Estimation of Distribution Algorithms

Petr Pošík

Czech Technical University, Faculty of Electrical Engineering, Dept. of Cybernetics
Technická 2, 166 27 Prague 6, Czech Republic
E-mail: `posik@labe.felk.cvut.cz`, phone: +420-2-24357228

1 Introduction

Evolutionary algorithms (EAs) are known in many areas as a powerful and robust optimization and searching tool. Classical EAs rely on the well-known two phases: selection and variation. Variation is usually carried out by means of perturbation of promising individuals (searching local neigbourhoods), or by means of combining two promising individuals together (creating offsprings which embody some characteristics of both parents). However, classical EAs suffer from several problems. The linkage problem belongs among the most severe ones. It arises in situations when the individual components of chromosomes are not statistically independent of each other with respect to the fitness function. There exists no general way of EA modification that would enable the modified EA to account for the dependencies at hand. Usually, this problem is solved by constructing special crossover and mutation types of operators and by incorporating some problem-specific knowledge in them. The classical EA then looses its flavor of general problem solver and quickly becomes an algorithm highly specialized to the given problem.

2 Estimation of Distribution Algorithms

Recently, a new type of EAs emerged — *Estimation of Distribution Algorithms* (EDAs) [1]. Some researchers use names as *Probabilistic Model Building Genetic Algorithms* (PMBGAs), or *Iterated Density Estimation Algorithms* (IDEAs), but all these names describe basically the same concept. These algorithms don't rely on the 'genetic' principles anymore; instead, in each generation, they build an explicit probabilistic model of distribution of 'good' individuals in the search space. New individuals are created by sampling from this distribution. The *model-sample* step of EDA can be thought of as a generalized type of multiparent crossover operator. The strengths and weaknesses of a particular EDA are mainly determined by the used probabilistic model.

2.1 Probabilistic Models for Discrete Variables

The probabilistic models differ for EDAs in discrete and continuous spaces. The first EDAs were developed for the discrete spaces. They range from simple *Univariate Marginal Density Algorithm* (UMDA), which is comparable to simple genetic algorithm, to *Bayesian Optimization Algorithm* (BOA) [2] which uses Bayesian net as the underlying probabilistic model. Bayesian nets are able to encode general type of discrete probabilistic distribution, however, their learning from data involves either sophisticated methods for statistical dependency detection, or they are learnt by searching the space of possible Bayesian nets (usually by a greedy algorithm).

2.2 Probabilistic Models for Continuous Variables

In continuous spaces, the situation is even more complicated. The simplest continuous EDAs (continuous UMDAs) use models in which the joint probability density function (PDF) is factorized into a product of marginal univariate PDFs which take various forms: empirical histograms, normal (or any other well-known) distribution, finite mixtures of univariate Gaussians, etc. To take into account the dependencies between variables, we have to employ more complex models like Gaussian nets (GN), which results in *Estimation of Gaussian Networks Algorithm* (EGNA) [1]. GN has the power to encode general multi-dimensional Gaussian distribution, however, very often this type of probabilistic model is not sufficient. Then we should use even more flexible models which are empowered by (hard- or soft-) clustering, e.g. finite mixture of multidimensional Gaussians. To be objective, one must say that these models are capable in covering various types of interactions, however, learning them is not a trivial task. It is usually very time consuming and it must be performed using a kind of iterative learning scheme (usually by a variant of the expectation-maximization algorithm).

3 Marginal Models in EAs

My research is aimed at the EDAs in continuous spaces. I have examined the UMDA in continuous domain. The individual components of promising solutions are supposed to be statistically independent of each other. This means that the global distribution of promising solutions in the search space can be modeled by a set of univariate marginal distributions, i.e. the global model can be factorized as

$$p(\mathbf{x}) = \prod_{d=1}^{D} p_d(x_d), \tag{1}$$

where the $p(\mathbf{x})$ is the global multivariate density and the $p_d(x_d)$'s are the univariate marginal densities. I compared the suitability of four different marginal probability models, namely the equi-width histogram (HEW), equi-height histogram (HEH), max-diff histogram (HMD), and univariate mixture of Gaussians (MOG) (for the differences of individual models, see fig. 1). For the suite of test functions, see [3].
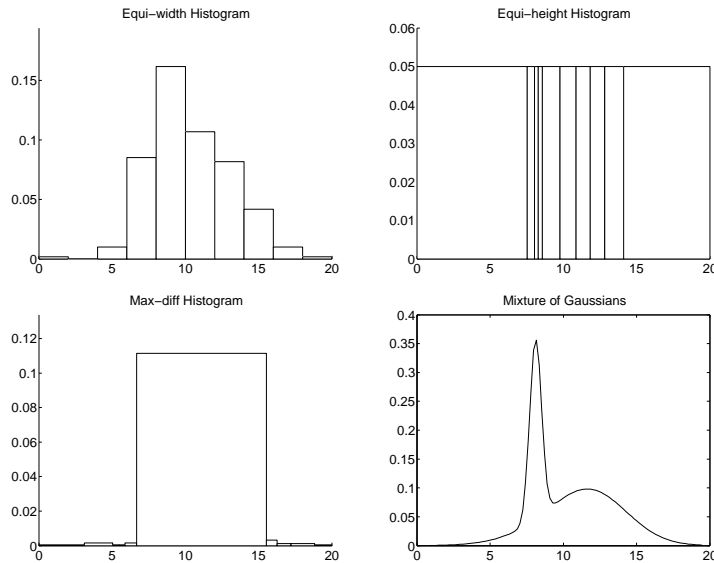

Figure 1: Equi-width, equi-height, and max-diff histograms with 10 bins, and mixture of gaussians with 3 components

In the experiments, I varried the population size (200, 400, 600, 800 individuals), the number of bins for histogram models (120 and 60 bins), or the number of components for the case of MOG model (6 and 3 components). Furthermore, all models were compared to the *line search* heuristic [4], which is very efficient for high-dimensional separable problems.

In each generation, new $PopSize$ individuals were created, joined with the old population, and using truncation selection, the population was reduced to its original size. For each of possible factor combination I run the algorithm 20 times. Each run continued until the number of 50,000 evaluations was reached. Let's say the algorithm found the global optimum if for each variable x_d the following relation holds: $\left|x_d^{best} - x_d^{opt}\right| < 0.1$ (if the difference of the best found solution x^{best} from the optimal solution x^{opt} is lower then 0.1 in each of coordinates). In all experiments, we track three statistics:

- The number of runs in which the algorithm succeeded in finding the global optimum (*NoFoundOpts*).

- The average number of evaluations needed to find the global optimum computed from those runs in which the algorithm really found the optimum (*AveNoEvals*).

- The average fitness of the best solution the algorithm was able to find in all 20 runs (*AveBest*).

The results can be found in table 1. From the experiments the following conclusion can be made: the HEW model is the least flexible one and the behaviour of EDAs with this model is unsatisfactory in comparison with the other models. The performance of HEH and HMD histograms was comparable. The MOG model showed a bit worse performance, however, it used considerably less components than the histogram models and offers other advantages over the histogram models (easy extension to mixture

Function	Model	Number of Bins (Components)								Statistics
		Low				High				
		Population Size								
		200	400	600	800	200	400	600	800	Statistics
Two Peaks	LS	20								NoFoundOpts
		2421								AveNoEvals
		0,00000								AveBest
	HEW	0	1	0	1	1	17	19	20	NoFoundOpts
			13200		47200	5600	10306	15189	19600	AveNoEvals
		5,00560	4,82110	5,05280	4,95510	3,48040	2,51180	2,50030	2,52340	AveBest
	HEH	20	20	20	20	20	20	20	20	NoFoundOpts
		6530	11080	16050	20760	7720	10620	15570	20400	AveNoEvals
		0,10710	0,02070	0,01040	0,07490	0,03740	0,00690	0,00430	0,03520	AveBest
	HMD	20	20	20	20	20	20	20	20	NoFoundOpts
		6270	14920	27960	47080	6770	10980	17490	25280	AveNoEvals
		0,00003	0,00008	0,04990	2,20380	0,00260	0,00001	0,00230	0,05740	AveBest
	MOG	7	20	20	19	16	20	20	20	NoFoundOpts
		6571	11860	17130	23032	5613	10480	15780	20720	AveNoEvals
		0,87390	0,00011	0,00930	0,13970	0,19650	0,00008	0,00660	0,06370	AveBest
Griewangk	LS	18								NoFoundOpts
		14056								AveNoEvals
		0,00250								AveBest
	HEW	13	17	16	14	1	15	18	19	NoFoundOpts
		15954	20353	25763	27886	5800	12320	18867	24926	AveNoEvals
		0,00370	0,00210	0,00230	0,00320	0,00500	0,00081	0,00095	0,00083	AveBest
	HEH	18	18	18	20	16	20	20	20	NoFoundOpts
		6667	12867	18967	25000	6650	12711	18270	23920	AveNoEvals
		0,00074	0,00074	0,00074	0,00000	0,00150	0,00074	0,00000	0,00000	AveBest
	HMD	17	17	18	16	18	17	19	20	NoFoundOpts
		6482	12376	19200	25850	6456	12235	18221	23720	AveNoEvals
		0,00110	0,00110	0,00074	0,00140	0,00074	0,00110	0,00037	0,00000	AveBest
	MOG	15	15	18	19	19	19	17	18	NoFoundOpts
		5693	10613	16100	21726	5894	12084	17682	23644	AveNoEvals
		0,00190	0,00180	0,00074	0,00037	0,00037	0,00037	0,00110	0,00074	AveBest

Table 1: Results of carried out experiments

of multidimensional Gaussians). Typical tracks of evolution of bin boundaries for histogram models and component centers of MOG for one of the test functions is shown in figure 2.
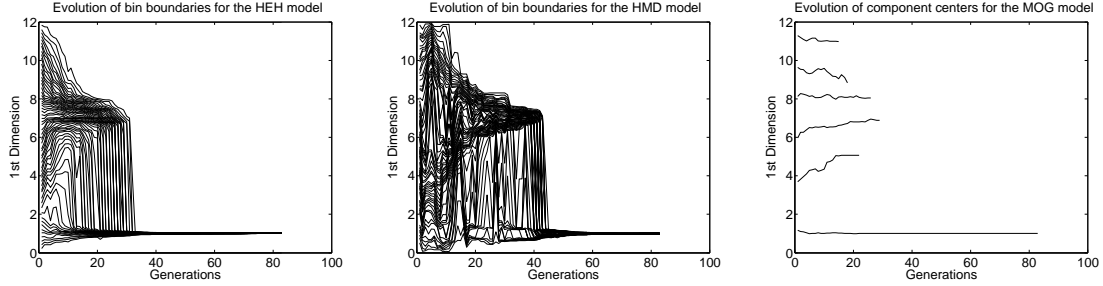

Figure 2: Two Peaks function — Evolution of bin boundaries for equi-height and max-diff histogram models and evolution of component centers for mixture of Gaussians model.

4 Vestibulo-Ocular Reflex Analysis

The above described algorithm was succesfully applied to vestibulo-ocular reflex (VOR) signal processing. By analyzing the VOR signal, physicians can recognize some pathologies of the vestibular organ of a patient in a non-invasive way. The principle is simple: the patient is situated in a chair which is then rotated in a defined way (following some reference signal – sine wave or sum of sine waves). The patient is said to visually track some points on surrounding walls and the movements of his eyes are monitored. The resulting eye signal must be first processed (it is distorted by the fast eye movements) to get 'eye-filtered' response to the reference signal. The differences in amplitudes and phases of the sine waves are the indicators of the vestibular organ pathologies. EDA was applied in the signal processing phase in a co-evolutionary manner, i.e. the following two parts were iteratively alternated: (1) one population searched for the best biases of individual signal segments (when fitted to the best representant of estimated signal
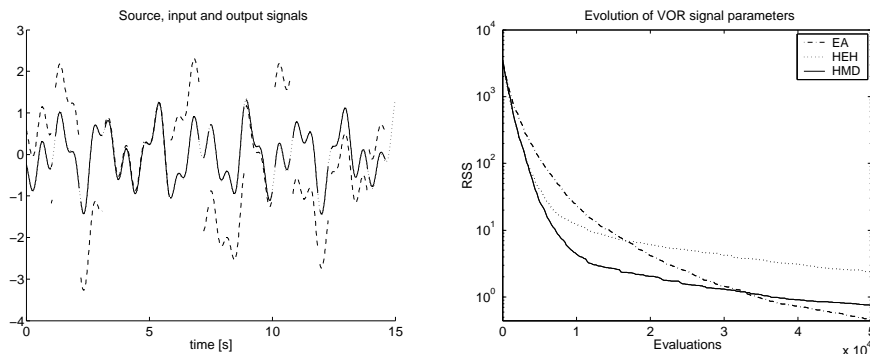
Figure 3: VOR signals and their evolution.

parameters), while (2) the other population searched for the best signal parameters (when fitted to the best set of biases).

I compared two of the above mentioned UMDAs (HEH and HMD models) with an ordinary EA (truncation selection, 2-point crossover, mutation with probability 0.05 by means of adding a random value from distribution $\mathcal{N}(0, 0.1)$). The results are presented in fig. 3. On the left-hand side of the picture, we can see the original signal (dotted line, not known to the EA), the signal segments before EA started (dashed line), and the same signal segments after EA processing (solid line). We can see, that all the segments are very precisely arranged so that they follow the original signal very closely. Both UMDAs were much faster than the EA in the initial phase of evolution. The EA is able to discover more accurate solution, however, the final differences are not very large – measured in residual sum of squares (RSS), the EA reached the score of 0.44 on average, the UMDA/HMD reached 0.75, and the UMDA/HEH reached 2.36 (solutions of these scores are almost identical when compared by human eye). In spite of these differences, all these EAs reach much more accurate results (in terms of RSS) than conventionally used methods based on some form of interpolation.

5   Future Work

In the near future, I would like to implement an EDA using mixture of principal component analysers (MPCA) and test it on several artificial and practical problems (e.g. on Hough's transformation used in image processing, or for hidden Markov models training). The aim of these comparative studies is to find out if it is worth to use such complex models (e.g. MPCA), in other words, if the time spent on learning the model each generation is lower than the time the simple EA needs to find a solution of comparable quality.

Acknowledgements

References

[1] Larrañaga, P., Lozano, J.A. Estimation of Distribution Algorithms. Kluwer Academic Publishers, 2001.

[2] Pelikan, M. Bayesian Optimization Algorithm: From Single Level to Hierarchy. IlliGAL Report No. 2002023. University of Illinois at Urbana-Champaign. 2002.

[3] Pošík, P. Optimization of Evolutionary Algorithms. Technical Report GL 164-03, Department of Cybernetics, Czech Technical University in Prague, 2003.

[4] Whitley, D., Mathias, K., Rana, S., Dzubera, J. Evaluating Evolutionary Algorithms. Artificial Intelligence Volume 85, p. 245-276, 1996.