Société de Calcul Mathématique, S. A.
Algorithmes et Optimisation

Some general comments about randomness in algorithms

Bernard Beauzamy

July 13th, 2007

The following remarks are inspired by the talk given by Marc Arnaudon during our colloquium, July 10th, 2007. This talk concerned the splitting of a region into smaller zones, of homogeneous nature, where garbage collecting has to be made.

The use of Monte-Carlo methods in order to compute an integral is widely spread out and completely justified by a theoretical result (law of large numbers). In a K dimensional space, if you have a subset A of measure a inside a set B of measure b, if you throw N points at random inside B, the proportion of points which fall into A tends to a/b when N goes to infinity, and the convergence is independent of the dimension of the space, which is very useful in practice.

If now your space E is divided into (say) 10 boxes $E_j,\ j=1,...,10$, each of them of probability $1/10$, if you throw (say) 1000 points at random, it is very likely that each of the boxes will receive some of the points, but it is very unlikely that the points will be equally distributed among the boxes : some boxes will receive many, some will receive few. Randomness does not lead to equality, but to stronger and stronger differences between the boxes (contrarily to what most of people think), when the number of points increases.

If you want to cover a cube, or a hypercube, with balls of equal radius (covering problem), a random choice of these balls will not give a good result. If you throw them at random, you will need many more than if you put them carefully at selected places. As a particular case, if you want to devise an experience plan, for instance exploring a space for 50 parameters, with only 300 experiences, you should put the experiences at selected places : if fact, the centers of the 300 balls of minimal radius, which cover the cube of dimension 50. The problem is of geometrical nature : how to cover a cube of dimension 50 with 300 balls : what is their radius and where to put the centers ? If you throw them at random, you have less information (this is obvious already in dimension 2).

The same holds for packing problems, such as the "knapsack problem" : given some objects of known dimensions, how to put them together in a minimal volume ? Obviously, if you try to arrange them at random, you get a very poor result.

Randomness in algorithms gives good results, but only :

- When the result is related to measure theory ;
- Only when the sample is sufficient (recall that in order to increase the precision by 10 you need to increase the sample by 100).

Randomness will not give good results when :

- The number of points in the sample is constrained : if you have only 300 possible experiments, for some problem, you have better put them in an intelligent manner than at random ;
- If you are looking for the solution of a specific optimization problem.

In fact, randomness insures that, sooner or later, any region of the space will be penetrated. So, if for your problem there is a "good" region, you will find it, sooner or later, but it might be very late, and you will have first found a large number of bad regions. So, for all problems which require delicate use of a small sample, randomness should be avoided in practice.

It should also be avoided in theory, if there is no demonstration that the algorithm converges. A solution proposed by a mathematician differs from a solution of empirical nature, or from a black box, in the sense that we are normally capable of establishing a theory (at least in a limited manner) which substantiates the algorithm and "proves" that the result we are establishing really holds. At a preliminary stage, of course, all trials may be considered, but we should not release any tool if we cannot prove something about this tool.