UNIT II SEARCH METHODS AND VISUALIZATION 9

Search by simulated Annealing – Stochastic, Adaptive search by Evaluation – Evaluation Strategies –Genetic Algorithm – Genetic Programming – Visualization – Classification of Visual Data Analysis Techniques – Data Types – Visualization Techniques – Interaction techniques – Specific Visual data analysis Techniques

---

EVOLUTION STRATEGIES

Evolution Strategies (ES) were invented during the 60s and 70s as a technique of evolutionary experimentation for solving complex optimization problems, mainly within engineering domains. The preferred ES data structures are vectors of real numbers. Specifically tailored mutation operators are used to produce slight variations on the vector elements by adding normally distributed numbers (with mean zero) to each of the components. Recombination operators have been designed for the interchange or mixing of "genes" between two or more vectors. These recombinations range from simply swapping respective components among two vectors to component-wise computation of means. Evolution Strategies have developed sophisticated methods of selection, which are especially important when the evolution scheme involves several subpopulations.

Representation of Individuals

The basic data structures today's Evolution Strategies deal with, and around which most of the ES theory is built, are vectors of real numbers representing a set of parameters to be optimized. Therefore, an ES chromosome g, can be simply defined as follows:

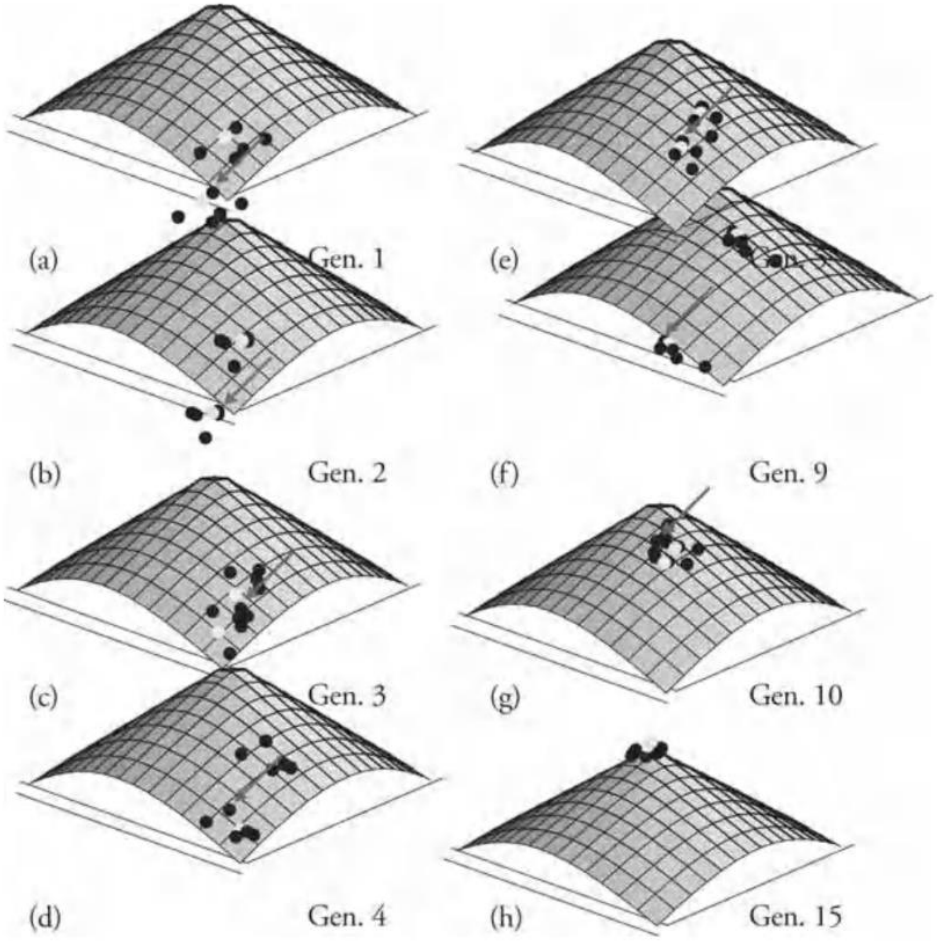
$$g = (p_1,p_2, ... ,p_n) \text{ with } p_i \in K$$

Usually the p_i are referred to as the object parameters. In the most elementary versions of Evolution Strategies only these parameter vectors are subject to evolution. However, in order to be able to solve more complex optimization tasks it turns out to be advantageous to keep a second vector of strategy parameters, which are used as variances to control the range of mutations on the object parameters. Thus we can extend our definition for an ES chromosome g the following way:

$$g = (p,s) = (p_1,p_2, ... ,p_n), (s_1,s_2, ... ,s_n) \text{ with } p_i , s_i \in K$$

Now the ES chromosome vector is two-fold, with each Sj representing the mutation variance for the corresponding ft. The additional control parameters s may be considered as an internal model of the optimization environment

ES Mutations



We start our journey at some distance from the top plateau with one individual, depicted as a gray spot. From the genotype, an ES chromosome containing the x- and y-coordinates of this individual, we generate five mutated offspring(black spot), the locations of which are somewhat close to the parent individual. This set of points comprises our initial population on which we will perform a selection-mutation procedure as follows. We choose the individual currently with the highest location, from which we generate another five children, and continue the same way by selecting the best, generating mutants, etc.

Repeatedly choosing the best individual and generating mutated offspring, gradually makes the population move upwards, like following a gradient to the top plateau. A closer look at Figure reveals that smaller mutations are prevalent, i.e. the majority of mutants (children) are located near to the wildtype (parent).

Not only the object parameters, the coordinates in our example, but also the strategy parameters, controlling the mutation step sizes, are subject to change. This effect is also visible in the figure. Each strategy parameter controls the mutation step size, hence the distance of the children from their parents.

Mutating the Object Parameters

Mutation is considered the major ES operator for variations on the chromosomes. Mutations of the object and strategy parameters are accomplished in different ways. Basically, ES mutation on a chromosome g = (p, s) can be described as follows:

$$g_{mut} = (p_{mut}, s_{mut}) = (p + N_0(s), \alpha(s)).$$

Here $N_0(s)$ denotes normal distribution with mean zero and the vector of variances s. α defines a function for adapting the strategy parameters. The variations on the object parameters are to be applied to each vector element separately:

$$p_{mut} = (p_1 + N_0(s_1)), \ldots\ldots, p_n + N_0(s_n))$$

The strategy parameters are adjusted in an analogous way as

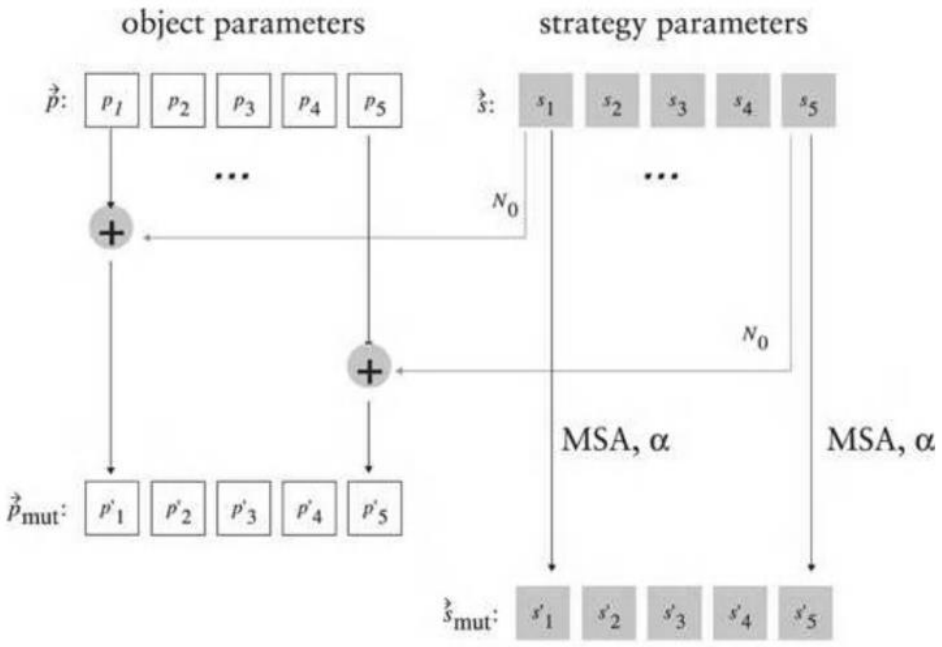
$$s_{mut} = (\alpha(s_1), \alpha(s_2), \ldots\ldots, \alpha(s_n))$$

Let us assume that strategy parameters remain unchanged, i.e., we use the identity mapping $\alpha(x) = x$.

Figure graphically depicts this basic mutation scheme. Each strategy parameter controls the mutation range for its respective object parameter. In order to have mutations, prefer smaller changes to larger ones. Evolution Strategies use normal (Gaussian) distributed random numbers that are added to the object parameters. The characteristics of Gauss distributions, $N_m(d)$ with mean **m** and standard deviation $\sqrt{\mathbf{d}}$,

$$\mathcal{N}(m,d) = \frac{1}{\sqrt{2\pi d}} \cdot e^{-\frac{(x-m)^2}{2d}}$$

are their preference for numbers around mean m. Thus, in the case of m = 0, smaller values close to zero are selected more often than values with greater distance to m.



Mutating the Strategy Parameters

There are two methods generally used for adapting the strategy parameters s of an ES chromosome ***g = (p,s):***

$$g_{mut} = (p_{mut}, s_{mut}) = (p + N_0(s), \alpha(s)).$$

Here α defines a function for adapting the strategy parameters:

$$s_{mut} = (\alpha(s_1), \alpha(s_2), \ldots\ldots, \alpha(s_n))$$

Each strategy parameter controls the mutation range of its respective object parameter, defining the variance for the normal distributed random values added to the object parameters.

There are several methods for defining α which work fairly well in changing the strategy parameters; this adaptation is usually referred to as Mutative Step size Adaptation (MSA).

$$\boldsymbol{s}_{\mathrm{mut}} = (s_1\zeta_1, \ldots, s_n\zeta_n)$$

$$\zeta_i = \begin{cases} \beta & : \quad \chi < 0.5 \\ \frac{1}{\beta} & : \quad \chi \geq 0.5 \end{cases}$$

Here χ denotes a uniformly distributed random variable from the interval [0,1]. For n < 100 parameters, β values will be between 1.3 and 1.5. For β = 1.3 means that half of the strategy parameters are multiplied by 1.3, the rest is multiplied by 0.77 = 1/1.3.

ES Recombinations

Recombination operators create new chromosomes by composing corresponding parts of two or more chromosomes. For the binary case, where two ES chromosomes,

$$g_a = (p_a, s_a) \text{ and } g_b = (p_b, s_b)$$

are to be recombined by an operator ω_{rec} and describe the composition of a new chromosome as follows:

$$\omega_{\mathrm{rec}}(\boldsymbol{g}_a, \boldsymbol{g}_b) = (\boldsymbol{p}', \boldsymbol{s}') = ((p'_1, \ldots, p'_n), (s'_1, \ldots, s'_n)).$$

Each element of the object and strategy parameter vector is a combination of the respective entries of g_a and g_b, by two functions ρ_p and ρ_s :

$$p'_i = \rho_p(p_{ai}, p_{bi}) \quad \text{and} \quad s'_i = \rho_s(s_{ai}, s_{bi}).$$

Here the functions ρ_p and ρ_s define the component-wise recombination mapping for the object and strategy parameters, respectively. In order to keep the formulas simpler, we will assume an identical recombination mapping for both the object and strategy parameters, $\rho = \rho_p = \rho_s$

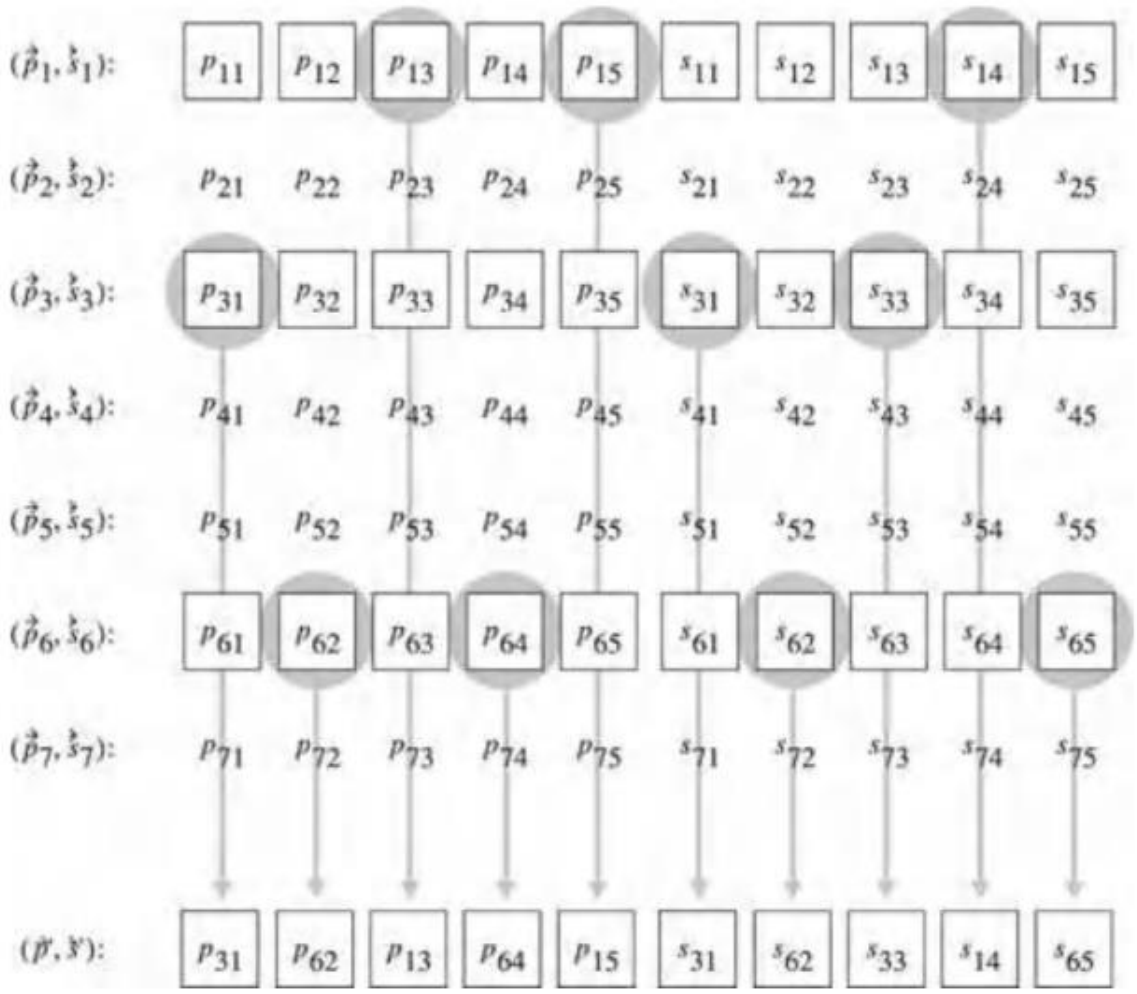
Two recombination mappings in Evolution Strategies are discrete $\rho = \rho_{dis}$ and intermediate $\rho = \rho_{int}$ recombination.

Discrete Recombination:

With a discrete recombination function, ρ_{dis} one of the two vector components is chosen at random and declared to be the new vector entry. For the case of binary recombination this means:

$$\rho_{\mathrm{dis}}(x_a, x_b) = \begin{cases} x_a & : \quad \chi < 0.5 \\ x_b & : \quad \chi \geq 0.5 \end{cases}$$

Here χ computes a uniformly distributed random number from the interval [0,1]. Each component x_a or x_b is selected with a 50-percent probability. In general, for μ values $x_1, x_2, \ldots x_\mu$ to be recombined in discrete manner, $\rho_{dis}(x_1, x_2, \ldots x_\mu)$, the probability to choose parameter x_i is $1/\mu$.



The above figure illustrates discrete recombination of three ES chromosomes ***(p_1,s_1), (p_3,s_3) and (p_6,s_6)*** into a new chromosome ***(p',s')***. In Evolution Strategies discrete recombinations are mainly used for interchanging strategy parameters, i.e., usually $\boldsymbol{\rho = \rho_{dis}}$

Intermediate Recombination:

For many ES application domains dealing with real numbers that represent some control parameter settings, taking the mean value of corresponding elements turns out to be a sensible and natural operator. This is exactly what intermediate recombination does. Recombining **μ** chromosomes intermediately means that the following mapping, $\boldsymbol{\rho_{int}}$, is applied to each set of corresponding vector components:

$$\rho_{\text{int}}(x_1, \ldots, x_\mu) = \frac{1}{\mu} \cdot \sum_{i=1}^{\mu} x_i$$

In Evolution Strategies intermediate recombination is inter-chromosomal operator used for object parameters, i.e., usually $\boldsymbol{\rho = \rho_{int}}$

Local and Global Recombinations:

Local recombinations work on a subpopulation of chromosomes, whereas for global recombinations each component can be selected from the set of corresponding entries among all chromosomes in a population. This results in an increased mixing of genotypic information. The multi recombination on a (sub-)population G = (xi,... ,Xn) of n ES chromosomes, each being of the form

$$\boldsymbol{x_i} = (x_{i1}, \ldots, x_{in}) \quad \text{with} \quad 1 \leq i \leq \mu.$$

A multi-recombination operator Wrec working on r chromosomes will be formalized as:

$$\omega_{\text{rec}}(\boldsymbol{x}_{i_1}, \ldots, \boldsymbol{x}_{i_r}) = \boldsymbol{x'} = (x'_1, \ldots, x'_n).$$

Local intermediate and discrete recombinations $\omega_{\text{rec}}^{\text{local}}$, i.e., for r recombinants the new chromosome is computed as follows:

$$\omega_{\text{rec}}^{\text{local}}(\boldsymbol{x}_{i_1}, \ldots, \boldsymbol{x}_{i_r}) = \boldsymbol{x'} = (x'_1, \ldots, x'_n)$$

$$x'_k = \begin{cases} x_{k,i_1} \text{ or } \ldots \text{ or } x_{k,i_r} & : \quad \rho = \rho_{\text{dis}} \\ \frac{1}{r} \cdot \sum_{i=1}^{r} x_{k,i_m} & : \quad \rho = \rho_{\text{int}} \end{cases}$$

We define a global recombination operator, $\omega_{\text{rec},r}^{\text{global}}$ that combines elements "per column" as follows:

$$\omega_{\text{rec},r}^{\text{global}}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) = \boldsymbol{x'} = (x'_1, \ldots, x'_N)$$

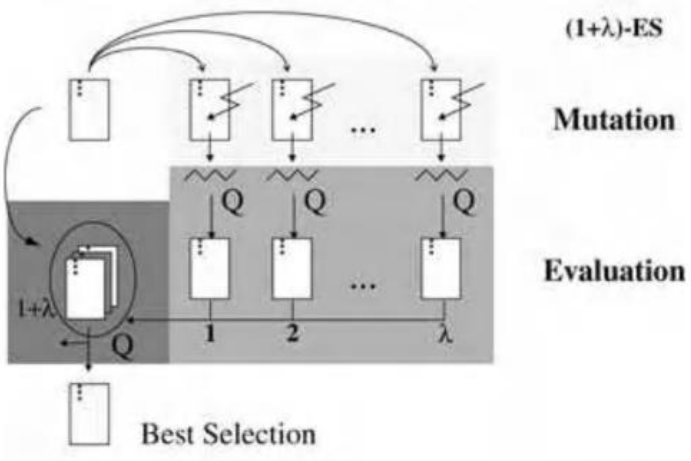
$$x'_k = \rho(x_{i_{1k}k}, \ldots, x_{i_{rk}k}).$$

Evolution Scheme

ES schemes are known as **comma** and **plus** strategies. Initial scheme is started with one parent producing one or more mutated offspring and extends with **μ** parents producing **λ** mutated offspring $(\mu/\rho \dot{+} \lambda)$.

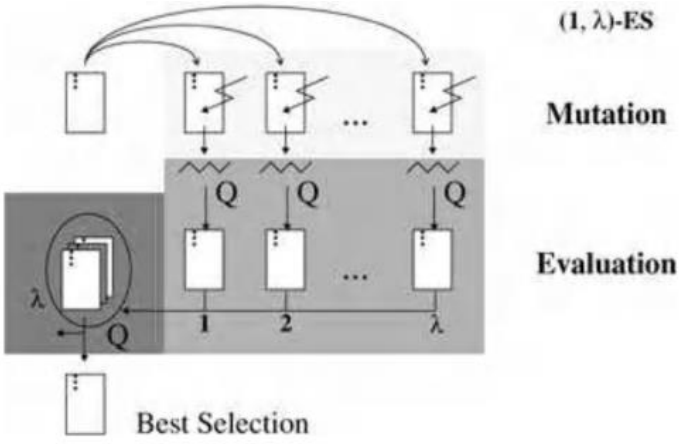
$(1 \dot{+} \lambda)$ Evolution Schemes:

The most simple and original ES scheme is known as a (1 + **λ**) Evolution Strategy. In Figure, a single parent individual produces **λ** offspring by mutation.



The parent genotype is duplicated **λ** times and all copies are subsequently mutated. The offspring's phenotypes are evaluated and both sets of individuals, the parent and offspring, find themselves in a selection pool. Only the best of these individuals will survive and serve as the parent for the next generation loop. The abbreviated notation for this kind of reproduction process is (1 + **λ**)-ES, where the 1 refers to the number of parents and **λ** is the number of offspring. The '+' sign is used to describe the composition of the final selection pool which contains the parent as well as its children. With a (1+**λ**)-ES the single parent survives into the following generation if all its offsprings' fitnesses are worse. Thus the parent can only be replaced by a superior offspring individual, which means that the fitness of the best-so-far individual either remains the same or increases.

A simple remedy for this is to exclude the parent individual from the final selection pool using a (1, **λ**)-strategy. By excluding the parent from the selection pool, the best individual among the offspring becomes the new parent individual for the next generation. This selection scheme is referred to as a (1, **λ**)-ES, the **comma** symbolizing the parent's exclusion from the selection pool.



$(\mu/\rho \dagger \lambda)$ **Evolution Schemes:**

Taking into account that several μ parents produce a population of (λ) offspring, we arrive at a ($\mu + \lambda$)-ES or a (μ , λ)-ES, respectively. Instead of selecting a single individual from the selection pool as the designated parent, now the μ best individuals among the selection pool of λ individuals will survive into the next generation. In the case of a (μ , λ)-ES strategy, we must ensure that there are enough individuals from which to select, that is, $\lambda \geq \mu$
