IMPLEMENTATION OF GENETIC ALGORITHMS ON THE MULTIPROCESSOR SYSTEM MBC-1000

V.V. Korneev, A.P. Kovalenko, A.I. Masalovich

Scientific Research Institute KVANT,

15, the 4-th Likhachevsky Lane, 125438, Moscow, RUSSIA

tel:(095)1534700

fax:(095)1541418

e-mail:korv@a5.kiam1.rssi.ru

The most crucial parameter of the genetic algorithm is the population contents. The idea of unrolling the genetic algorithm "in space" is first of all related to the use of a multiprocessor computer system. A population is subdivided into several fragments each of them being processed for a long enough time. After that there comes the moment when all the processors exchange the parts of populations being processed. Strange as it may seem, the exchange structure has the secondary meaning while the fact of echange plays the decisive role. One may use the cycle exchange procedure, and the populations may only exchange 20% of its best solutions. After that the genetic algorithm continues to work

with the renewed populations. The exchange of informationis is an infrequent event occuring once in about 100--1000 iterations.

Therefore, the time expenses for it are negligibly small.

To estimate the efficiency of using a multiprocessor computer system let us consider two border cases assuming that the computing resource has p processors.

Let it be known that the genetic algorithm for the problem of a given dimension converges, on the average, for some time t while using a population of size d. Let us place on each of p processors an independent population of size d. The experimental results showed that in this case the time gain is proportional to sqrt{p}.

On the other hand, if we try to do without increasing the size of the total population and just to allocate the initial population of size d among p processors, then each of them will de-

al with the population of size d1=d/p. In this case the time gain is proportional to the number of processors squared, if d1 is large enough to prevent the population from degeneration. It should be noted that in this case the number of iterations of the genetic algorithm, necessary to reach the satisfactory solution, may somewhat increase, but in any case it is not comparable with the gain provided by parallel computer systems.

Thus, one may consider the obtained results as the lower and the upper efficiency bounds. In practice, however, the above approaches are seldom used as they are.

The parallel genetic algorithm makes it possible to solve problems whose dimension exceeds the admissible limits for serial processors.