PARALLEL ALGORITHMS FOR THE CREATION OF MEDICAL DATABASE SYSTEMS

RAKHIMOV BAKHTIYAR SAIDOVICH

Doctor of technical science, Urgench branch of Tashkent Medical Academy, Uzbekistan

MEKHMANOV MUKHIDDIN SULTANOVICH

Doctor of chemical science, Urgench branch of Tashkent Medical Academy, Uzbekistan

BEKCHANOV BAKHTIYAR GAFUROVICH

Doctor of medical science, Urgench branch of Tashkent Medical Academy, Uzbekistan

Analysis of parallel algorithms for graphics processors allows you to determine the bottlenecks of the algorithm that affect its performance on a particular computing system. Algorithms can be analyzed both at the AGM level and at the level of the CPU-GPU system. Despite the fact that the model does not reflect the overlap of the execution of computational operations and operations of access to the global memory, but estimates the upper limit of the execution time of the algorithm on the AGM, in any parallel algorithms for the AGM it is necessary to reduce this parameter of the algorithm due to the large number of clock cycles spent by the multiprocessor to implement global memory access operations.

When analyzing a parallel algorithm at the level of a graphics multiprocessor, you can use the methods of analyzing parallel algorithms for the PRAM model, which are based on the working and step complexity of the algorithm. A graphics multiprocessor contains a finite number of scalar processors, and even though software can emulate significantly more processors, you need to pay special attention to the operational complexity of the algorithm, which should be sufficient to cover the cost of accessing the global memory of the graphics processor.

- Thus, according to the proposed AGM model, when analyzing each stage of the algorithm, it is necessary to estimate the following set of step parameters:
- 1) the working complexity of the algorithm for the AGM. With small amounts of processed data;
- 2) the complexity of access to the global memory of the graphics processor of the parallel algorithm for the AGM.

1) total working complexity:

-
- $W(N) = \sum_{i=1}^{B(N)} W^i(N)$ (1)
- 2) total complexity of accessing global GPU memory:
- $R(N) = \sum_{i=1}^{B(N)} R^i(N)$ (2)