Non-Standard Crossover for a Standard Representation -- Commonality-Based Feature Subset Selection

Stephen Chen
The Robotics Institute
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
chens@ri.cmu.edu
http://www.cs.cmu.edu/~chens

César Guerra-Salcedo
Department of Computer Science
Colorado State University
Fort Collins, CO 80523
guerra@cs.colostate.edu

Stephen F. Smith
The Robotics Institute
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
chens@ri.cmu.edu
http://www.cs.cmu.edu/~sfs

Abstract

The Commonality-Based Crossover Framework has been presented as a general model for designing problem specific operators. Following this model, the Common Features/Random Sample Climbing operator has been developed for feature subset selection--a binary string optimization problem. Although this problem should be an ideal application for genetic algorithms with standard crossover operators, experiments show that the new operator can find better feature subsets for classifier training.

1 INTRODUCTION

A classification system is used to predict the decision class of an object based on its features. When training a classifier, it is beneficial to use only the features *relevant* to prediction accuracy, and to ignore the *irrelevant* features [Koh95]. The benefit arises from an increase in the "signal-to-noise ratio" of the data, and a reduction in the time required to train the classifier. Thus, the objective of feature subset selection is to identify the (most) relevant features.

Feature subset selection easily fits the standard (binary string) representation for genetic algorithms. For each feature, a '1' causes the feature to be used in training (i.e. identifies it as a relevant feature), and a '0' causes the feature to be ignored (as irrelevant). Since the standard representation allows standard crossover operators to be used, it has been argued that this "eliminates the need for designing new genetic operators" [VdJ93]. However, it has also been argued that it is "essential ... to incorporate ... local improvement operators" [SG87] to make a competitive genetic algorithm (GA). In this paper, a non-standard crossover operator is developed for feature subset selection.

The Commonality-Based Crossover Framework presents a new general model for the design of problem specific operators. Specifically, it defines crossover as a two-step process: 1) preserve the maximal common schema of two parents, and 2) complete the solution with a construction heuristic [CS98][CS99a]. The model follows from the commonality hypothesis which suggests that schemata common to above-average solutions are above-average. For feature subset selection, this hypothesis implies that the relevant features can be identified by observing which (selected) features are common to good solutions.

Common schemata are preserved by all standard crossover operators. However, 1's and 0's are treated equally. In feature subset selection, the 1's (potentially relevant features) may be more informative than the 0's. Therefore, selected features (1's) are chosen to form the basis of commonality. Next, the Random Sample Climbing (RSC) heuristic, a "constructive" local improvement operator based on mu-lambda evolution strategies (ES) [FOW66][Sch81], is developed. Using the Common Features (CF) partial solution to restart RSC, the Common Features/Random Sample Climbing (CF/RSC) crossover operator is defined.

Experiments have been conducted on the new crossover operator, and on the RSC local improvement operator alone. On average, the CF/RSC solutions have the smallest feature subsets, and these subsets lead to the fewest testing

errors. It is possible that larger subsets lead to over-fitting of the training data. Further, comparing CF/RSC with RSC, it can be observed that (adaptive) commonality-based restarts may improve the effectiveness of ES-based methods. Comparing CF/RSC to standard GAs, the benefit of the Commonality-Based Crossover Framework for standard representations is demonstrated.

The remainder of this paper is presented as follows. In section 2, an overview of feature subset selection is presented. In section 3, the Random Sample Climbing local improvement operator is developed. In section 4, the data sets and basic experimental parameters are described. In section 5, results for Random Sample Climbing are presented. In section 6, the Common Features/Random Sample Climbing crossover operator is developed, and its results are presented in section 7. The results are discussed in section 8, and final conclusions are summarized in section 9.

2 BACKGROUND

Building a classification system involves two distinct parts. The first part is to select a classifier. For this paper, the classifier is Euclidean Decision Tables (EDT) [GW98] [GCW99]. Second, the feature subset selection (optimization) problem is isolated by using the "wrapper" approach [Koh95]. The EDT classifier is "wrapped" into the search method by using it as the objective function. For each candidate feature subset, an EDT classifier is trained and evaluated. This evaluation function is based first on having the fewest classification errors, and then on having the fewest features. However, the overall goal is to find the feature subset that minimizes the number of classification errors during final testing[1].

The usefulness of the wrapper approach has been demonstrated using basic search methods like bit climbing [Koh95]. However, most of the data sets in the original study had small feature spaces (less than 30). The effectiveness of search strategies is better tested on large problems. This paper uses the LandSat data set (36 features), the DNA data set (180 features), and the Cloud data set (204 features). For standard crossover operators, Guerra-Salcedo & Whitely [GW98] have shown that CHC [Esh91] performs better than GENESIS [Gre84], and that the performance difference is most pronounced for the largest (Cloud) data set.

[1] The "best" feature subset during training can over-fit the initial training data, and thus it may not be the best during final testing.

The offspring of standard crossover operators inherit all of the common 1's and common 0's from their parents. If the commonality hypothesis is valid[2], the common 1's should identify relevant features and the common 0's should identify irrelevant features. However, there can also be "weakly relevant" features. If these features are unselected (0) in both parents, it may be inappropriate to hypothesize that these features are irrelevant and to necessarily exclude them from the offspring. Therefore, standard crossover, as described by Convergence Controlled Variation--"allele values that are not present in the two parents cannot be introduced into the offspring" [EMS96], may be poorly suited for this (standard representation) problem.

To determine which weakly relevant features are desirable, "One possibility is to estimate which features are strongly relevant, and start the search from this subset ..." [Koh95]. Using the commonality hypothesis to identify the strongly relevant features, search should be (re)started from the common 1's. It has been suggested that bit climbing be used to conduct these searches [Koh95]. However, preliminary experiments with bit climbing were unpromising. Thus, the following method was developed instead.

3 A CONSTRUCTION HEURISTIC FOR FEATURE SUBSET SELECTION

Bit climbing is a form of hill climbing. However, probabilistic search methods (e.g. simulated annealing [KGV83] and tabu search [Glo89]) often perform better than hill climbing. The Lin-Kernighan (variable-depth) heuristic [LK73] has similarities to a mu-lambda (μ-λ) evolution strategies approach with μ equal to one. Combining these observations, Random Sample Climbing was developed for feature subset selection.

Random Sample Climbing (RSC) is initialized with a seed solution of all 0's--no features selected. From the seed solution, λ new solutions are created. The new solutions are each created by randomly selecting "samples" with up to n bits (with replacement) and mutating them. If the best of the new solutions is better than the previous seed solution, the seed solution is replaced[3]. Overall, k generations of λ solutions each are created.

When RSC starts with only a few (or no) features selected, mutations tend to increase the number of features[4]. Thus,

[2] Results presented in [CS99b] validate the commonality hypothesis--schemata common to above-average solutions are indeed above average.

[3] The initial seed solution (all 0's) is not evaluated. It is always replaced by the best solution of the first generation.

RSC acts like both a construction heuristic and a local improvement operator. The use of mutations facilitates backtracking by allowing features to be de-selected. With n greater than one, multiple mutations allow escapes from one-bit neighborhood local optima. The use of populations ($\lambda > 1$) helps reduce negative effects caused by the greedy/ myopic nature of the search process.

4 THE DATA SETS

All data sets are taken from the UC Irvine repository[1]. The LandSat data set has 4435 training instances and 2000 testing instances. The DNA data set has 2000 training instances and 1186 testing instances. For these data sets, candidate feature subsets have classifiers trained on 400 instances and evaluated on 700 instances (both drawn without replacement from the training instances). The best feature subset then has a final classifier trained on all of the training instances and tested on all of the testing instances.

The Cloud data set has 1633 total instances--no preset definition of training and testing instances is made. Thus, during search, classifiers are trained on 400 instances and evaluated on 500 instances, both drawn without replacement. But, final testing is done by 10-fold cross validation. For each "fold", 60% of the data set is used for classifier training, and 40% is used for testing. The average testing errors for 10 random folds are reported.

[4] RSC does not easily climb to solutions with more than half of the available features selected.

[1] http://www.ics.uci.edu/~mlearn/MLRepository.html

5 RESULTS: RANDOM SAMPLE CLIMBING

RSC has been tested on the above data sets. For all experiments, n is set to 3 and λ is set to 30. To match previous experiments [GW98], 15,000 total evaluations (trained classifiers) are allowed. The available evaluations have been distributed into one run of (k equals) 500 generations, 10 runs of 50 generations, 25 runs of 20 generations each, and 50 runs of 10 generations. Each parameter set has been run on 30 different pairs of training and evaluation sets. (See Table 1.)

The range of results with RSC tend to be highest for one run of 500 generations. In one run, RSC can act like a depth-first search. Similarly, to use more runs of fewer generations makes a trade-off between the breadth and the depth of the search. It appears that breadth (more runs) may improve the worst trial, but the lack of depth (generations) may negatively affect the best trial. Overall, the best balance between breadth and depth occurs with 10 runs of 50 generations each.

Table 1: Results for RSC on LandSat, DNA, and Cloud data sets. Best, worst, and average values are for 30 independent trials. The best balance of breadth and depth is for 10 runs of 50 generations.

			Train				Test		
Data Set	Runs	k	errors			average subset size	errors		
			best	worst	average		best	worst	average
LandSat	1	500	52	95	78.1	14.8	220	289	242.4
	10	50	50	91	**75.0**	14.2	217	267	**242.2**
	25	20	55	86	75.6	13.5	216	289	246.7
	50	10	55	95	77.2	12.9	223	270	246.7
DNA	1	500	47	114	80.8	18.1	80	203	158.4
	10	50	60	87	**73.0**	12.0	80	174	**134.2**
	25	20	62	93	78.0	11.4	84	168	138.8
	50	10	73	100	83.5	10.1	111	167	140.5
Cloud	1	500	95	134	110.5	15.6	121.4	170.6	145.4
	10	50	82	119	**102.7**	19.3	114.1	159.8	132.9
	25	20	88	160	109.8	24.8	92.0	168.6	**126.7**
	50	10	93	125	108.2	18.1	117.3	158.6	133.0

Table 2: Results for CF/RSC on LandSat, DNA, and Cloud data sets. Best, worst, and average values are for 30 independent trials. Performance is quite robust across parameter settings, but (5, 10-90) has a slight advantage in finding smaller subsets and better "best test errors".

				Train				Test		
Data Set	GA parameters			errors			average subset size	errors		
	k	RSC	CF/RSC	best	worst	average		best	worst	average
LandSat	20	4	21	52	92	73.9	13.6	215	315	247.1
	10	7	43	53	90	73.8	13.4	215	275	245.2
	5	10	90	54	93	75.1	12.8	205	313	247.7
DNA	20	4	21	45	69	54.9	9.5	74	140	98.9
	10	7	43	46	68	55.8	9.3	74	126	99.3
	5	10	90	45	72	57.4	8.5	74	142	104.1
Cloud	20	4	21	85	115	99.7	18.4	116.2	143.3	130.6
	10	7	43	86	118	99.3	18.2	113.5	151.2	128.4
	5	10	90	90	119	102.7	14.0	101.9	141.0	124.7

6 A HEURISTIC OPERATOR FOR FEATURE SUBSET SELECTION

The above results analyze RSC as a local search/local improvement operator. However, RSC can also be viewed as a construction heuristic. The Commonality-Based Crossover Framework presents a design model that allows the effectiveness of construction heuristics to be amplified [CS99b]. Essentially, commonality-based selection identifies partial solutions with high proportions of fit schemata. "Back-tracking" to this promising restart location, a new solution is (heuristically) rebuilt.

For feature subset selection, selected features (1's) are chosen as the basis of commonality. The commonality hypothesis suggests that these features are strongly relevant. Thus, the Common Features partial solution is used to initialize RSC, which is used to search for additional (weakly relevant) features. Overall, Common Features/Random Sample Climbing (CF/RSC) is presented as a problem specific crossover operator for feature subset selection[1].

[1] The Common Features seed solution is not evaluated. This avoids the cost of an evaluation, and it forces RSC to take an initial step of exploration.

Table 3: Results for CF/RSC and CHC on LandSat, DNA, and Cloud data sets. Best, average, and standard deviation values are for 30 independent trials.

		Train				Test		
Data Set	Algorithm	accuracy (% correct)			average subset size	accuracy (% correct)		
		best	average	std. dev.		best	average	std. dev.
LandSat	CF/RSC (5, 10-90)	92.3	89.3	1.08	12.8	**89.8**	87.6	1.01
	CHC	86.8	85.3	0.81	12.6	88.9	87.6	1.25
DNA	CF/RSC (5, 10-90)	93.6	91.8	0.89	8.5	**93.8**	**91.2**	1.33
	CHC	96.4	95.1	0.51	11.2	92.5	89.4	2.13
Cloud	CF/RSC (5, 10-90)	82.0	79.5	1.68	14.0	**84.4**	**80.9**	1.67
	CHC	84.8	82.2	1.37	42.1	80.8	79.3	1.95

7 RESULTS: COMMON FEATURES/ RANDOM SAMPLE CLIMBING

To keep the total number of evaluations constant at 15,000, the choice for GA parameters is quite constrained. Keeping n and λ as control variables, they are again set to 3 and 30 respectively. This allows the product of k (generations) and the total number of RSC and CF/RSC solutions to be 500. Values of 20, 10, and 5 have been used for k. The corresponding GAs are allowed to have 25, 50, and 100 total solutions. The populations of size 4, 7, and 10 are initialized with RSC, and the remaining solutions are generated by CF/RSC. The experiments are run using GENITOR [WS90] with a selection bias of 1.00 (random parent selection) and with duplicate solutions disallowed.

Compared to RSC alone, the results with CF/RSC tend to be better overall. (See Table 2.) They also appear to be more robust across parameter settings. Commonality-based restarts of ES may provide the benefits of both breadth and depth, without sacrificing either.

Compared to CHC [GCW99] (which produces the best results for standard crossover operators), the feature subsets found by CF/RSC tend to perform better during final testing. (See Table 3.) The CF/RSC subsets also tend to be much smaller than the CHC subsets[1]. Thus, as the time to train an EDT classifier grows approximately with the square of the number of features, CF/RSC also has a significant time advantage over CHC. On average, CF/RSC takes 50-90% less time than CHC to evaluate 15,000 solutions. Lastly, on the DNA and Cloud data sets, the smaller CF/RSC feature subsets perform significantly worse during training, but they tend to perform better during testing. (See Table 4.)

Table 4: T-tests -- are the CF/RSC and CHC results different?

Data Set	Training	Testing
LandSat	Yes	No
DNA	Yes	Yes
Cloud	Yes	No

8 DISCUSSION

This paper presents Common Features/Random Sample Climbing, a non-standard crossover operator for feature subset selection. The experimental results with CF/RSC can be examined with respect to standard GAs, the ES-style RSC operator alone, and machine learning concepts.

8.1 GENETIC ALGORITHMS

The Commonality-Based Crossover Framework presents a new model for the design of (heuristic) crossover operators. The primary goal of this paper is to demonstrate the benefit that a novel design perspective can provide, even for standard representations. It appears this goal has been achieved--CF/RSC identifies better subsets for classifier training than standard crossover operators.

8.2 RESTARTS AND LOCAL SEARCH

For greedy, deterministic local optimization techniques, restarts allow fair time-based comparisons to be made with other (non-deterministic) methods. And, for these techniques, the benefit of commonality-based restarts has been previously demonstrated [Boe96][CS99a]. Comparing the results of CF/RSC to those for RSC alone, it can be observed that (stochastic/probabilistic) local search methods may also benefit from commonality-based restarts.

For the same operators, tabu search should be able to back-track out of local-minima wells faster than simulated annealing. Tabu search can "march" out in linear time if all the backward moves are tabu. However, simulated annealing can take exponential time with respect to the number of back-tracking steps required. Restarts can eliminate back-tracking altogether. However, it can still take significant time to bring a random solution into the near-optimal region [UPvL91]. Commonality-based restarts are more efficient because the restart solution (consisting of strongly relevant features) is already in the near-optimal region.

8.3 MACHINE LEARNING

It has been suggested that it is desirable to train classifiers with small feature subsets. In addition to speed, small subsets allow for better intuitive interpretations of the data [Koh95]. Comparing the results of CF/RSC and CHC, another advantage is suggested: greater accuracy. The smaller subsets of CF/RSC tend to do worse in training, but better in testing. Large feature subsets (like those found by CHC) may lead to over-fitting of the training data.

[1] Further, the initial subsets used to (re)start RSC are even smaller.

9 CONCLUSIONS

The Commonality-Based Crossover Framework presents a new model for the design of problem specific (heuristic) crossover operators. By following this model, Common Features/Random Sample Climbing is developed for feature subset selection--a problem that naturally fits the standard (binary string) representation. Experimental results demonstrate that CF/RSC can find better feature subsets for classifier training than standard crossover operators.

Acknowledgments

The work described in this paper was sponsored in part by the Advanced Research Projects Agency and Rome Laboratory, Air Force Material Command, USAF, under grant numbers F30602-95-1-0018 and F30602-97-C-0227, and the CMU Robotics Institute. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Advanced Research Projects Agency and Rome Laboratory or the U.S. Government.

César Guerra-Salcedo is a visiting researcher at Colorado State University supported by CONACyT under registro No. 68813 and by ITESM.

References

[Boe96] K.D. Boese. (1996) *Models for Iterative Global Optimization*. Ph.D. thesis, University of California at Los Angeles, 1996.

[CS98] S. Chen and S.F. Smith. (1998) "Experiments on Commonality in Sequencing Operators." In *Genetic Programming 1998: Proceedings of the Third Annual Conference*.

[CS99a] S. Chen and S.F. Smith. (1999) "Putting the "Genetics" back into Genetic Algorithms (Reconsidering the Role of Crossover in Hybrid Operators)." To appear in *Foundations of Genetic Algorithms 5*, W. Banzhaf and C. Reeves, eds. Morgan Kaufmann.

[CS99b] S. Chen and S.F. Smith. (1999) "Introducing a New Advantage of Crossover: Commonality-Based Selection." In these proceedings.

[EMS96] L.J. Eshelman, K.E. Mathias, and J.D. Schaffer. (1996) "Convergence Controlled Variation." In *Foundations of Genetic Algorithms 4*, R. Belew and M. Vose, eds. Morgan Kaufmann.

[Esh91] L.J. Eshelman. (1991) "The CHC Adaptive Search Algorithm: How to Have Safe Search When Engaging in Nontraditional Genetic Recombination." In *Foundations of Genetic Algorithms*, G. Rawlins, ed. Morgan Kaufmann.

[FOW66] L.J. Fogel, A.J. Owens, and M.J. Walsh. (1966) *Artificial Intelligence through Simulated Evolution*. Wiley.

[GCW99] C. Guerra-Salcedo, S. Chen, L.D. Whitley, and S.F. Smith. (1999) "Fast and Accurate Feature Selection Using Hybrid Genetic Strategies." To appear in *CEC99: Proceedings of the Congress on Evolutionary Computation*.

[Glo89] F. Glover. (1989) "Tabu Search--Part I." In *ORSA Journal on Computing*, 1:190-206, 1989.

[Gre84] J. Grefenstette. (1984) "GENESIS: A System for Using Genetic Search Procedures." In *Proceedings of a Conference on Intelligent Systems and Machines*.

[GW98] C.Guerra-Salcedo and L.D. Whitley. (1998) "Genetic Search for Feature Subset Selection: A Comparison Between CHC and GENESIS." In *Genetic Programming 1998: Proceedings of the Third Annual Conference*.

[Koh95] R. Kohavi. (1995) *Wrappers for Performance Enhancement and Oblivious Decision Graphs*. Ph.D. thesis, Stanford University, 1995.

[KGV83] S. Kirkpatrick, C.D. Gelatt, Jr., and M.P. Vecchi. (1983) "Optimization by Simulated Annealing." In *Science*, 220:671-680, 1983.

[LK73] S. Lin and B.W. Kernighan. (1973) "An Efficient Heuristic Algorithm for the Traveling Salesman Problem." In *Operations Research*, 21:498-516, 1973.

[Sch81] H.-P. Schwefel. (1981) *Numerical Optimization of Computer Models*. Wiley.

[SG87] J.Y. Suh and D. Van Gucht. (1987) "Incorporating Heuristic Information into Genetic Search." In *Proc. Second International Conference on Genetic Algorithms and their Applications*.

[UPvL91] N.L.J. Ulder, E. Pesch, P.J.M. van Laarhoven, H.-J. Bandelt, E.H.L. Aarts. (1991) "Improving TSP Exchange Heuristics by Population Genetics." In *Parallel Problem Solving from Nature*, R. Männer and H.-P. Schwefel, eds. Springer-Verlag.

[VdJ93] H. Vafaie and K. de Jong. (1993) "Robust Feature Selection Algorithms." In *Proceedings of the International Conference on Tools with AI*.

[WS90] L.D. Whitley and T. Starkweather. (1990) "GENITOR II: A distributed Genetic Algorithm." In *Journal of Experimental and Theoretical Artificial Intelligence*, 2:189-214, 1990.