Ontological approach to formalization of knowledge in computational plasma physics

A V Snytnikov\textsuperscript{1,2}, B M Glinskiy\textsuperscript{3}, G B Zagorulko\textsuperscript{3}, Y A Zagorulko\textsuperscript{3}

Institute of Computational Mathematics and Mathematical Geophysics SB RAS
Institute of Automation and Electrometry SB RAS
A.P. Ershov Institute of Informatics Systems SB RAS

Corresponding author: snytav@gmail.com

Abstract. Computational plasma physics is a wide area of research, which includes a certain set of physical phenomena, mathematical equations, numerical methods, programming strategies, and hardware architectures that directly follow each other. To design an efficient code for solving a computational plasma physics problem, the researcher should have a clear understanding of the relationships between the concepts of physics, mathematics, computer science, and computer architecture. Ontology is one of the effective ways to realize (provide) such an understanding. Authors propose an approach to design an ontology which will show how physical phenomena determine mathematical equations being used, how the equations define numerical methods, and how methods enable programming strategies to form an architecture-efficient implementation of problem solution.

1. Introduction

Currently, ontologies are the main means for formalizing and systematizing scientific knowledge and data in various fields. In the modern sense, an ontology is a precise detailed specification (model) of a certain part of the world regarding to a specific area of interest. The basis of the ontology is a set (vocabulary) of terms that serve to name such domain entities as classes, attributes, relations, functions, and individuals (class instances or concrete objects). Ontology also includes definitions of terms (in natural and/or formal language) and axioms that constrain the possible interpretations of the defined terms and support the correct use of them [1].

The ontology of a scientific area usually describes the objects and subjects of research typical of it, the problems solved in it, the methods of research and problem solving used in it, the scientific activity carried out in it and the scientific results obtained in its framework. It should be noted that for the practical application of the ontology of a specific scientific area, it is critically important to present in it knowledge that describes not only this area, but also the most important entities of adjacent areas.

This paper is devoted to the development of an ontology for such a scientific area as computational plasma physics (CPP). This area covers a fairly large range of physical phenomena and processes described by many mathematical equations, which are solved by certain numerical methods. Implementation of these methods requires computers with a high level of performance, appropriate architectures and software strategies. To develop efficient code for solving a computational plasma physics problem, a researcher must have a clear understanding of the relationships between the concepts of physics, mathematics, computer science, and computer architecture theory. Ontology can provide such an understanding and give detailed answers to the following questions: what mathematical equations can be used to model the studied physical phenomena, what numerical methods can solve these equations, what software strategies need to be applied to get effective implementations of these methods on the existing architecture.

The considered ontology is a part of a more general ontology built for an intelligent support system for solving compute-intensive problems of mathematical physics on supercomputers [2, 3]. The paper briefly
describes the basic concepts of the general ontology. The main attention is paid to concretization of general concepts regarding to CPP. When constructing the CPP ontology, a number of ontology design patterns (ODPs) [4] were developed. The ontology is formalized in the OWL language [5] using the Protégé editor [6]. The paper provides an example of constructing semantic chains connecting different types of plasma and the processes occurring in this area with parallel algorithms, technologies and architectures.

2. Ontology of intelligent support for solving compute-intensive problems of mathematical physics on supercomputers

Let us consider the upper level of the ontology of intelligent support for solving compute-intensive problems of mathematical physics (see Fig. 1). The main Objects of Research of this subject domain (SD) are Physical Objects and Physical Phenomena, which are studied in certain Branch of Science and are based on the Fundamental Laws of Nature, in turn derived from Experiments and Observations. Objects of Research are modeled by a Physical Model, which is described by a Mathematical Model in the form of a certain Equation System. The Equation System is solved by the Numerical Method, which is implemented by one or another Parallel Algorithm optimized for the Target Architecture and represented using a certain Parallel Programming Technology. The final representation of the Parallel Algorithm is coded by the program Code, which includes a set of Program Components and is executed on the Target Architecture.

![Figure 1. An upper level of the ontology of intelligent support for solving compute-intensive problems of mathematical physics](image)

This ontology contains the basic concepts of the subject domain under consideration and their relationships. To describe the properties of concepts, a system of ontology design patterns was developed. Such patterns represent documented descriptions of practical solutions to typical problems of ontological modeling [7]. They are designed to simplify and facilitate the ontology building process and help developers avoid frequently repetitive errors of ontological modeling.
Let us consider as an example the pattern for describing the Parallel Algorithm class (Fig. 2). This pattern belongs to the group of content patterns. In essence, such a pattern is a semantic neighborhood of the central concept, which in this case is the Parallel Algorithm class. For this concept, properties, such as attributes and relations, are defined. Attributes are represented as data properties for those properties whose values have a standard data type (Name, Description, Scalability Evaluation, etc.), or as object properties for properties with values from an enumerated data type (Domain Decomposition). Relations define links of objects of the considered class with objects of other classes and are represented as object properties. Some relations can have their own attributes. In Fig. 2 such relations are highlighted in bold and light blue background. So, the isDetermined relation defined between Parallel Algorithm and Data Structure classes has the string attribute Description and the hasInput and hasOutput relations have string attributes specifying the Format and Volume of Data.

![Figure 2. The pattern for describing the Parallel algorithm](image)

### 3. Ontology of computational plasma physics

Based on the ontology described above, the ontology of the subject domain "Computational Plasma Physics" [8] was built by specifying and supplementing its concepts. Thus, the main Objects of Research in the CPP are Plasma, Phenomena and Processes in it [9].

Plasma is present in space (solar crown, ionized gas in interstellar medium, etc.) and in the laboratory (fusion devices and plasma reactors used in microelectronics). In accordance with this, two main classes of Plasma are distinguished in the ontology. These are Cosmic Plasma and Laboratory Plasma. Plasma might be cold (e.g. 1000 K) or hot (over 1 million degrees). Plasma also differs in density from $10^{15}$ cm$^{-3}$ in fusion devices like tokamak to 1 particle per cubic centimeter (or even less) in interstellar gas.

Another property of Plasma is the degree of its ionization. To describe these properties of Plasma, the attributes Temperature, Density, Degree of Ionization with values from the enumerated type (high, medium, low) were defined.

Processes in plasma are systematized according to the types of plasma. For example, such processes as Burst in Solar Crown, Gas Discharge in Plasma, Controlled Thermonuclear Fusion are distinguished.

In the area under consideration, the problems of modeling the phenomena and processes occurring in plasma are being solved. To do this, various physical models such as Kinetic, Hydrodynamic, and Hybrid are used. The Physical Model is described by the Mathematical Model, the role of which one or another Equation System plays.

Plasma is generally described by Vlasov Equation [10], which is the Phase Volume Conservation Law (electromagnetic field is determined by Maxwell Equation):
\[
\frac{\partial f_{i,e}}{\partial t} + v \frac{\partial f_{i,e}}{\partial r} + \mathbf{F}_{i,e} \frac{\partial f_{i,e}}{\partial p} = 0, \quad \mathbf{F}_{i,e} = q_{i,e} \left( \mathbf{E} + \frac{1}{c} \left[ \mathbf{v} \times \mathbf{B} \right] \right),
\]
\[
\text{rot } \mathbf{B} = \frac{4\pi}{c} + \frac{1}{c} \frac{\partial \mathbf{E}}{\partial t},
\]
\[
\text{rot } \mathbf{E} = -\frac{1}{c} \frac{\partial \mathbf{B}}{\partial t},
\]
\[
\text{div } \mathbf{E} = 4\pi p,
\]
\[
\text{div } \mathbf{B} = 0.
\]

Here \( f \) is the distribution function (number of real particles in some definite point with some definite velocity), or phase density. Indices \( i \) and \( e \) refer to ion and electron components of plasma, \( r \) is coordinate and \( v \) is the velocity, \( p \) gives momentum, \( \mathbf{F} \) is the force acting on each particle, and depending on electric field \( \mathbf{E} \) and magnetic field \( \mathbf{B} \). Note that the right-hand-side is 0, which means no collisions (collisions are neglected).

**Vlasov Equation** is the most general **Mathematical Model** which is correct in all cases. But since the numerical solution of **Vlasov Equation** in real 2D or 3D cases is extremely time-consuming, a more simple way is required. It is the hydrodynamical description: if the local distribution function is close to Maxwellian, then **Vlasov Equation** could be integrated by velocities resulting in the hydrodynamic set of equations. Since it also involves magnetic field equations, it is called **Magneto Hydrodynamic (MHD) Equations**.

One or another numerical method is used to solve the selected system of equations. For example, the system of equations describing the kinetic model can be solved either using the **Particle-In-Cell Method** [11,12], or using the method of **Direct Solution of Vlasov Equation** [13], or using **Monte Carlo Methods** [14]. In some cases, both the **Particle-In-Cell Method** and the **Monte Carlo Method** can be applied to resolve the mathematical model [15, 16].

As can be seen from Fig. 1, the **Numerical method** is implemented by the **Parallel Algorithm** encoded by the program **Code**. These two classes already have direct connections with **Parallel Programming Technologies** and **Parallel Architectures**.

To formalize the ontology, the Protégé editor was used [6]. In Fig. 3 in the left part of the editor window the concepts of the upper level of the ontology are presented, and the properties of the **Ionized Interstellar Gas** object are in its upper right part.

**Figure 3.** The window of the Protégé editor.
The presented ontology and the means for its formalization make it possible to systematize the knowledge of the CPP subject domain. However, some of the knowledge may not be contained explicitly in the ontology, but they may be deduced by built-in inference engines based on special rules. Let us consider the following example. To solve the problems of modeling objects and phenomena of the CPP subject domain, it is necessary, first of all, to select an adequate physical model. It is known that a low-temperature plasma with a low density strongly requires a kinetic description, whereas a highly ionized plasma allows to use a hydrodynamic description. Such knowledge can be presented in the form of If-Then rules in the SWRL language [17], which is supported by the Protége editor. In the lower right part of Fig. 3, two such rules are shown, and in the upper right part of this figure, the result of the interpretation of these rules for Ionized Interstellar Gas is highlighted in yellow.

In a like manner, you can describe the rules that allow you to determine the appropriate Architecture Elements according to the requirements imposed on them by Numerical Methods, Parallel Architectures and program Codes that must be executed on these elements. Thus, the methods for solving the MHD Equations should be implemented mostly for classical processors (CPUs), since these methods require a fairly large amount of RAM together with large Cache Memory and random access, though GPU implementation is also possible. Sometimes it is good to implement these methods also using MIC (Many Integrated Cores, a powerful coprocessor from Intel), although, as shown by Kulikov [18], it requires a very low level of programming. The direct solution of the kinetic equation (explicit numerical scheme) can be efficiently implemented by hardware such as FPGA (Field Programmable Gate Array [19]). But it should be remembered that the direct solution of the Vlasov Equation requires a very large amount of memory [20]. In works [21-25] it is shown that the Particle-In-Cell Method, another method of kinetic solution of problems with plasma, is effectively used for graphics processing units (GPUs).

4. Solving problems of computational plasma physics
The interconnected objects of the ontology form a semantic network, on which, as has been shown, one can perform logical inference and deduce facts that are not explicitly presented in the ontology. In addition, it is possible to organize navigation through the semantic network by traversing from one object to another. For example, you can build a chain from the problem that the user is investigating to parallel architectures, technologies, and codes which the user can employ to solve this problem.

Fig. 4 shows the semantic chain for solving the problem of modeling Ionized Interstellar Gas (the elements of the chain are highlighted in green). Since the interstellar gas belongs to a low-temperature and rarefied plasma, the Kinetic Model is used to describe it. This model can be described by the Vlasov Equation. An effective method for solving it is the Direct Method implemented by the Explicit Scheme with 3D-Domain Decomposition.

The scheme is programmed with Code that can be executed on both the CPU and GPU using FPGA and Parallel Programming Technologies such as MPI, Open MP, and CUDA.

5. Conclusion
The paper presents the ontology of the subject domain "Computational Plasma Physics". This ontology is intended to be used as the backbone for the knowledge base of an intelligent system for supporting the solution of compute-intensive problems of mathematical physics on supercomputers. When constructing the ontology, ontology design patterns were developed. They were then used for a convenient and consistent description of specific objects representing the concepts of the subject domain. In addition, the rules that allow using logical inference to obtain facts which are not explicitly presented in the ontology have been developed. These rules work under a semantic network formed by a collection of interconnected objects. It is also possible to organize navigation through the semantic network by traversing from one object to another.

The paper provides an example of a semantic chain that allows for an Ionized Interstellar Gas to determine the Physical and Mathematical Model describing it, a suitable Numerical Method, Parallel Algorithm implemented this method, a special Code programming this algorithm, as well as the Parallel Programming Technologies and Architecture Elements on which this Code will be executed. This example demonstrates how, having a formalized structured description of the subject area and the specification of the type of plasma and / or the processes occurring in it, user can obtain information about methods, software and parallel architectures and technologies designed to solve related problems.

Acknowledgements
«The study of high-performance algorithms for computational plasma physics was carried out under state contract with ICM MG SB RAS (0315-2019-0001), the study of ontology was financially supported by

Acknowledgements
«The study of high-performance algorithms for computational plasma physics was carried out under state contract with ICM MG SB RAS (0315-2019-0001), the study of ontology was financially supported by
RFBR (grant No. 19-07-00085 and No. 19-07-00762). The development of plasma physics code was financially supported by RFBR (grant No. 18-07-00364).

Figure 4. The semantic chain for solving a problem of modeling Ionized Interstellar Gas

References
[1] Sharman R, Kishore R and Ramesh R 2006 Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems (Boston: Springer) p 930
[2] Glinskiy B, Zagorulko Y, Zagorulko G, Kulikov I and Sapetina A 2019 The Creation of Intelligent Support Methods for Solving Mathematical Physics Problems on Supercomputers *Communications in Computer and Information Science Proc. Supercomputing. RuSCDays* vol 1129 ed V Voevodin and S Sobolev pp 427-438

[3] Zagorulko G, Zagorulko Y, Glinskiy B and Sapetina A 2019 Ontological Approach to Providing Intelligent Support for Solving Compute-Intensive Problems on Supercomputers *Communications in Computer and Information Science Artificial Intelligence. Proc. RCAI 2019* vol 1093 ed S Kuznetsov and A Panov (Cham: Springer) pp 363-375

[4] Gangemi A, Presutti V 2009 Ontology Design Patterns *Handbook on Ontologies* ed S Staab and R Studer (Berlin: Springer Verlag) p 767

[5] Musen M A 2015 The Protégé Project: A Look Back and a Look Forward *AI Matters* 1(4) pp 3-12.

[6] Hockney R W, Eastwood J W 1988 *Computer Simulation Using Particles* (Boca Raton, Florida, USA: CRC Press) p 540

[7] Grigoryev Yu N, Vshivkov V A, Fedoruk M P 2002 *Numerical “Particle-in-Cell” Methods* (Utrecht–Boston: VSP) p 249

[8] Birdsell C, Langdon A 2005 *Plasma physics via computer simulation* (Bristol: Institute of Physics Pub) p 479

[9] Krall N, Trivelpiece A 1973 *Principles of Plasma Physics* (McGraw-Hill) p 674

[10] Vlasov A 1945 Theory of Vibrational Properties of an Electron Gas and Its Applications *Scientific notes of the Moscow State University, Physics* vol 75

[11] Vahedi V, Surendra M 1995 A Monte Carlo collision model for the particle-in-cell method: applications to argon and oxygen discharges *Computer Physics Communications* 87 (1-2) pp 179-198

[12] Burau H, Widera R, Honig W, Juckeland G, Debus A, Kluge T, Schramm U, Cowan T E, Sauerbrey R, Bussmann M 2010 *PIConGPU: a fully relativistic particle-in-cell code for a GPU cluster* *IEEE Transactions on Plasma Science* 38 (10) pp 2831–2839

[13] Rossi F, Londrillo P, Sgattoni A, Sinigardi S, Turchetti G 2012 Towards robust algorithms for current deposition and dynamic load-balancing in a GPU particle in cell code *AIP Conference Proceedings* vol 1507 (1) p 1-15

[14] Kong X, Huang M, Ren Ch, Decyk V 2011 Particle-in-cell simulations with charge-conserving current deposition on graphic processing units *Journal of Computational Physics* 230 (4) p 1676–1685

[15] Rieke M, Trost T, Grauer R 2015 Coupled Vlasov and two-fluid codes on GPUs *Journal of Computational Physics* 283 p 436–452

[16] Lotov K, V. Timofeev I, Mesyats E A, Snytkov A V, Vshivkov V A 2015 Note on quantitatively correct simulations of the kinetic beam-plasma instability *Physics of Plasmas* 22 (2) 024502