

PACS 07.05.Tp, 02.70.-c, 02.60.Cb

DOI: 10.22363/2658-4670-2019-27-3-263-267

# Computational experiment in era of HPC

**Alexander S. Ayriyan**<sup>1,2</sup><sup>1</sup> *Laboratory of Information Technologies  
Joint Institute for Nuclear Research**6 Joliot-Curie St., Dubna, Moscow Region 141980, Russian Federation*<sup>2</sup> *Computational Physics and IT Division**A. I. Alikhanyan National Science Laboratory, Yerevan, Armenia*

(received: December 19, 2019; accepted: December 23, 2019)

In this note we discuss the impact of development of architecture and technology of parallel computing on the typical life-cycle of the computational experiment. In particular, it is argued that development and installation of high-performance computing systems is indeed important itself regardless of specific scientific tasks, since the presence of cutting-age HPC systems within an academic infrastructure gives wide possibilities and stimulates new researches.

**Key words and phrases:** computational experiment, high performance computing systems, Mathematical modelling

## 1. Introduction

The scheme of a computational experiment is the procedure for carrying out research of physical phenomena and technical devices based on the triad of mathematical modeling: `model` → `algorithm` → `program` [1], [2]. More often, the study is conducted for granted; in this sense, the scheme of a computational experiment is retrospective. However, to some extent, it is useful to consider such a scheme in detail for both creating an overall picture of research *a posteriori* and planning it *a priori*.

## 2. Stages of computational experiment

A computational experiment is based on *a priori* knowledge of the object under study, namely, theoretical and empirical data, established and generally accepted phenomenological models. Each cycle of a computational experiment consists of the following stages:

1. Building a model of the considered phenomenon under study (making up equations describing the phenomenon).



2. Formulation of a mathematical problem (formulating initial, boundary value, initial boundary value problems, optimization problems, etc.).
3. Choosing numerical methods for solving the problem (building difference schemes, choosing and/or developing algorithms, parallelization of calculations, etc.).
4. Realization of a program that implements the algorithms.
5. Choosing a suitable computing architecture, compiling and running programs on a computing system (selecting a compiler, collecting libraries of necessary algorithms).
6. Carrying out calculations and processing the obtained data (bringing the output data to the form necessary for their further analysis).
7. Analyzing the results; if possible, comparing them with a full-scale experiment.

### 3. Description of the stages

Quite often, at the last stage one starts to understand what changes should be made to the previous stages. At the same time, the cycle of a computational experiment begins once again, with the first stage, when some changes should be made to the model of the considered phenomenon, if the built model does not reflect well enough the peculiarities of the phenomenon or it becomes necessary to make changes to the model parameters.

The cycle can be repeated in a shorter form, for example, from the second stage, if any changes need to be made to the mathematical formulation of the problem. It may turn out that the used numerical methods have some drawbacks in terms of accuracy, efficiency or applicability; in this case, one will have to start the cycle of a computational experiment from the third stage.

The cycle of a computational experiment will begin with the fourth stage if the software implementation turns out to be ineffective in terms of calculation speed, code readability, its extensibility and scalability; in this case, one will have to make changes to the existing code or choose another programming environment.

Due to the unsatisfactory time of calculations, one may need to choose another computing environment that has more RAM, a better data transfer interface (in case of parallel computing), a more suitable file system, a better architecture and/or a set of compilers, which, in turn, can significantly reduce the estimated time and increase the efficiency of the cycle of a computational experiment. In this case, the cycle starts from the fifth stage, as there is no need to make changes to stages 1 to 4.

The exit from the cycle of a computational experiment happens when, after analyzing the results, the study is considered complete.

In the literature the fifth stage is not included in the description of a computational experiment (“Computational system” in Figure 1) (see, for example, [2], [3]), however, due to the development of novel computing architectures showing high efficiency in solving certain classes of problems, and to the increase of their availability, it becomes necessary to explicitly include this stage in the cycle of a computational experiment. When a researcher has access to

hybrid computational systems, such as HybriLIT [4], [5] containing different computing architectures, it gives him wide potential for the study.

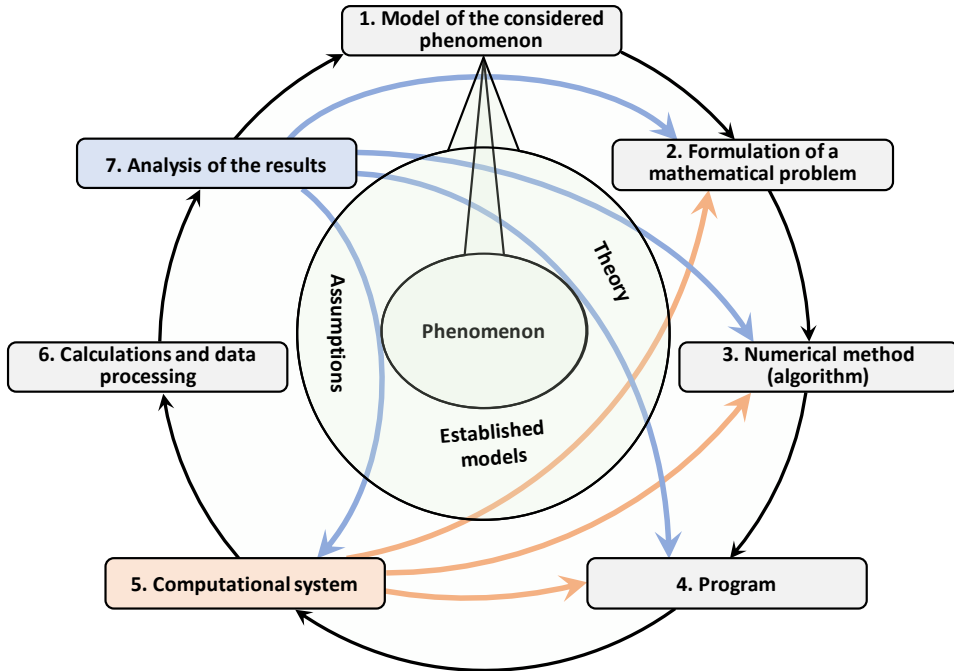


Figure 1. The scheme of a computational experiment

In this case, it becomes possible to reformulate the mathematical problem to another, which will require large computational costs, but will be more suitable for carrying out research, for example, to replace the initial boundary value problem with an optimization problem if the model is parameterized and it is possible to formulate an optimization criterion (see, for example, [6]).

The choice of a computational method becomes wider; one can choose a method that is easier to implement, but more effective with a larger amount of computing resources. One of the great illustrations is the direct enumeration (or search) method; if there are sufficient computing resources, such a method can be extremely effective in terms of the calculation time. Finally, the characteristics of a computing system can directly influence the software implementation of computational methods.

#### 4. Conclusion and discussion

Thus, one may conclude that the efficiency of a computational experiment directly depends on the available computing ecosystem. From this it follows that an HPC ecosystem must be developed regardless of the demands and tasks, since it itself affects the development of scientific studies.

## References

- [1] “Mathematical modeling and computational experiment [Matematicheskoye modelirovaniye i vychislitel’nyy eksperiment],” *Bulletin of USSR Academy of Science [Vestnik AN SSSR]*, vol. 49, no. 5, A. A. Samarsky, Ed., pp. 38–49, 1979, in Russian.
- [2] “Computational experiment in technology problems [Vychislitel’nyy eksperiment v zadachakh tekhnologii],” *Bulletin of USSR Academy of Science [Vestnik AN SSSR]*, vol. 54, no. 3, A. A. Samarsky, Ed., pp. 77–88, 1984, in Russian.
- [3] M. M. Gorbunov-Posadov, *Extensible programs [Rasshiryayemyye programmy]*. Moscow: Polyptych, 1999, in Russian.
- [4] HybriLIT group. (2019). Heterogeneous platform “HybriLIT”, [Online]. Available: <http://hlit.jinr.ru/en/>.
- [5] G. Adam *et al.*, “IT-ecosystem of the HybriLIT heterogeneous platform for high-performance computing and training of IT-specialists,” English, in *CEUR Workshop Proceedings*, V. Korenkov, A. Nechaevskiy, T. Zaikina, and E. Mazhitova, Eds., vol. 2267, 2018, pp. 638–644.
- [6] A. Ayriyan, J. Buša Jr., H. Grigorian, and E. E. Donets, “Solving the optimization problem for designing a pulsed cryogenic cell,” *Physics of Particles and Nuclei Letters*, vol. 16, no. 5, pp. 300–309, 2019. DOI: 10.1134/S1547477119030026.

### For citation:

A. S. Ayriyan, Computational experiment in era of HPC, *Discrete and Continuous Models and Applied Computational Science* 27 (3) (2019) 263–267. DOI: 10.22363/2658-4670-2019-27-3-263-267.

### Information about the authors:

**Alexander S. Ayriyan** (Russian Federation) — Researcher of the Laboratory of Information Technologies, Joint Institute for Nuclear Research (Dubna, Russia), and Computational Physics and IT Division, A.I. Alikhanyan National Science Laboratory (Yerevan, Armenia) (e-mail: [ayriyan@jinr.ru](mailto:ayriyan@jinr.ru), ORCID: <https://orcid.org/0000-0002-5464-4392>, ResearcherID: O-5259-2015, Scopus Author ID: 24528624400)

PACS 07.05.Tr, 02.70.-с, 02.60.Cь

DOI: 10.22363/2658-4670-2019-27-3-263-267

## Вычислительный эксперимент в эпоху HPC

А. С. Айриян<sup>1,2</sup>

<sup>1</sup> *Научный отдел вычислительной физики*

*Лаборатория информационных технологий*

*Объединённый институт ядерных исследований*

*ул. Жолио Кюри, д. 6, г. Дубна, Московская область, 141980, Россия*

<sup>2</sup> *Научный отдел вычислительной физики и ИТ*

*Национальная научная лаборатория им. А. Алиханяна, Ереван, Армения*

В данной заметке обсуждается влияние развития архитектуры и технологии параллельных вычислений на цикл вычислительного эксперимента. В частности, делается вывод, что разработка и установка высокопроизводительных вычислительных систем действительно важна сама по себе, вне зависимости от конкретных научных задач, поскольку наличие современных высокопроизводительных вычислительных систем в академической инфраструктуре даёт широкие возможности и стимулирует новые исследования.

**Ключевые слова:** вычислительный эксперимент, высокопроизводительные вычислительные системы, математическое моделирование