# Sensible Initialization of a Computational Evolution System Using Expert Knowledge for Epistasis Analysis in Human Genetics

Joshua L. Payne, Casey S. Greene, Douglas P. Hill, and Jason H. Moore

**Abstract.** High throughput sequencing technologies now routinely measure over one million DNA sequence variations on the human genome. Analyses of these data have demonstrated that single sequence variants predictive of common human disease are rare. Instead, disease risk is thought to be the result of a confluence of many genes acting in concert, often with no statistically significant individual effects. The detection and characterization of such gene-gene interactions that predispose for human disease is a computationally daunting task, since the search space grows exponentially with the number of measured genetic variations. Traditional artificial evolution methods have offered some promise in this problem domain, but they are plagued by the lack of marginal effects of individual sequence variants. To address this problem, we have developed a computational evolution system that allows for the evolution of solutions and solution operators of arbitrary complexity. In this study, we incorporate a linkage learning technique into the population initialization method of the computational evolution system and investigate its influence on the ability to detect and characterize gene-gene interactions in synthetic data sets. These data sets are generated to exhibit characteristics of real genome-wide association studies for purely epistatic diseases with various heritabilities. Our results demonstrate that incorporating linkage learning in population initialization via expert knowledge sources improves classification accuracy, enhancing our ability to automate the discovery and characterization of the genetic causes of common human diseases.

## 1 Introduction

Recent technological advances have allowed for inexpensive and dense mappings of the human genome, making genome-wide association studies (GWAS) a standard

Joshua L. Payne · Casey S. Greene · Douglas P. Hill · Jason H. Moore
Computational Genetics Laboratory, Dartmouth Medical School, 1 Medical Center Drive, Lebanon, NH, USA
e-mail: {Joshua.L.Payne,Casey.S.Greene,Douglas.P.Hill,
Jason.H.Moore}@Dartmouth.edu

Y.-p. Chen (Ed.): Exploitation of Linkage Learning, ALO 3, pp. 215–226.
springerlink.com

form of analysis in the detection of common human disease. The goal of GWAS is to identify genetic markers that differ significantly between diseased and healthy individuals, through a comparison of allele frequencies at specific loci. One commonly employed genetic marker is the single nucleotide polymorphism (SNP), which is a single location in the genome that varies between people. To provide sufficient coverage of the human genome for GWAS, it is estimated that over one million SNPs have to be considered [8], and samples of this size are now readily provided by high-throughput technologies. However, analyses of these data have rarely identified single sequence variants that are predictive of common human disease. Given the robustness and complex structure of metabolic and proteomic networks [18], it is reasonable to assume that such monogenic diseases are the exception, not the rule, and that many diseases are caused by two or more interacting genes. Such gene-gene interactions, or epistasis, dramatically increase the difficulty of using GWAS to uncover the genetic basis of disease [13]. For one million candidate SNPs, there are $5 \times 10^{11}$ pairwise combinations and $1.7 \times 10^{17}$ three-way combinations. For higher order interactions, the number of possible combinations is enormous. A major charge for bioinformatics is to develop efficient algorithms to navigate through these astronomical search spaces, in order to detect and characterize the genetic causes of common human disease.

Due to the combinatorial nature of this problem, algorithms designed to discover gene-gene interactions in GWAS will need to rely on heuristics. Methods that employ exhaustive search will not be feasible. Statistical and machine learning techniques, such as neural networks [10], have been applied in this problem domain, but have only proven successful for cases with a small number of SNPs. Alternative approaches, such as multifactor dimensionality reduction [17] and random chemistry [3], have also shown promise, though they are similarly limited to data sets with only a small number of SNPs. Artificial evolution techniques, such as genetic programming, have been investigated in this problem domain, but they have had limited success because individual SNPs often show little or no marginal effects, and as such, there are no building blocks for evolution to piece together. However, recent results have demonstrated that the inclusion of expert knowledge, such as information gained from feature selection methods, can be used to bias such nature-inspired classification algorithms toward SNPs that are suspected to play a role in disease predisposition [6, 7, 14, 11, 15].

One source of expert knowledge that has proven useful in this domain is a family of machine learning techniques referred to as Relief [5, 9, 16]. These algorithms are able to detect SNPs that are associated with disease via independent or main effects, although they cannot provide a model of the genetic architecture of disease. However, the information provided by Relief can be used to supply artificial evolution with the building blocks needed to successfully generate such an architectural model. For example, improvements in classification power have been obtained by using Relief variants to bias mutation operators [6] and population initialization [7] in genetic programming. Such feature selection techniques are a form of linkage learning, where potential interactions between SNPs are inferred and subsequently exploited to bias evolutionary search.

Though classical artificial evolution methods, such as genetic programming, have shown promise in this problem domain if guided by expert knowledge [6, 7, 14, 11, 15], it has been suggested that the inclusion of a greater degree of biological realism may improve algorithm performance. Specifically, Banzhaf et al. [1] have called for the development of computational evolution systems (CES) that embrace, and attempt to emulate, the complexity of natural systems. To this end, we have developed a hierarchical, spatially-extended CES that includes evolvable solution operators of arbitrary complexity, population memory via archives, feedback loops between archives and solutions, hierarchical organization, and environmental sensing. In a series of recent investigations [4, 7, 11], this system has been successfully applied to epistasis analysis in GWAS for human genetics.

Here, we investigate the inclusion of linkage learning via sensible initialization in CES for the detection of epistatic interactions in GWAS. Specifically, we develop an expert-knowledge-aware initialization method that uses the feature weights provided by a machine learning technique to bias the selection of attributes for the initial population. We compare this initialization method to both random and enumerative initialization on synthetic data sets generated to exhibit representative characteristics of GWAS.

## 2 Computational Evolution System

In order to directly infer the influence of the initialization strategy on algorithm performance in the absence of other confounding effects, we use a simplified version of the computational evolution system (CES) discussed in [11]. In this section, we describe the CES as it is employed in this study.

In Fig. 1, we provide a schematic diagram of the system. Solutions are organized on a lattice at the bottom layer of the hierarchy, where competition between solutions occurs locally among adjacent lattice sites (Fig. 1D). At the second layer of the hierarchy is a lattice of solution operators of arbitrary size and complexity, which are used to modify the solutions (Fig. 1C). At the third layer, is a lattice of mutation operators that modify the solution operators (Fig. 1B). At the fourth layer is the mutation frequency, which governs the rate at which the mutation operators are modified (Fig. 1A).

### *2.1 Solution Representation, Evaluation, and Selection*

Solutions are represented using stacks, where each element in the stack consists of a function and two input arguments (Fig. 1D). The function set contains $+,-,*,/,\%,<,\leq,>,\geq,==,\neq$, where % is a protected modulus operator. The input arguments are SNPs.

Each solution produces a real valued output $S_i$ when applied to an individual $i$ in a SNP data set. These outputs are used to classify individuals as healthy or diseased using symbolic discriminant analysis (SDA) [12], as follows. The solution is applied to all healthy individuals in the data set and a distribution of outputs $S^{healthy}$

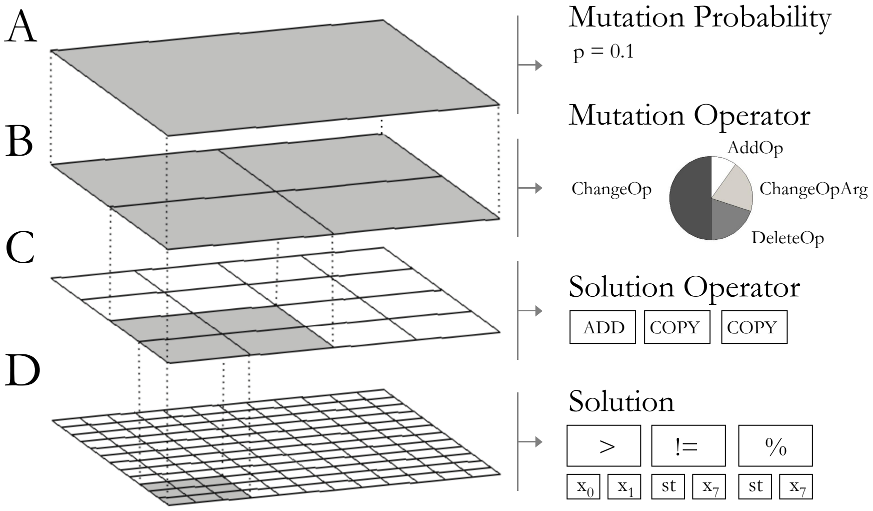

**Fig. 1** Schematic diagram of the simplified computational evolution system considered in this study. The hierarchical lattice structure is shown on the left and specific details of each layer are provided on the right. At the lowest level (D) is a two-dimensional toroidal lattice of solutions, where each lattice cell contains a single solution. Solutions are represented using stacks. In the above example, the Boolean output of $x_0 > x_1$ will be tested for inequality with $x_7$ via the stack (denoted by st) and this Boolean result will be an operand of the modulus operator (again, via st). At the second level (C) is a grid of solution operators that each consist of some combination of the building blocks ADD, ALTER, COPY, DELETE, and REPLACE. The top two levels of the hierarchy (A and B) generate variability in the solution operators. The experiments considered herein used a solution lattice of 32 × 32 cells. A 12 × 12 lattice is shown here for visual clarity

is recorded. Similarly, the solution is applied to all diseased individuals in the data set and a distribution of outputs $S^{diseased}$ is recorded. A classification threshold $S_0$ is then calculated as the arithmetic mean of the medians of the $S^{healthy}$ and $S^{diseased}$ distributions. The classification rule then assigns an individual $i$ healthy status if $S_i > S_0$ and diseased status if $S_i \leq S_0$.

The classification rule of a given solution can be used to calculate the number of true positives ($TP$), false positives ($FP$), true negatives ($TN$), and false negatives ($FN$), through a comparison of the predicted and actual clinical endpoints. This information can then be used to calculate a measure of solution accuracy

$$A = \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right). \quad (1)$$

The fitness $f$ of a solution is given by its accuracy, weighted by solution length to encourage parsimony

$$f = A + \frac{\alpha}{L}, \quad (2)$$

where $L$ is the number of elements in the solution stack and $\alpha$ is a tunable parameter (for all experiments considered here, $\alpha = 0.001$).

The population is organized on a toroidal, two-dimensional lattice where each solution resides in its own cell. Selection is synchronous and occurs within spatially-localized, overlapping neighborhoods. Specifically, each solution competes with those solutions residing in the eight surrounding cells (Moore neighborhood) and the solution with the highest fitness is selected to repopulate that cell for the next generation. Reproduction occurs using the evolvable solution operators described in the next section.

### 2.2 Solution Operators

One of the simplifying assumptions of traditional artificial evolution methods is that genetic variation is introduced via point mutations and linear recombination events. However, the variation operators of biological systems are myriad, with insertions, deletions, inversions, transpositions, and point mutations all occurring in concert. In order to better mimic these salient features of natural systems, our CES allows for the evolution of variation operators of arbitrary complexity. This is achieved by initializing the solution operator lattice (Fig. 1C) with five basic building blocks, ADD, ALTER, COPY, DELETE, and REPLACE, which can be recombined in any way to form new operators.

These operators work as follows. ADD places a new function and its arguments into the focal solution stack. ALTER randomly chooses an element of the focal solution stack, and mutates either the function or one of its input arguments. COPY inserts a random element of the focal solution stack into the stack of a randomly chosen neighboring solution. DELETE removes an element from the focal solution stack and REPLACE extracts a sequence of random length from a neighboring solution stack and overwrites a randomly chosen sequence of the focal solution stack with that information.

In the extended version of CES [11], each solution operator also has an associated vector of probabilities that determine the frequency with which functions and attributes are modified at random, via expert knowledge sources, or archives. In the simplified CES considered here, all modifications occur at random.

Similar to the solutions, the solution operators reside on a two-dimensional lattice (Fig. 1C). However, the granularity of the solution operator lattice is more coarse than the solution lattice, such that each solution operator is assigned to operate on a $3 \times 3$ sub-grid of solutions. The solution operators are also under selective pressure, and are assigned a fitness score based on how much change they evoke in the solutions they control [11]. Competition among solution operators occurs locally in a manner similar to the competition among solutions.

### 2.3 Mutation Operators

The solution operators are modified by mutation operators that reside in the third layer of the hierarchy (Fig. 1B). The granularity of this lattice is further coarsened, with each cell controlling one quarter of the solution operator lattice below. We consider four mutation operators. The first (DeleteOp) deletes an element of a solution operator. The second (AddOp) adds an element to a solution operator. The third (ChangeOp) mutates an existing element in a solution operator. The fourth (ChangeOpArg) alters the probability vectors associated with a solution operator. (In the simplified CES considered herein, this is a null operation.)

A four-element vector is used to store the probabilities with which each mutation operator is employed (Fig. 1B). These probabilities undergo mutation at a rate specified by the highest level in the hierarchy (Fig. 1A). The probability vectors of the four lattice cells are in competition with one another, with fitness assessment analogous to the solution operators.

## 3 Population Initialization

We consider three forms of population initialization. In each case, all initial solutions begin as a single, randomly chosen function with two input arguments, which can subsequently evolve into arbitrarily complex functional forms. The selection of the initial input arguments varies between the three methods. The first form of initialization is the approach taken in most artificial evolution systems, where the population is initialized at random. In CES, this entails choosing the attributes for each initial function with uniform probability from all available attributes, with replacement.

The second initialization method attempts to maximize diversity in the population, by ensuring that all attributes are represented at least once. This enumerative initializer works by selecting attributes at random from the pool of all attributes, without replacement, until all attributes have been selected. The attribute pool is then refreshed and the process continues until all initial solutions possess their required input arguments.

The third initialization method capitalizes on the expert knowledge gained from a member of the Relief family of machine learning algorithms. This algorithm is referred to as Spatially Uniform ReliefF (SURF) [5], an extension of Tuned ReliefF [16] that has proven effective in detecting interacting SNPs in GWAS with noisy data sets and small interaction effects. In brief, SURF provides weights to each SNP based on how likely that SNP is to be predictive of disease. Weights are adjusted by iteratively selecting individuals that are within a specified similarity threshold, and then increasing the weights of common SNPs if these individuals have different disease status or decreasing their weights by the same amount if the individuals have the same disease status. These SNP weights are then used to bias the selection of attributes in the expert-knowledge-aware initialization function. Each attribute is selected with probability proportional to its weight, with the caveat that the same attribute cannot be included twice in the same function.

## 4 Data Simulation

The artificial data sets considered in this study were generated to exhibit pure epistasis (i.e., no marginal effects) and specific heritabilities, where heritability is defined as the proportion of disease cases attributable to genetic effects. We consider heritabilities of 0.025, 0.05, 0.1, 0.2, 0.3, and 0.4. For each heritability, we created five two-locus penetrance functions according to the method described in [2], and from each penetrance function we generated 100 data sets. Each data set consists of an equal number of diseased (800 cases) and healthy (800 controls) individuals and possesses 1000 SNPs. Of the 1000 SNPs, only two are predictive of disease and the other 998 are generated at random to exhibit no correlation with clinical endpoint, other than by chance alone.

## 5 Experimental Design

To facilitate a fair comparison between the three initialization methods, we ensure that for each replicate the same functions are used to seed all three initial populations. Specifically, for each cell in the solution lattice we choose a random initial function to place in that cell. These initial functions are held constant across the three initialization methods; only the selected attributes differ.

To assess the performance of CES using each initialization method, we report (i) the evolutionary dynamics of the best training accuracy and (ii) the testing accuracy obtained using the best model found by CES. The latter is calculated by applying the best model found during training to another data set generated for that particular penetrance table. Thus, for each heritability, we have 500 independent training and testing pairs. Both training and testing accuracy are calculated using Eq. 1.

## 6 Results and Discussion

In Fig. 2 we depict the evolutionary dynamics of the best training accuracy for the CES using random, enumerative, and expert-knowledge-aware initializers, for the six heritabilities considered in this study. For all heritabilities, the best training accuracy found in the initial population was highest when expert-knowledge-aware initialization was used (in each panel of Fig. 2, compare the height of the symbol types at generation zero). The random and enumerative initializers produced initial populations with nearly identical best training accuracies.

In most cases, the CES improved the training accuracy of the models supplied in the initial population by each of the initialization methods. For example, the insets of Fig. 2 depict the distributions of improvements in training accuracy obtained by the CES using the expert-knowledge aware initializer. The distributions are always bimodal, with one peak at zero and another centered between 0.05 and 0.15. The lower mode indicates that in some cases, the CES is unable to improve upon the best solution provided in the initial population. However, the higher mode indicates that in the majority of cases, some improvement in training accuracy is observed

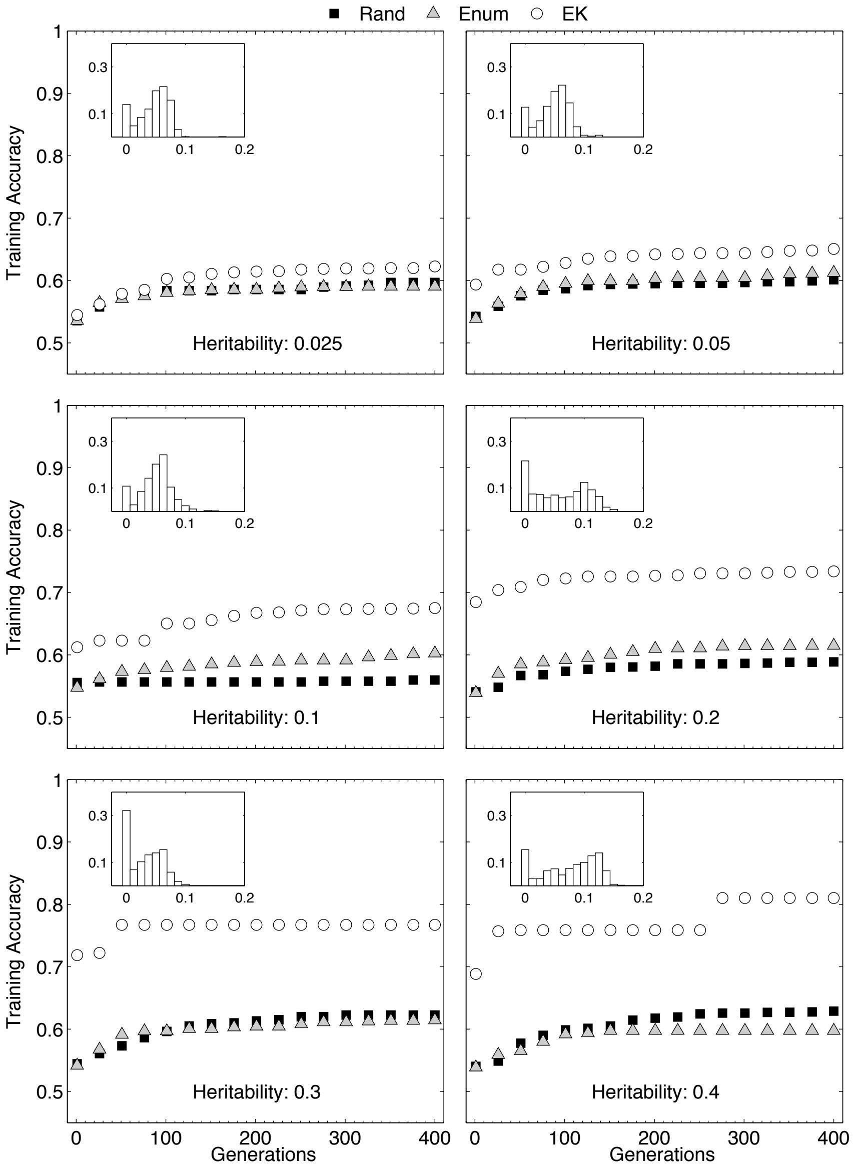

**Fig. 2** Evolutionary dynamics of best training accuracy for CES using random (black squares), enumerative (gray triangles), and expert-knowledge-aware (open circles) initializers for the six heritabilities considered in this study. The data presented in each panel correspond to a single replicate. The insets depict the distributions of improvements in training accuracy for CES with expert-knowledge-aware initialization, measured as the difference between the training accuracy at generation 0 and 400

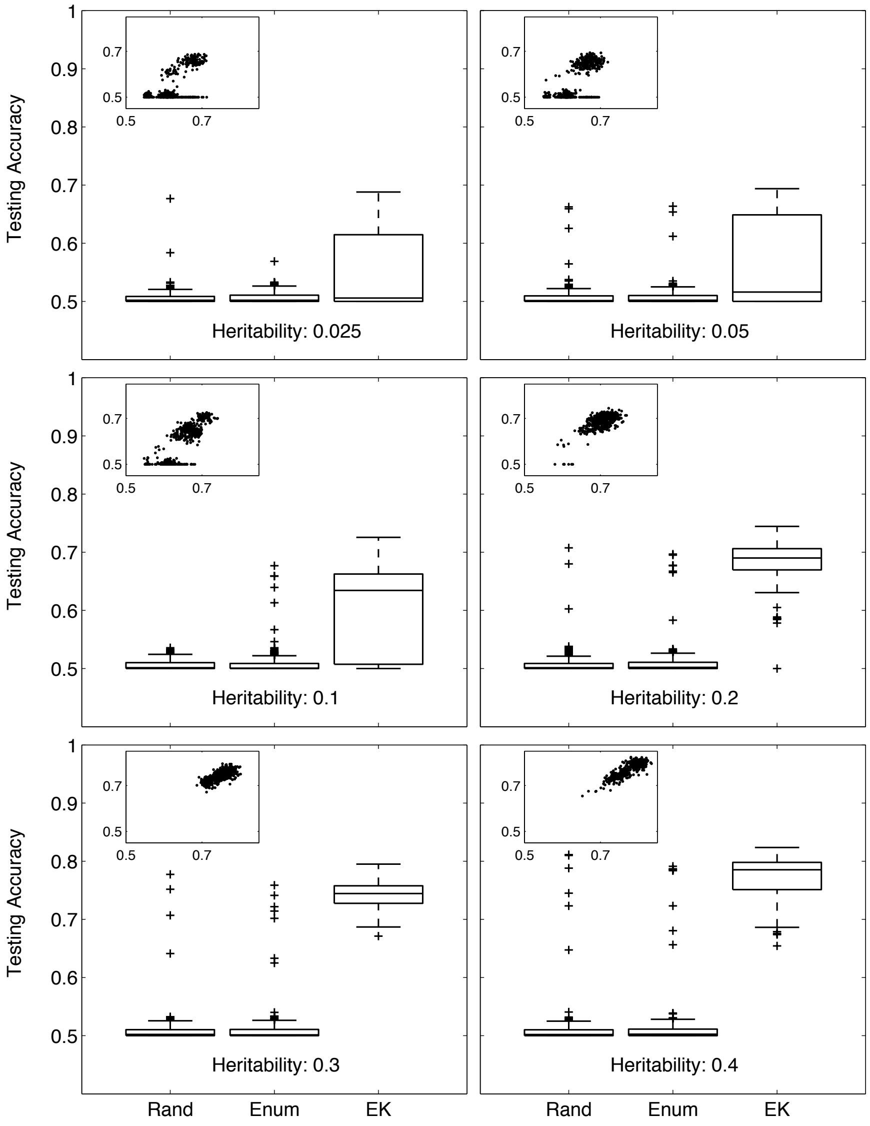

**Fig. 3** Testing accuracy of the best models found by CES using the random (Rand), enumerative (Enum), and expert-knowledge-aware (EK) initialization methods, for the six heritabilities considered in this study. The insets depict the testing accuracy (y-axis) as a function of the training accuracy (x-axis) for CES with expert-knowledge-aware initialization, across the 500 data sets considered for each heritability

using the CES. This indicates that SURF is able to correctly identify SNP linkage and that CES can exploit this information to build an architectural model of genetic predisposition to disease.

Using the expert-knowledge-aware initializer, the best training accuracy found in the initial population generally increased with increasing heritability. In contrast, using the random and enumerative initialization methods, the best training accuracy of the initial population remained approximately constant across heritabilities. These observations stem from the frequency with which the three methods supplied the two target SNPs to the initial population. Of the 500 replicates considered for each heritability, the percentage of trials in which the two interacting SNPs were correctly identified within a single solution by the expert-knowledge-aware initializer increased linearly from 41% at a heritability of 0.025 to 100% at a heritability of 0.2 ($r^2 = 0.99$). For heritabilities greater than or equal to 0.2, the two target SNPs were always identified within a single solution. Using the random and enumerative initializers, less than 1% of all trials contained the two target SNPs in a single solution, a figure that remained consistent across heritabilities.

In Fig. 3, we depict the testing accuracies of the best solutions obtained by the CES, using random, enumerative, and expert-knowledge-aware initialization. For all heritabilities, the testing accuracies of the best solutions found using the expert-knowledge-aware initializer were significantly higher than those obtained using either random or enumerative initialization. Following the trends of the training data (Fig. 2), the testing accuracies obtained with expert-knowledge-aware initialization increased as heritability increased, whereas the testing accuracy of the random and enumerative methods remained consistently low. The insets of Fig. 3 depict the testing accuracy of the best solution found by CES with expert-knowledge-aware initialization, as a function of its training accuracy. For low heritabilities, the data is clustered into two distinct groups, one in which testing accuracy is not correlated with training accuracy and another in which testing accuracy is positively correlated with training accuracy. As heritability increases, the data begin to migrate toward the cluster that exhibits positive correlation between testing and training accuracy, indicating a reduction in overfitting.

## 7 Concluding Remarks

We have investigated the influence of population initialization on the ability of a computational evolution system (CES) to detect epistatically interacting single nucleotide polymorphisms (SNP) in genome-wide association studies (GWAS). Our results demonstrate that the CES finds solutions of higher quality, both in terms of training and testing accuracy, when the population is initialized using an expert knowledge source than when it is not. Specifically, we found that biasing the selection of attributes in the initial population using a machine learning algorithm called Spatially Uniform ReliefF (SURF) [5] is superior to both random and enumerative initialization schemes.

These results complement those presented in [7], where it was shown that expert-knowledge-aware population initialization can improve the classification power of genetic programming for detecting gene-gene interactions in GWAS. Taken together, these results further highlight the critical need for expert knowledge sources in this problem domain [15]. Alternative approaches to incorporating expert knowledge sources, such as their inclusion in fitness assessment, selection, and mutation have also proven valuable [6, 14, 15]. Future work will investigate the combination of these expert-knowledge guided operators with expert-knowledge-aware initialization. Of particular interest is the utilization of alternative sources of expert knowledge, such as the causal information provided by metabolic and proteomic interaction networks. The incorporation of the many available sources of expert knowledge into artificial and computational evolution systems offers the potential to improve our ability to detect and characterize the genetic causes of human disease.

## References

1. Banzhaf, W., Beslon, G., Christensen, S., Foster, J.A., Képès, F., Lefort, V., Miller, J.F., Radman, M., Ramsden, J.J.: From artificial evolution to computational evolution: a research agenda. Nature Reviews Genetics 7, 729–735 (2006)
2. Culverhouse, R., Suarez, B.K., Lin, J., Reich, T.: A perspective on epistasis: limits of models displaying no main effect. American Journal of Human Genetics 70(2), 461–471 (2002)
3. Eppstein, M.J., Payne, J.L., White, B.C., Moore, J.H.: Genomic mining for complex disease traits with 'random chemistry'. Genetic Programming and Evolvable Machines 8, 395–411 (2007)
4. Greene, C.S., Hill, D.P., Moore, J.H.: Environmental noise improves epistasis models of genetic data discovered using a computational evolution system. In: Proceedings of the Genetic and Evolutionary Computation Conference, pp. 1785–1786 (2009)
5. Greene, C.S., Penrod, N.M., Kiralis, J., Moore, J.H.: Spatially uniform ReliefF (SURF) for computationally-efficient filtering of gene-gene interactions. BioData Mining 2(5) (2009)
6. Greene, C.S., White, B.C., Moore, J.H.: An expert knowledge-guided mutation operator for genome-wide genetic analysis using genetic programming. In: Rajapakse, J.C., Schmidt, B., Volkert, L.G. (eds.) PRIB 2007. LNCS (LNBI), vol. 4774, pp. 30–40. Springer, Heidelberg (2007)
7. Greene, C.S., White, B.C., Moore, J.H.: Sensible initialization using expert knowledge for genome-wide analysis of epistasis using genetic programming. In: Proceedings of the IEEE Congress on Evolutionary Computation, pp. 1289–1296 (2009)
8. Hirschhorn, J.N., Daly, M.J.: Genome-wide association studies for common diseases and complex traits. Nature Reviews Genetics 6, 95–108 (2005)
9. Kononenko, I.: Estimating attributes: analysis and extensions of RELIEF. In: Bergadano, F., De Raedt, L. (eds.) ECML 1994. LNCS, vol. 784, pp. 171–182. Springer, Heidelberg (1994)
10. Lucek, P.R., Ott, J.: Neural network analysis of complex traits. Genetic Epidemiology 14, 1101–1106 (1997)

11. Moore, J.H., Greene, C.S., Andrews, P.C., White, B.C.: Does complexity matter? Artificial evolution, computational evolution, and the genetic analysis of epistasis in common human diseases. In: Genetic Programming Theory and Practice VI, ch. 9. Springer, Heidelberg (2009)
12. Moore, J.H., Parker, J.S., Olsen, N.J., Aune, T.M.: Symbolic discriminant analysis of microarray data in autoimmune disease. Genetic Epidemiology 23, 57–69 (2002)
13. Moore, J.H., Ritchie, M.D.: The challenges of whole-genome approaches to common diseases. Journal of the American Medical Association 291(13), 1642–1643 (2004)
14. Moore, J.H., White, B.C.: Exploiting expert knowledge in genetic programming for genome-wide genetic analysis. In: Runarsson, T.P., Beyer, H.-G., Burke, E.K., Merelo-Guervós, J.J., Whitley, L.D., Yao, X. (eds.) PPSN 2006. LNCS, vol. 4193, pp. 969–977. Springer, Heidelberg (2006)
15. Moore, J.H., White, B.C.: Genome-wide genetic analysis using genetic programming: The critical need for expert knowledge. In: Genetic Programming Theory and Practice IV, ch. 2. Springer, Heidelberg (2007)
16. Moore, J.H., White, B.C.: Tuning ReliefF for genome-wide genetic analysis. In: Marchiori, E., Moore, J.H., Rajapakse, J.C. (eds.) EvoBIO 2007. LNCS, vol. 4447, pp. 166–175. Springer, Heidelberg (2007)
17. Ritchie, M.D., Hahn, L.W., Roodi, N., Bailey, L.R., Dupont, W.D., Parl, F.F., Moore, J.H.: Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. American Journal of Human Genetics 69(1), 138–147 (2001)
18. Wagner, A.: Robustness and evolvability in living systems. Princeton University Press, Princeton (2007)