The Ministry of Education and Science of the Russian Federation

Lobachevsky State University of Nizhni Novgorod

Computing Mathematics and Cybernetics faculty

The competitiveness enhancement program

of the Lobachevsky State University of Nizhni Novgorod

among the world's research and education centers

Strategic initiative

“Achieving leading positions in the field of supercomputer technology

and high-performance computing”

INTRODUCTION TO PARALLEL PROGRAMMING

Lecture 19. Parallel Computation Modeling and Analysis

Nizhni Novgorod

2014

Lecture 19_. Parallel Computation Modeling and Analysis

The analysis of parallelism efficiency is a crucial point in the development of parallel algorithms for solving complicated research and engineering problems. Parallelism efficiency analysis is, as a rule, the evaluation of the computation process speedup (reducing the time needed for solving a problem). Forming the speedup estimation may be carried out for selected computational algorithm (the efficiency estimation of parallelizing a specific algorithm). Another important approach may be the construction of the maximum possible speedup estimation for the solution of a certain problem type (the efficiency estimation of the best parallel approach for solving a problem).

The lecture describes the computational model as "operations-operands" graph, which may be used for the description of the existing information dependencies in the selected algorithms of problem solving. The model is based on an acyclic-oriented graph, the vertices of which represent operations, and the arcs correspond to the data dependencies of operations . If such a graph is available, it is enough to set the schedule, according to which the distribution of the executed operations among processors is fixed, to define a parallel algorithm.

Representation of calculations with the help of such models allows us to analytically obtain a number of characteristics of the parallel algorithms being developed. Among those characteristics there is the execution time, the optimal schedule scheme, the estimates of maximum possible processing speed of the problem solving methods. The concept of paracomputer as a parallel system with an unlimited number of processors is considered for simpler constructing theoretical estimates.

In order to estimate the efficiency of the parallel computation methods we have discussed such widely used in theory and practice of parallel programming basic quality indicators as speedup and efficiency. Speedup shows how many times faster solving the problem is carried out, if several processors are used. Efficiency characterizes the fraction of time when the processors of a computing system are actually used. The cost of computations is also an important characteristic of the developed algorithm. It is defined as the product of parallel problem solving time and the number of processors used.

To demonstrate the applications of the models and the methods of parallel algorithm analysis we have considered the problem of finding the partial sums of a numeric value sequence. The example helps us to illustrate the problem of sequential algorithm parallelizing complexity. The complexity arises, as these algorithms are not initially oriented at the possibility of parallel computation arrangements. To demonstrate the "hidden" parallelism we have shown the possi-

bility of converting the initial sequential computation scheme and described the cascade scheme, which is obtained as a result of the conversion. Considering the same problem we have shown the possibility to introduce redundant computations for achieving greater parallelism in the executed computations.

In conclusion we have considered the problem of creating the estimates of maximum attainable values of efficiency criteria. Amdahl's law may be used for the creation of such estimates, which allows us to take into account the existing sequential (non-parallelilzed) computations in the problem solving methods. Gustafson-Barsis's law provides the construction of scaled speedup estimates used to characterize how efficiently parallel computations may be organized with the increase of problem complexity. To define the dependence between the problem complexity and the number of processors, the observation of which provides the necessary efficiency level of parallel computations, we have introduces the concept of the *isoefficiency function*.

Additional information on parallel computation modeling and analysis may be found in, for instance, Bertsekas and Tsitsiklis (1989). Useful information is also contained in Kumar et al. (1994), Quinn (2004).

The consideration of the academic problem of the numeric value sequence summation was carried out in Bertsekas and Tsitsiklis (1989).

For the first time Amdahl's law was stated in Amdahl (1967). Gustafson-Barsis's law was published in Gustavson (1988). The concept of isoefficiency was proposed in Grama et al. (1993).

A systematic discussion (for the time when the book was published) of the parallel computation modeling and analysis issues is given in Zomaya (1996).

Test questions

1. How is the "operations-operands" model defined?
2. How is the schedule for the distribution of computations among processors defined?
3. How is the time of parallel algorithm execution defined?
4. What schedule is optimal?
5. How can the minimum possible time of problem solving be defined?
6. What is a paracomputer? What can this concept be useful for?
7. What estimates should be used as the characteristics of the sequential problem solving time?
8. How to define the minimum possible time of parallel problem solving according to "operands-operations" graph?

9. What dependences may be obtained for parallel problem solving time if the number of processor being used is increased or decreased?

10. What number of processors corresponds to the parallel algorithm execution time (periods) comparable in the order with the estimates of minimum possible time of problem solving?

11. How are the concepts “speedup” and “efficiency” defined?

12. Is it possible to attain superlinear speedup?

13. What is the contradictoriness of the speedup and efficiency characteristics?

14. How is the concept of computation cost defined?

15. What is the concept of the cost-optimal algorithm ?

16. What does the problem of parallelizing a sequential algorithm of the numeric values summation lie in?

17. What is the essence of the summation cascade scheme? What is the aim of considering the modified version of the scheme?

18. What is the difference between the speedup and efficiency characteristics for the discussed versions of the summation cascade scheme?

19. What is the parallel algorithm of all the partial sums computation of a numeric value sequence?

20. How is Amdahl’s law formulated? Which aspect of parallel computation does it allow to take into account?

21. What suppositions are used to ground the Gustafson-Barsis's law?

22. How is the isoefficiency function defined?

23. Which algorithm is scalable? Give examples of methods with different level of scalability.

Practice

1. Develop a model and evaluate speedup and efficiency of the parallel computations:

- For the problem of the scalar product of two vectors

$$y = \sum_{i=1}^{N} a_i b_i \ ,$$

- For the problem of choosing the maximum and minimum values for the given set of numeric values

$$y_{\min} = \min_{i \le i \le N} a_i \ , \ y_{\max} = \max_{i \le i \le N} a_i \ ,$$

- For the problem of finding the mean value for the given set of numeric values

$$y = \frac{1}{N} \sum_{i=1}^{N} a_i \ .$$

2. Evaluate according the Amdahl's law the maximum attainable speedup for the problems given in 1

3. Evaluate the scalability speedup for the problems in 1

4. Construct the isoefficiency function for the problems given in 1

5. Work out a model and make a complete analysis of parallel computation efficiency (speedup, efficiency, maximum attainable efficiency, scalability speedup, isoefficiency function) for the problem of matrix – vector multiplication

References

1. **Amdahl, G.** (1967). Validity of the single processor approach to achieving large scale computing capabilities. In AFIPS Conference Proceedings, Vol. 30, pp. 483-485, Washington, D.C.: Thompson Books.
2. **Bertsekas, D.P., Tsitsiklis, J.N.** (1989). Parallel and distributed Computation. Numerical Methods. - Prentice Hall, Englewood Cliffs, New Jersey.
3. **Grama, A.Y., Gupta, A. and Kumar, V.** (1993). Isoefficiency: Measuring the scalability of parallel algorithms and architectures. IEEE Parallel and Distributed technology. 1 (3). pp. 12-21.
4. **Gustavson, J.L.** (1988) Reevaluating Amdahl's law. Communications of the ACM. 31 (5). pp.532-533.
5. **Kumar V., Grama, A., Gupta, A., Karypis, G.** (1994). Introduction to Parallel Computing. - The Benjamin/Cummings Publishing Company, Inc. (2nd edn., 2003)
6. **Quinn, M. J.** (2004). Parallel Programming in C with MPI and OpenMP. – New York, NY: McGraw-Hill.