19991	10000	10000	20000	9999	10000	10000
	Lost Packets	14	0	18	8	209	2	10
	PER	$7x10^{-4}$	0	$1.8x10^{-3}$	$8x10^{-4}$	$2.09x10^{-2}$	$2x10^{-4}$	$1x10^{-3}$
SU–BSU	Packets	10000	10000	10000	20000	5526	7278	10000
	Lost Packets	17	0	98	0	4754	3459	1
	PER	$1.7x10^{-3}$	0	$9.8x10^{-3}$	0	0.8603	0.4752	$1x10^{-4}$

Table 5. Simulation Adaptable Parameters

Parameter	Range
Power (dBm)	0 – 30
Frequency (MHz)	2400 – 2480
Modulation	M-PSK, M-QAM
Modulation, M	2 – 64
PSF roll-off factor	0.01 – 1
PSF order	5 – 50
Symbol Rate (Msps)	1 – 20

The simulation's enhanced function flexibility enables a much larger set of applicable fitness evaluations leading to more creative solutions so that the radio can achieve more complicated objectives such as minimize BER, minimize bandwidth, maximize spectral efficiency, minimize power, maximize data rate, and minimize interference (taken and given), individually or collectively.

We ran three experiments to test the effectiveness and power of the WSGA in SDR: minimize spectral occupancy for applications like text messaging and email, maximize throughput for broadband video, and avoid interferers in a situation such as the ISM band where WiFi devices in channel 1, 6, and 11 are used as interferers. The fitness function weighting is shown in Table 6, which is representative of weightings that represent fairly standard requirements from a radio.

The configuration results from the WSGA for each scenario are shown in Table 7. Fig. 3 shows the frequency domain representation of the signals for each scenario.

Table 6. Simulation Test Conditions

Functions	Weights		
	Minimize spectral occupancy	Maximize throughput	Interference avoidance
BER	255	100	200
BW	255	10	255
Spectral Efficiency	100	200	200
Power	225	10	200
Data Rate	100	255	100
Interference	0	0	255

As Fig. 3 and Table 7 show, the WSGA successfully created solutions of radio parameter sets by balancing the combinatorial objectives. The most interesting results were from the interference experiment, where the results of two runs are shown in Figs. 3c and d. The Pareto front is a tight balance of the weightings associated with different objectives, which is typically not stabile between different GA processes because it can be dominated by any one or more highly weighted objectives. Avoiding interference does dominate, but not always completely successfully, as in the case of Fig. 3d, which found a good center frequency, but too large of a bandwidth. These results show that the WSGA can now adjust a radio, but it might seem that we have just displaced the control burden from the radio knobs to the WSGA weights; however, this is desirable as the weights directly affect the radio performance, which an intelligent and learning machine can easily manipulate in the place of a human user. The learning machine can adjust the weights and learn about the success of those weightings through feedback from the radio as well as from higher layers, and even positive or negative stimuli from the user as corrective measures.

Table 7. Simulation Test Results

Radio Parameter	Weights			
	Minimize spectral occupancy (a)	Maximize throughput (b)	Interference avoidance (c)	(d)
Power (dBm)	18	28	29	23
Frequency (MHz)	2440	2430	2436	2436
Symbol Rate (Msps)	1	18	3	8
PSF roll-off	0.05	0.33	0.04	0.04
PSF order	46	20	18	13
Modulation, type	PSK	QAM	PSK	QAM
Modulation, M	2	16	4	8
BER	0	0	0	0.12
Data Rate (Mbps)	1	72	6	24

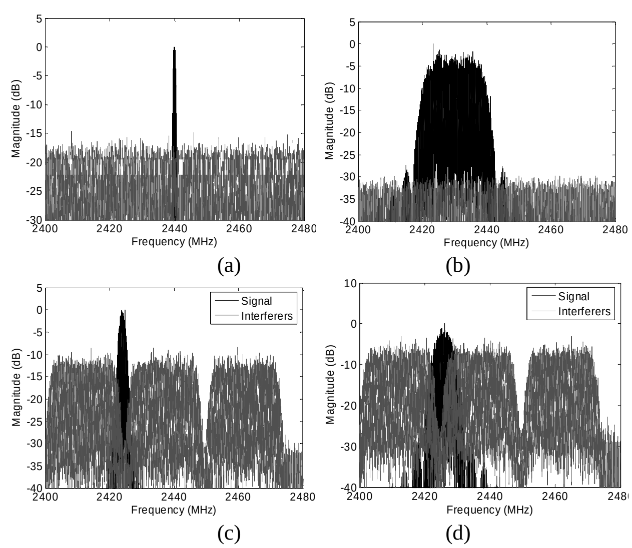

Fig. 3. Spectrum Plots for scenarios of Table 6.

5. CONCLUSIONS

While the results of the WSGA prove its success and usefulness in controlling radio parameters to provide a user with different QoS's, much work remains. The current GA itself works, but many other techniques could greatly improve its effectiveness and convergence speed such as adaptive GAs [15], migration and niching techniques along the Pareto front [5] [6] [8], and analyzing the tournament selection proposed by [14].

Another benefit of the GA is the ability to distribute the work across many processors [12]. We wish to distribute the WSGA among many different radios within the same network so each can parallel process to find a better solution faster.

With the cognitive radio, we plan to incorporate a learning machine to automatically adjust the weights and determine the fitness functions that the WSGA should use to optimize the radio given feedback from all layers of the system and the user. As we then tie this work into sensing techniques, we will have a fully-realized cognitive radio.

6. ACKNOWLEDGEMENTS

This work was supported by the National Science Foundation through Digital Government (award 9983463) and IGERT (award 9987586) programs.

7. REFERENCES

[1] C. J. Rieser, "Biologically Inspired Cognitive Radio Engine Model Utilizing Distributed Genetic Algorithms for Secure and Robust Wireless Communications and Networking," *Ph.D. Dissertation*, Virginia Tech, Aug., 2004.

[2] C. W. Bostian, S. F. Midkiff, T. M. Gallagher, C. J. Rieser, and T. W. Rondeau. "Rapidly Deployable Broadband Communications for Disaster Response," *Sixth International Symposium on Advanced Radio Technologies (SAFECOM Session)*. Boulder, CO, March 2-4, 2004 (Proceedings pp. 87-92).

[3] T. M. Gallagher, "Characterization and Evaluation of Non-Line-of-Sight Paths for Fixed Broadband Wireless Communications," *Ph.D. Dissertation*, Virginia Tech, Jun., 2004.

[4] T. W. Rondeau, C. J. Rieser, T. M. Gallagher, and C. W. Bostian, "Online Modeling of Wireless Channels with Hidden Markov Models and Channel Impulse Responses for Cognitive Radios," *IEEE Proc. IMS*, 2004, pp. 739 – 742.

[5] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, MA, 1989.

[6] C. M. Fonseca and P. J. Fleming, "Genetic Algorithms for multiobjective optimization: formulation, discussion, and generalization," *Proc. Int. Conf. Genetic Algorithms*, pp. 416 – 423, 1993.

[7] C. M. Fonseca and P. J. Fleming, "Multiobjective optimization and multiple constraint handling with evolutionary algorithms. I. A unified formulation," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 28, pp. 26 – 37, 1998.

[8] H. Lu and G. G. Yen, "Multiobjective optimization design via genetic algorithm," *IEEE Proc. International Conference on Control Applications*, pp. 1190 – 1195, Sept. 2001.

[9] C. Hwang and A. Syeed, *Multiple Objective Decision Making – Methods and Applications*, Springer-Verlag, New York, 1979.

[10] P. Fleming, "Designing control systems with multiple objectives," *IEE Colloquium Advances in Control Technology*, pp. 4/1 – 4/4, May, 1999.

[11] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms - a comparative case study and the strength Pareto approach," *IEEE Trans. Evolutionary Computation*, vol. 3, pp. 257 – 271, 1999.

[12] T. Hiroyasu, M. Miki, and S. Watanabe, "Distributed genetic algorithms with a new sharing approach in multiobjective optimization problems," *IEEE Proc. Congress on Evolutionary Computation*, vol. 1, pp. 69 – 76, Jul, 1999.

[13] J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," *Proc. Int. Conf. Genetic Algorithms*, pp. 93 – 100, 1985.

[14] J. Horn, N. Nafpliotis, and D. E. Goldberg, "A Niched Pareto Genetic Algorithm for Multiobjective Optimization," *IEEE Proc. World Congress on Computational Intelligence*, 1994, vol. 1, pp. 82 – 87.

[15] M. Srinivas and L. M. Patnaik, "Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, No. 4, pp. 656-666, 1994.