A DOUBLY LAYERED, GENETIC PENETRANCE LEARNING SYSTEM

Larry A. Rendell

Department of Computing and Information Science
University of Guelph
Guelph, Ontario, Canada. N1G 2W1

ABSTRACT

The author's original state-space learning system (based on a probabilistic performance measure clustered in feature space) was effective in optimizing parameterized linear evaluation functions. However, more accurate probability estimates would allow stabilization in cases of strong feature interactions. To attain this accuracy and stability, a second level of learning is added, a genetic (parallel) algorithm which supervises multiple activations of the original system. This scheme is aided by the probability clusters themselves. These structures are intermediate between the detailed performance statistics and the more general heuristic, and they estimate an absolute quantity independently of one another. Consequently the system allows both credit localization at this mediating level of knowledge and feature interaction at the derived heuristic level. Early experimental results have been encouraging. As predicted by the analysis, stability is very good.

I INTRODUCTION

In [7] the author described a successful state-space learning system (PLS1). Given a set of features, PLS1 will decide which are useful, and incrementally and efficiently determine the weight vector for the heuristic, a linear evaluation function. Heuristics have been repeatedly generated which solve the fifteen puzzle, and which are locally optimal in the weight space. This is a new result.

The underlying concept is a refinement of Doran and Michie's [4] search penetrance which measures solution density in feature space (see Fig. 1). Derived from repeated observations of this search statistic, the evolving evaluation function is designed not to estimate path distance remaining to the goal from a state A, as is often the case, but rather to predict the probability of A's eventual solution participation. For a given feature space volume r, the elementary penetrance $p(r, H, P)$ depends on the particular problem instance P and heuristic H used for solving. However the true penetrance $\bar{p}(r)$ is the ideal, defined for breadth-first search of all possible problem intances combined. Of course finding these values is infeasible, but penetrance learning systems (PLS's) estimate them.

As modelled by Buchanan et al [3] and exemplified in Fig. 2, a typical learning system (LS) comprises:

1. An algorithm schema for some primary task, the performance element PE

2. Some separable control structure S for the PE

3. The critic, whose role is "[to analyse] the current abilities of the performance element" PE[S], by assessment of the overall effectiveness of S, and sometimes also by localization of credit within S

4. The learning element LE, designed to improve S according to recomendations of the critic

5. A blackboard on which to store S and other information between activations of these algorithms.

In a penetrance learning system (PLS), the blackboard retains knowledge of the relationship of penetrance to feature values in a partition of the feature space, a set of regions of various sizes and shapes (Figs. 1,2). In a somewhat simplified form [c.f. 7,8], this cumulative region
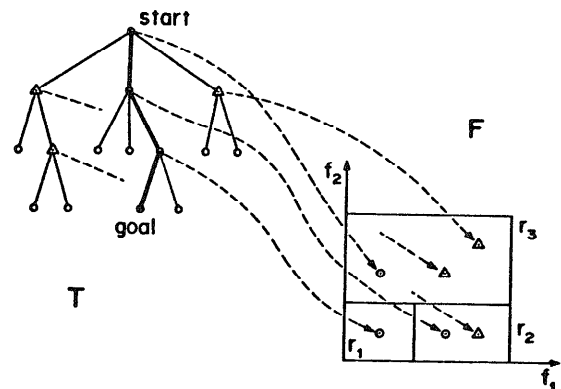

Figure 1. Localized penetrance discriminates. Developed nodes from search tree T are mapped into feature space F. The whole space penetrance of T is 3/6, whereas localization in F gives (e.g.) three elementary penetrance values: $p(r_1,T) = 1/1$, $p(r_2,T) = 1/2$, and $p(r_3,T) = 1/3$.

set can be written as $C = \{ (r, \hat{p}_r) \}$ where r is the feature space volume and $\hat{p}_r$ is its estimated true penetrance. This structure C is the essence of heuristic knowledge for the PLS, both determining the control structure for the solver, and also being the foundation on which the adaptive elements build. (See [7] for details.)

In the original system PLS1 there is no critic; no overall measure of solver capability is required. Rather the learning element operates locally on the cumulative regions.

The learning element of PLS1 is detailed in [6,7,8] and briefly described in the following: The LE includes two major algorithms, the clusterer and the regresser. The clusterer modifies the cumulative region set C using solver statistics. As information accumulates over several iterations, the regions are incrementally resolved into smaller units just adequate to express known relationships. In addition, the true penetrance estimates of C are revised each iteration: fresh, elementary values are unbiased (normalized to true penetrance), then averaged in. The result is an effective economy, a refinement of Samuel's [9] signature tables which did not alter data categories automatically. Although the clusterer is data-driven, its product C is unsusceptible to noise, because of the stochastic nature of the process.

From $C = \{ (r, \hat{p}_r) \}$ the feature coefficient vector b is determined by stepwise regression of log $\hat{p}_r$ on the centroid of r. This is a selective procedure which screens features and expresses their relative importance. The resulting evaluation function $H = \exp(\underline{b}.\underline{f})$ predicts true penetrance.

Rather than being confined to these linear combinations, features must ultimately be merged more flexibly if the system is to attain full generality (e.g. see Berliner [1]). In [8] feature interaction is accommodated using piecewise linearity, localizing b to individual regions. However, handling nonlinearities presents a severe problem of stability unless true penetrance estimates are quite reliable. To tackle this difficulty, the genetic model is applied. A genetic or reproductive plan is an inherently parallel scheme which can efficiently locate global optima. The theory was developed by Holland [5], summarized and exemplified in Brindle [2], and successfully incorporated in a learning system by Smith [10].

Shown in Fig. 3, the present extension PLS2 can be considered as a second layer LS which activates its performance element PLS1 multiply, with a different control structure each time. Essentially PLS1 operates in parallel, each process using an individual cumulative region set of the competing population. (The blackboard of PLS2 is the union of PLS1 blackboards.) The critic and learning component of PLS2 make comparisons and improve the population. Briefly outlined in [6], PLS2 is developed below.
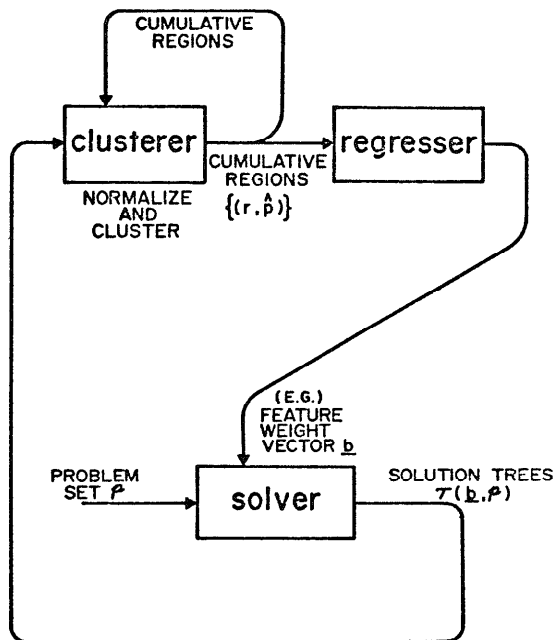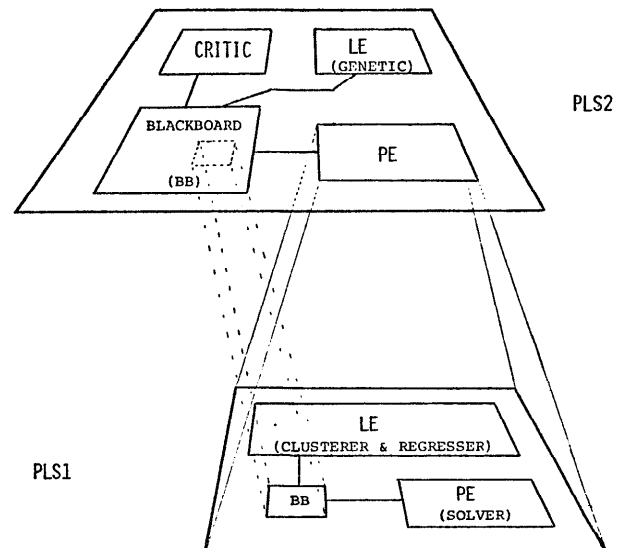

Figure 2. Penetrance learning system PLS1. The simplest control structure for a solver is a vector of weights for features of an evaluation function. The essence of PLS knowledge is a set of feature space penetrance regions, used to determine this heuristic and to accumulate experience.

Figure 3. The second layer of learning. PLS2 activates PLS1 with different region sets, which it continually improves.

344

II THE CRITIC

As explained above, the cumulative region set is essentially the control structure for the solver. In PLS2 a different region set guides each solver activation (Fig. 3), and a resulting performance statistic provides a basis of comparison to measure the relative worth of each structure. By taking advantage of performance patterns across region sets, credit can be localized to individual regions.

Suppose that K cumulative sets $c^{(k)}$ are available $(1 \leq k \leq K)$. Let each $c^{(k)}$ determine control for the solver in K separate runs, each attempting problem instances of difficulty d. (A "presolver" decides on an appropriate value of solution depth d for otherwise random selection of training problems.) Already calculated as a by-product by the solver (but not used in PLS1) are two overall measures of performance: the solution length L and number of nodes developed D; these are stochastic functions of $c^{(k)}$ and d. Choose some functional F of L and D (e.g. simply D). Then, given a cumulative region set $R = c^{(k)}$ $(k \leq K)$, define its coarse utility $\mu(R)$ to be $\bar{F}(d) / F(R, d)$, where $\bar{F}$ is the mean value of F over all K sets. Values of μ will therefore center around one. μ is a typical fitness measure for a reproductive plan (see Section III); however the critic is designed to extract more information than this.

To quantify credit localization, (pairwise) comparison of regions is used. Consider first the simplification in which each cumulative region has a counterpart in every other parallel set; i.e. feature space rectangles match precisely, and only true penetrance estimates differ. An example of this situation is shown in Fig. 4. For each focus region R, and comparison region Q, define the likeness$(R, Q) = 1 - (\hat{p}_R - \hat{p}_Q) / \max(\hat{p}_R, \hat{p}_Q)$,

where $\hat{p}_R$ and $\hat{p}_Q$ are the true penetrance estimates of R and Q. In the general case of dissimilar rectangles, this definition becomes asymmetric: Now a focus region R is compared with each cumulative set $Q = C^{(k)}$. Depending on the extent of its intersection with R, each $Q \in Q$ contributes to a varying degree to the overall penetrance similarity of Q to R.

This likeness measure, together with region set performance, provide veiled information relating $\hat{p}_r$ to its accuracy. Assuming coincidence of true penetrance with optimal utility, $R \in R$ will tend to improve the performance $\mu(R)$ if R is accurate, and regions similar to R will be likely to aid their sets. Consider again Fig. 4. Each cumulative set $c^{(k)}$ will have determined a heuristic for attempting training problems of similar difficulty, and the resulting coarse utilities might be as indicated. We can conclude that regions in set $c^{(2)}$ have generally better estimates than those in $c^{(1)}$ or $c^{(3)}$. If, for each focus region R, the coarse utilities are plotted as a function of likeness, a peak will likely occur at the point of greatest accuracy, and if a curve is fitted to the data, the resulting utility at likeness = 1 will estimate the pure utility of $R \in R$, rather independently of other regions in R.

Various outcomes are illustrated in Fig. 5, in which parabolas are fitted. The first (a) would never occur since it suggests that regions from a single location of feature space in every cumulative set are nearly exclusively responsible for the overall performance of the heuristic. The other two diagrams are more likely: There is little utility attributable to a single focus region R (R could be quite accurate while its neighbours vitiate the heuristic). Fig. 5(b) indicates a situation in which R is quite inaccurate but parallel sets typically do not suffer as severe a deficiency. Fig. 5(c) might result if all regions in the set containing R are fairly sound. Here competing sets are often poorer; in particular many rivals of R are less accurate (and therefore unlike R).
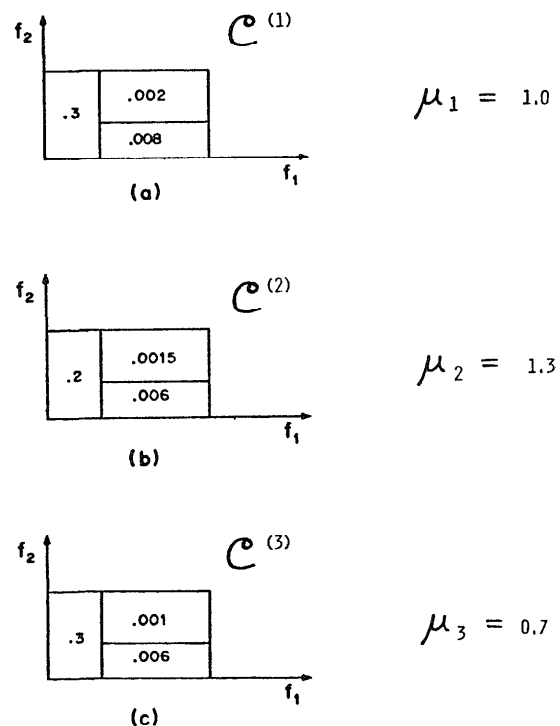

$$c^{(1)} \qquad \mu_1 = 1.0$$

$$c^{(2)} \qquad \mu_2 = 1.3$$

$$c^{(3)} \qquad \mu_3 = 0.7$$

Figure 4. Different region sets cause varying performance μ. In this simplified picture (of just 3 parallel sets $c^{(k)}$), rectangles always match and only true penetrance estimates (shown inside) differ.

One would expect generally poor fits for most individual focus regions but significant knowledge overall, since many focus regions are assessed (all JK of them, with an average of J in each set and K sets). The precise mechanism for this information extraction is straightforward: Perform a regression, then compute the <u>fine utility</u> (R) of focus region R which can be defined variously but similarly as $\mu'_r{}^2$ or $\mu'_r{}^2 g^{(K)}$, where μ' is the predicted utility at likeness = 1, r is the correlation coefficient, and g is some function increasing monotonically with argument K (population size). $\nu = 1$ indicates uncertainty while larger or smaller values of ν show confidence in greater or lesser quality. The fine utilities corresponding to Fig. 5 (b) and (c) are 0.7 and 1.2, respectively, with the simpler definition above.
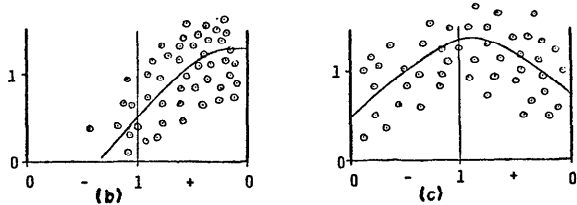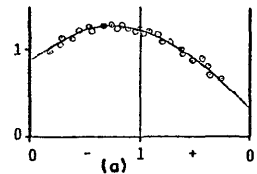

Figure 5. Credit localization. Examinination of region similarity (abscissa) across multiple cumulative sets allows extraction of patterns in performance (ordinate).

III THE REPRODUCTIVE PLAN

To utilize this information, PLS2 incorporates a novel version of Holland's genetic plan [5]. A genetic algorithm uses parallel structures called genotypes; each determines the phenotype, a set of attributes characterizing an individual. The fitness of an individual is its performance in the environment. This measure is used to favour selection of successful parents for new offspring, so the whole population incrementally evolves toward greater utility. Theory [5] shows that knowledge about desirable phenotypes is advantageously stored in the population itself, implicitly in the surviving genotypes. Reproductive plans can locate global optima efficiently [5,10].

In the design of an artificial genotype, one issue is whether to use many loci (variables) with few alleles (values), or vice virsa. To allow greater adaptability, binary alleles are generally chosen, although this can cause problems [2, pp. 24-26, 44-47]. This issue dissolves in PLS2: First, population variance is aided by the learning element of PLS1, so the alleles need not be binary. Instead the allele set is the continuous interval [0,1], representing true penetrance. Secondly, the loci of a PLS2 genotype correspond to feature space coordinates; however they are compressed into unordered volumes, and their number depends on current knowledge refinement. The genotype is the cumulative region set.

Since regions estimate true penetrance, and only within their own boundaries, these "loci" are independent, thus precluding another problem: inefficiency due to loci interaction [2, p. 170]. The consequent phenotype, the feature coefficient vector, can still be nonlinear (when a high order model is used or this vector is regionalized -- see [8]).

To optimize the population, a reproductive plan includes algorithms for parent selection and offspring generation. Parent selection is natural: Each individual has an associated fitness measure such as the coarse utility of Section II. This simply defines a probability distribution for candidates so that successful parents are favoured. The fitness measure is normally a property of the individual as a whole; typical applications do not admit localization of credit since loci usually interact. However the PLS2 genotype -- the cumulative region set -- allows the assignment of fine utility (Section II).

The genotypes of a population are both repositories of knowledge and also sources of subtle variation for exploration. To achieve balance in this mutual role, offspring generation typically adopts biological operators such as mutation (unary) and crossover (binary or bisexual). In contrast, PLS2 is K-sexual (where K is population size); all regions are merged into a single set before selection. Moreover, alleles (true penetrance estimates) are untouched (it is the lower level learning element PLS1 which alters these); regions are simply chosen stochastically as loci/alleles according to their fine utility. To create reasonable offspring region sets, the utility of every candidate region (its probability of selection) is continually adjusted to account for the current feature space cover V defined by the regions so far selected (candidates are less useful if they overlap much of V). This formation of a new set halts when a V is attained which is close to the maximum. Hence a new individual arises which has a high likelihood of penetrance accuracy. K' genotypes are created in this manner to replace all of the old population.

IV PRELIMINARY RESULTS AND CONCLUSIONS

The second layer system PLS2 has been programmed and testing has begun. In particular, comparisons are being made with the already successful PLS1 [7]. Perhaps the most obvious improvement is in terms of stability. Whereas PLS1 is sensitive to various run parameters and appropriate training problems, PLS2 overcomes any abnormalities by immediately dismissing aberrant information. Additional time costs appear small. Investigations are continuing to discover the effects of varying system parameters such as population size, and in particular to determine the ability of PLS2 using more highly interacting features, which PLS1 cannot handle alone.

In summary, PLS2 is promising from several viewpoints: As support for PLS1, PLS2 improves region accuracy and stability, important for feature interaction [8]. As a genetic algorithm, PLS2 seems especially efficient because of the independence and flexibility of individual loci (regions). These characteristics avoid typical problems which can degrade efficiency, and also aid credit localization which usually improves it. Despite the absence of explicit genetic operators, the ability to discover global optima may be retained since PLS1 already provides (controlled) population variance. Finally, as a scheme for knowledge accumulation, PLS2 benefits from information layering. A mediating structure, the cumulative feature space region set (storing conditional probability of success in task performance), allows both credit localization and variable interaction: The elements of this set, the regions, are independent of one another, but determine the task heuristic which can incorporate feature nonlinearities.

ACKNOWLEDGEMENT

REFERENCES

1. Berliner, H., On the construction of evaluation functions for large domains, Proc. Sixth IJCAI, 1979, 53-55.

2. Brindle, A., Genetic algorithms for function optimization, C.S. Department Report TR81-2 (PhD Dissertation), University of Alberta, 1981.

3. Buchanan, B.G., Johnson, C.R., Mitchell, T.M., and Smith, R.G., Models of learning systems, in Belzer, J. (Ed.), Encyclopedia of Computer Science and Technology 11 (1978) 24-51.

4. Doran, J. and Michie, D., Experiments with the graph-traverser program, Proc. Roy. Soc., A, 294 (1966) 235-259.

5. Holland, J.H., Adaptation in Natural and Artificial Systems, University of Michigan Press, 1975.

6. Rendell, L.A., State-space learning systems using regionalized penetrance, Proc. Fourth Biennial Conference of the Canadian Society for Computational Studies of Intelligence, 1982, 150-157.

7. Rendell, L.A., A new basis for state-space learning systems and a successful implementation, Artificial Intelligence 20, 4, (July 1983).

8. Rendell, L.A., A learning system which accommodates feature interactions, Proc. Eighth IJCAI, 1983.

9. Samuel, A.L., Some studies in machine learning using the game of checkers II -- recent progress, IBM J. Res. and Develop. 11 (1967) 601-617.

10. Smith, S.F., A learning system based on genetic adaptive algorithms, PhD Dissertation, University of Pittsburgh, 1980.