

GURU: КОНЦЕПЦИЯ ИНТЕГРИРОВАННОГО ВЫЧИСЛИТЕЛЬНОГО ОКРУЖЕНИЯ ДЛЯ ВЫСОКОПРОИЗВОДИТЕЛЬНОГО МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ И МАШИННОГО ОБУЧЕНИЯ

В.П. Ильин*

*Институт вычислительной математики и математической геофизики СО РАН,
Новосибирск, Российская Федерация
E-mail: ilin@sscc.ru

Аннотация. Рассмотрены концепция, архитектура и принципы реализации интегрированного вычислительного окружения (ИВО) для высокопроизводительного моделирования и машинного обучения на суперкомпьютерах новых поколений. Ядром данного программного обеспечения является базовая система моделирования, включая современную алгоритмическую и технологическую поддержку всех основных стадий решения больших прикладных проблем, а также функциональное наполнение методов оптимизации для обратных задач, реализующих принципы интеллектуальных инноваций. В целом ИВО представляет собой многоуровневую систему с широким развивающимся набором математических инструментов, в том числе средств автоматизации распараллеливания алгоритмов и создания гибких конфигураций для приложений, баз активных знаний для предметных областей, архивов задач и расчётных данных, конструкторов и анализаторов сценариев, средств принятия решений и т.д. Проект ориентирован на длительный жизненный цикл и эффективные применения, которые должны обеспечить гибкое расширение состава реализуемых моделей и применяемых методов, адаптацию к эволюции компьютерных платформ, переиспользование внешних продуктов и согласованное участие разных групп разработчиков.

Ключевые слова: математическое моделирование, высокопроизводительные вычисления, интегрированное программное окружение, интеллектуальные инновации, адаптация к эволюции платформ, эффективные применения алгоритмов

Финансирование. Исследование проведено в рамках госзадания ИВМиМГ СО РАН НИР–2022–0001.

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Ссылка для цитирования: Ильин В.П. GURU: Концепция интегрированного вычислительного окружения для высокопроизводительного математического моделирования и машинного обучения // Вестник РАН. 2026. Том 96. №2. С. 145–155. DOI: 10.7868/S3034520026020045



Поступила в редакцию 23.09.2025
После доработки 27.10.2025
Принята к публикации 15.11.2025

ИЛЬИН Валерий Павлович — доктор физико-математических наук, главный научный сотрудник лаборатории вычислительной физики ИВМиМГ СО РАН, Новосибирск, Российская Федерация, e-mail: ilin@sscc.ru, <https://orcid.org/0009-0000-0840-03>.

GURU: A CONCEPT OF AN INTEGRATED COMPUTATIONAL ENVIRONMENT FOR HIGH-PERFORMANCE MATHEMATICAL MODELING AND MACHINE LEARNING

V.P. Ilyin*

*Institute of Computational Mathematics and Mathematical Geophysics SB RAS,
Novosibirsk, Russian Federation*

**E-mail: ilin@sscc.ru*

Abstract. This article examines the concept, architecture, and implementation principles of an integrated computing environment (ICE) for high-performance modeling and machine learning on next-generation supercomputers. The core of this software is the Basic Modeling System (BMS), which includes modern algorithmic and technological support for all key stages of solving large-scale applied problems, as well as functional support for optimization methods for inverse problems implementing the principles of intelligent innovation. Overall, the ICE is a multi-level system with a wide, evolving set of mathematical tools, including automated algorithm parallelization and the creation of flexible application configurations, domain-specific active knowledge bases, problem and computational data archives, scenario designers and analyzers, decision-making tools, and more. The project is focused on a long lifecycle and effective applications, which should ensure flexible expansion of the implemented models and methods, adaptation to the evolution of computer platforms, the reuse of external products, and the coordinated participation of various development teams.

Keywords: mathematical modeling, high-performance computing, integrated software environment, intelligent innovation, adaptation to platform evolution, efficient application of algorithms

Funding. The study was conducted within the framework of the state assignment NIR-2022-0001 to the Institute of Computational Mathematics and Mathematical Geophysics of the Siberian Branch of the Russian Academy of Sciences.

Conflict of interest. The author declares that he has no conflict of interest.

For Citation: Ilyin V.P. GURU: a concept of an integrated computational environment for high-performance mathematical modeling and machine learning. *Herald of the Russian Academy of Sciences*. 2026, vol. 96, no.2, pp. 145–155. (In Russ.) DOI: 10.7868/S3034520026020045

Received September 23, 2025

Revised October 27, 2025

Accepted November 15, 2025

ILYIN Valery P. – Doctor of Physical and Mathematical Sciences, Chief Researcher of the Laboratory of Computational Physics of the Institute of Mathematics and Mathematical Geophysics of the Siberian Branch of the Russian Academy of Sciences, Novosibirsk, Russian Federation, e-mail: ilin@sscc.ru, <https://orcid.org/0009-0000-0840-03>.

Последнее десятилетие истории нашей цивилизации ознаменовали два события, которые невольно оказались завуалированы пандемией ковида и последующими политическими коллизиями международного масштаба. Однако научно-технический прогресс неотвратим, и в ближайшие годы он, несомненно, будет связан с массовым появлением постпетафлопсных и эксафлопсных компьютеров, а также с сопутствующим бурным развитием искусственного интеллекта на базе глубокого изучения процессов и явлений средствами математического моделирования с обработкой больших объёмов данных [1].

Эти направления уже имеют длительную историю развития. Актуальные вопросы человеко-машинного взаимодействия, автоматизации построения алгоритмов, их распараллеливания и отображения на архитектуру ЭВМ рассматривались ещё в пионерских работах Г.И. Марчука [2–4]. К настоящему времени в мире накоплены огромные объёмы математического и программного обеспечения в виде алгоритмических библиотек и проблемно-ориентированных пакетов прикладных программ, в том числе как дорогих коммерческих, так и свободно распространяемых. В качестве примеров отечественных и зарубежных продуктов можно назвать большие комплекты ЛОГОС [5], FLOW VISION [6], ANSYS [7], COMSOL [8], PFLOWTRAN [9], FENICs [10], а также библиотеки PARSOLVE [11], PETSc [12], HYPRE [13], NETLIB [14] и многие другие (по алгебраическим решателям содержательный обзор приведён в работе [15]).

В последние годы на смену такому многообразию разработок стали приходиться интегрированные программные окружения, ориентированные на инструментальную поддержку широких классов приложений: DUNE [16], OPEN FOAM [17], INMOST [18], проект ИВО [19], базирующийся на согласованном обеспечении всех основных технологических стадий математического моделирования. Перспективные тенденции таких методологий постпетафлопсных вычислений представлены в статье [20].

Настоящая работа посвящена развитию и конкретизации рассмотренных в [20] положений с точки зрения выработки концепции и архитектуры системы GURU – интегрированного вычислительного окружения для высокопроизводительного математического моделирования и машинного обучения на основе методов оптимизации сложных проектов и явлений. Предполагается, что рассматриваемые классы прикладных проблем можно описать многомерными краевыми и начально-краевыми междисциплинарными задачами для дифференциальных и/или интегральных уравнений в прямой или обратной формулировке.

В общем случае расчётные области имеют сложные конфигурации с кусочно-гладкими границами и контрастными материальными свойствами, обуславливающими наличие сингулярностей искомым решениям. Рассматриваемые непрерывные задачи дискретизируются с помощью различного типа структурированных или неструктурированных сеток, на которых строятся аппроксимации методами конечных разностей, конечных объёмов, конечных элементов или разрывных схем

Галёркина разных порядков точности. Используются соответствующие аббревиатуры: FDM, FVM, FEM, DGM [21]. Получаемые системы линейных алгебраических уравнений высокого порядка (10^8 – 10^{10} и более) с симметричными и несимметричными разреженными и плохо обусловленными матрицами (числа обусловленности 10^{13} и выше) решаются итерационными параллельными методами декомпозиции областей и многосеточными алгоритмами в подпространствах Крылова [22]. Для реализации обратных задач применены современные методы оптимизации на основе условной или безусловной минимизации целевого функционала [23].

В архитектурном плане платформа GURU представлена тремя основными уровнями, нижний из которых – это базовая система моделирования (БСМ), включающая функциональное наполнение технологических стадий вычислительного эксперимента. Второй уровень – средства формирования крупноблочных вычислительных схем для типовых уравнений математической физики (УМФ), в том числе междисциплинарных задач из различных отраслевых приложений, а верхний уровень содержит системное наполнение с интеллектуальными инструментами для автоматизации построения и распараллеливания алгоритмов, базы знаний, средств управления вычислительными процессами и принятия решений на основе анализа результатов расчётов.

В целом ИВО следует рассматривать как сложную экосистему со взаимосвязанными компонентами. Безостановочный научно-технический прогресс побуждает создавать математическое и программное обеспечение (МПО) с длительным жизненным циклом и динамическим развитием всех компонентов.

Система GURU имеет трёхцелевое назначение. Во-первых, это инструмент для кардинального повышения производительности труда разработчиков математического и программного обеспечения. Вторая цель заключается в эффективных инновациях с экономичным решением ресурсёмких фундаментальных и прикладных задач. В-третьих, рассматриваемые разработки должны предусматривать учебные версии для университетских образовательных практик и для переподготовки производственных специалистов новой формации при условии активного использования интеллектуального предсказательного эксперимента на основе глубокого изучения моделируемых объектов.

ОСНОВНЫЕ КОМПОНЕНТЫ И СТРУКТУРЫ ДАННЫХ БСМ

Базовая система моделирования является методоориентированной и состоит из достаточно автономных подсистем, каждая из которых отвечает за свою технологическую стадию математического моделирования и взаимодействует с другими посредством согласованных структур данных.

LANDAU – этап формирования функциональных данных математической модели, исходная информация для которой задана пользовательским интерфейсом (GUI – Graphic User Interface), а результат

формируется в виде функциональной структуры данных (FDS – Functional Data Structure). На содержательном уровне FDS включает информацию о типе решаемых уравнений, их коэффициентах, граничных и начальных условиях, имеющихся параметрах, о возможных априори известных свойствах и т.д.

VORONOI – формирование геометрической структуры данных (GDS – Geometric Data Structure) на основе пользовательской исходной информации о конфигурации расчётной области, её подобластей и границы, которая в общем случае может быть многосвязной и кусочно-гладкой.

DELAUNAY – подсистема дискретизации исходной расчётной области и генерации сеток, адаптивных к особенностям конфигурации границ. Исходной информацией в данном случае являются FDS и GDS, а результатом – сеточная структура данных (MDS – MeshDS), которая должна включать особенности эффективных блочных алгоритмов, в том числе декомпозиции областей и многосеточных методов. Методические и интеллектуальные аспекты реализации этой стадии рассмотрены в работе [24].

CHEBYSHEV – это стадия аппроксимации непрерывной модели задачи и формирования алгебраической структуры данных (ADS) на основе ранее сформированных FDS, GDS и MDS. Фактически данная подсистема представляет собой набор библиотек построения различных сеточных методов (FDM, FEM, FVM, DGM). В общем случае здесь применяются те или иные матричные форматы данных, позволяющие автоматизировать реализацию ресурсоёмких методов решения систем линейных и нелинейных алгебраических уравнений (СЛАУ или СНАУ [25]).

KRYLOV – библиотека программ для решения СЛАУ итерационными предобусловленными алгоритмами в подпространствах Крылова, включая современные высокопроизводительные методы декомпозиции областей и многосеточные подходы, распараллеленные средствами гибридного программирования на распределённой и общей памяти. Особенности структур данных и реализация алгоритмов изложены в работе [26].

KANTOROVICH – подсистема методов оптимизации для решения обратных задач, формируемых как проблема условной минимизации целевого функционала при наличии линейных и/или нелинейных ограничений. Соответствующая структура данных обозначается как ODS (Optimization Data Structure). Помимо прочего, рассматриваемые подходы составляют основу для машинного обучения. Вопросы параллельной реализации этой стадии, актуальной для интеллектуализации процессов моделирования, обсуждаются в работах [23, 27].

К перечисленным средствам необходимо добавить постпроцессинг, включающий ресурсоёмкие операции предварительной обработки числовых результатов, их визуализации и анализа.

Блок-схема взаимодействия функциональных компонентов и структуры данных БСМ приведена на рисунке 1. Отметим, что различные способы представления информации следует рассматривать как расширяемые и гибкие, с возможностью многих вер-

сий и создания «переходников» из одних форматов в другие. Данный программный комплекс является нижним уровнем функционального наполнения ИВО. С его помощью реализуется верхний уровень, который представляет собой программное обеспечение для решения типовых начально-краевых задач математической физики, описываемых, как правило, системами уравнений в частных производных (или эквивалентными интегральными формулами):

АН – (Aero-Hydro) – задачи аэрогидродинамики, представляемые уравнениями Навье–Стокса;

МНТ – (Mass-Heat Transfer) – уравнения тепло-массопереноса;

SDS – (Stress Deformation State) – задачи исследования напряжённо-деформированных состояний, включающие уравнения упругости Ламе;

ЕМ – (Electro-Magnetism) – задачи электромагнетизма, представляемые в общем случае системой уравнений Максвелла;

MPF – (Multi-Phase Filtration) – задачи многофазной фильтрации в пористых средах, включающие уравнения Дарси.

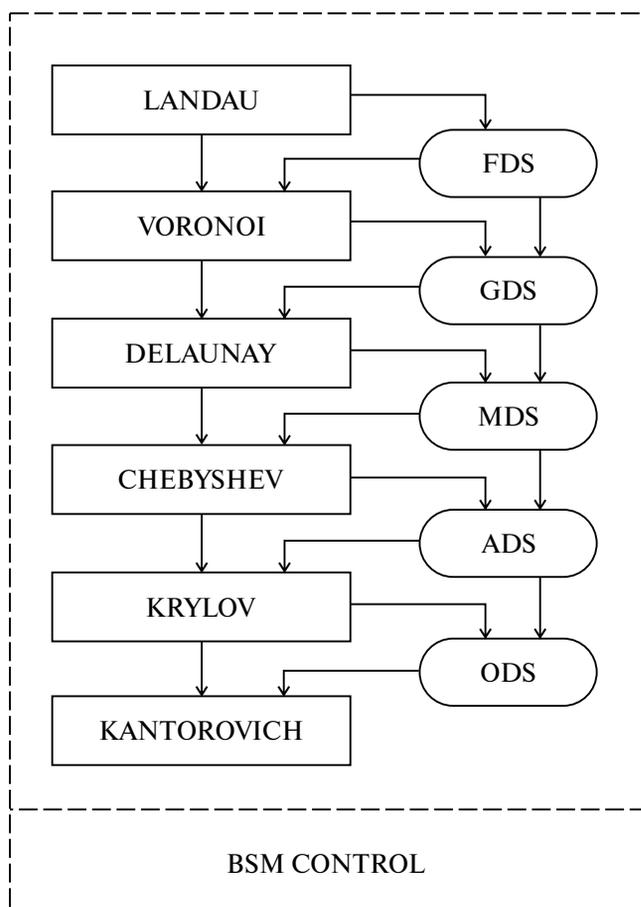


Рис. 1. Компоненты и структуры данных базовой системы моделирования

Fig. 1. Components and data structures of the basic modeling system

В реальных приложениях зачастую требуется решать междисциплинарные прямые и обратные задачи по моделированию процессов и явлений различной природы. Общая схема данной подсистемы представлена на рисунке 2. Отметим, что аналогичная классификация прикладных задач УМФ используется в платформе ЛОГОС [5], разработанной большим коллективом авторов из ВНИИЭФ (г. Саров). В предлагаемой нами архитектуре данный уровень (УМФ) экономично реализуется как программная надстройка над БСМ с эффективным использованием её богатых функциональных возможностей.

СИСТЕМНОЕ НАПОЛНЕНИЕ И ОБЩАЯ АРХИТЕКТУРА ИВО

Назначение системного наполнения – поддержка всех этапов жизненного цикла интегрированного вычислительного окружения, включая разработку и развитие его программно-информационных компонентов, а также эксплуатацию пользователями для практических целей и процессы обучения на университетских курсах или на производствах, использующих и развивающих суперкомпьютерные инновации.

В соответствии со сложившимися технологиями крупномасштабных вычислительных экспериментов, системное наполнение GURU представлено следующими основными блоками (рис. 3.):

LUNA – система автоматизации распараллеливания алгоритма, базирующаяся на разработке [28], которая включает инструментарию для кластерных конфигураций с распределённой и общей памятью;

AKB (Active Knowledge Base) – базы активных знаний с созданием онтологий и когнитивных технологий для предметных областей, пример концепции которых по задачам решения СЛАО представлен в [29]; в определённом смысле прототипом базы знаний является система ALGOWIKI [30];

VVT (Validation, Verification, Testing) – инструментарию для таких рутинных, но очень важных задач, как верификация, валидация, тестирование алгоритмов и программ;

GUI – интеллектуальные межмодульные и пользовательские интерфейсы с использованием наглядных графических средств;

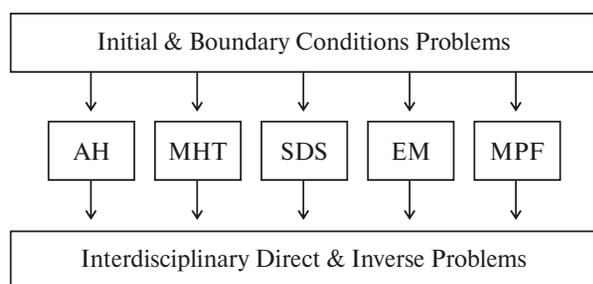


Рис. 2. Конфигурационное управление математическими задачами

Fig. 2. Configuration management of mathematical tasks

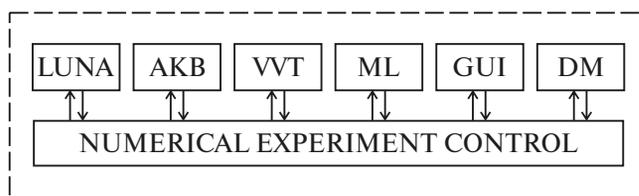


Рис. 3. Интеллектуальное системное наполнение интегрированного вычислительного окружения

Fig. 3. Intelligent system filling of the integrated computing environment

DM (Decision Making) – средства принятия решений по результатам вычислительных экспериментов [31];

ML (Machine Learning) – подсистема машинного обучения на основе формирования последовательности вычислительных сценариев с использованием методов оптимизации решения обратных задач. Здесь мы пока отметим существующие подходы, применяемые на основе средств искусственного интеллекта [32, 33].

Очевидно, что к этому списку следует добавить конструктор, который отвечает за поддержку конфигурации всего ИВО, в частности, обеспечивая многоверсионность и избыточность функциональных модулей,

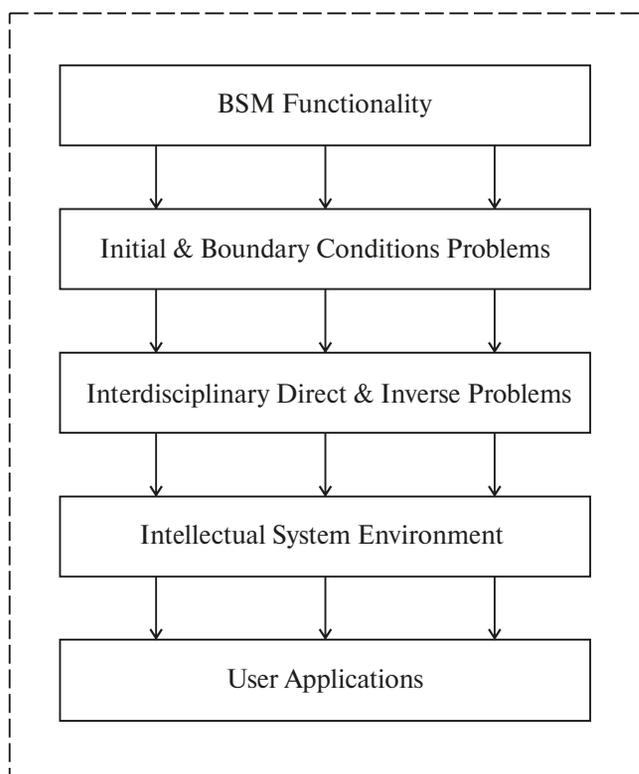


Рис. 4. Многоуровневая структура платформы GURU с иерархической общей памятью, реализующая распараллеливание алгоритмов

Fig. 4. Multi-level structure of the GURU platform with hierarchical shared memory, implementing parallelization of algorithms

позволяющих осуществлять контекстно-оптимальную сборку вычислительных схем.

Общая архитектура рассматриваемого интегрированного программного обеспечения представлена на рисунке 4 и содержит, кроме описанных выше основных блоков, систему пользовательских приложений, являющихся реализацией конкретных вычислительных схем для моделирования. Отметим, что они могут быть связаны как с фундаментальными исследованиями, так и с практическими прикладными задачами.

ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ И ПРОБЛЕМЫ РЕАЛИЗАЦИИ ИВО

Концепция системы GURU заключается в создании в определённом смысле идеального математического и программного обеспечения нового поколения, которое обладало бы высокими профессиональными качествами и экономичностью, с завершением его исходной рабочей версии за 2–3 года и с активной широкой эксплуатацией в течение более 10 лет. Разработка ориентируется главным образом на традиционные для последних десятилетий многопроцессорные вычислительные системы (МВС) кластерного типа средствами гибридного программирования с передачей сообщений между вычислительными узлами и использованием многопоточковой арифметики (средствами библиотеки MPI и командами OPENMP соответственно). Предполагается, что основные вычисления производятся со стандартной двойной точностью, а программирование алгоритмов осуществляется на широко распространённых языках C/C++ и Python, хотя многоязыковость не противоречит концепции проекта.

В силу принципа интегрируемости GURU открыто к естественным расширениям архитектуры МВС путём включения специализированных ускорителей, например с использованием CUDA, а также программируемых логических интегральных схем (ПЛИС). Что касается актуального автоматизированного распараллеливания программ, то здесь можно указать на перспективную разработку LUNA [28].

В общем случае к пользователям GURU следует отнести прежде всего математиков-программистов, которые разрабатывают функциональные и системные компоненты ИВО. Для них главная цель – кардинальное повышение производительности собственного труда и качества создаваемого продукта. Вторая категория пользователей – расчётчики, решающие конкретные прикладные или фундаментальные проблемы и заинтересованные в низкой ресурсоёмкости машинного эксперимента, а также в комфортных инструментах. Третья категория – тренеры и стажёры, или учителя и ученики, включая студентов и профессоров, участие которых необходимо учитывать в инновационном процессе в силу планируемого постоянного роста числа представителей профессии компьютерных «модделеров».

Методологически интегрированное вычислительное окружение следует рассматривать как наукоёмкое математическое и программное обеспечение (МПО), проблемно-независимое в широком смысле этого слова, то есть не ориентированное на применение в кон-



Рис. 5. Межотраслевая классификация инноваций интегрированного вычислительного окружения

Fig. 5. Inter-sectoral classification of innovations of the integrated computing environment

кретной отрасли. Другими словами, мы рассматриваем универсальный уровень, над которым должны создаваться различные надстройки для производственных отраслей, со своими инновационными и эксплуатационными службами и средствами. Примерная схема межотраслевой классификации МПО представлена на рисунке 5. Понятно, что при такой многоярусной архитектуре программное обеспечение для каждого прикладного направления экономично формируется средствами системного наполнения ИВО с использованием богатой функциональности БСМ, с обязательным созданием интеллектуальных специализированных пользовательских интерфейсов.

Требования к работе ИВО, с точки зрения его эффективности и успешности разработки, можно сформулировать следующим образом:

- гибкое расширение состава реализуемых моделей и применяемых алгоритмов, а также вычислительно-информационных технологий; их избыточность позволяет выбирать оптимальные методы из имеющихся в библиотеках, а также соответствующие математические постановки, обеспечивая самообучение GURU на основе анализа результатов накапливаемых предыдущих расчётов;
- интеллектуальные внутренние и внешние интерфейсы для разработчиков и пользователей; методики обучения и эксплуатации GURU;
- переиспользование внешних программных продуктов, представляющих огромный овеществлённый интеллектуальный потенциал; это позволяет значительно повышать возможности GURU, но одновременно и ужесточает условия совместимости интерфейсов;
- адаптация к эволюции компьютерных платформ; сюда входят, разумеется, требования многоязыковости и кроссплатформенности разработки, которые, в свою очередь, могут снизить её зависимость от изменений технических конфигураций;

• согласованное участие различных групп разработчиков; данное требование очевидно вследствие масштабности разработки: проект может быть реализован только сообществом профессионалов (математики, программисты и инженеры-технологи).

Для обеспечения работоспособности предлагаемого проекта он должен характеризоваться некоторыми особенностями, одни из которых общепризнаны, а другие специфичны и пока определены лишь в общих чертах. Прежде всего для сохранения всего программно-информационного контента необходим репозиторий, примером которого может служить широко распространённая система GITHUB, обеспечивающая управление крупными программными проектами и взаимодействие разностатусных пользователей. Известным её аналогом является ECLIPSE – свободная интегрированная среда разработки модульных кросс-платформенных приложений. Сюда также относится важная проблема качественной документации как для пользователей, так и для разработчиков программной продукции, чтобы можно было эффективно применять уже существующие интеллектуальные средства работы с текстами. Отметим также, что поскольку крупномасштабные вычислительные эксперименты зачастую требуют огромных объёмов операций и длительных расчётов, концепция интегрированного вычислительного окружения должна поддерживать контроль отказоустойчивости устройств многопроцессорных вычислительных систем.

Для кардинального повышения производительности труда разработчиков функционального и системного наполнения ИВО главный путь – автоматизация построения алгоритмов и их отображения на архитектуру ЭВМ. Решение этой задачи возможно одним из двух способов (или их комбинации). Первый – это создание специализированных библиотечных классов типа SPARSE BLAS для операций с разреженными матрицами, а второй – метоориентированные языки естественного типа. В качестве современного примера приведём проект MOLE (Mimetic Operation Library Expanded [34]) по реализации миметических аппроксимаций для типовых дифференциальных операторов типа градиента, ротора, дивергенции и Лапласа на различных сетках, в том числе неравномерных и криволинейных. Второй пример – разработка NEMESIS [35] для построения политопных схем дискретизации дифференциальных уравнений на полигональных сетках, по аналогии с прикладным программным комплексом FREE FEM [36], в котором используется специализированный язык написания математических формул. В идеале такие подходы следуют известному лозунгу «программирование без программирования». Существующие проблемы развития специализированных языков в работе [37] связываются с переходом от «палеоинформатики к неоинформатике» и с созданием «фабрики языков».

Несмотря на многочисленные попытки, всё ещё ждёт своего решения актуальная задача автоматического построения параллельных программ [38]. Определённые шаги в этом направлении предприняли авторы работы [28], где система LUNA интегрирует

средства гибридного программирования с распределённой и общей памятью. Другая перспектива заключается в использовании Q-детерминантов [39] для реализации оптимального ускорения параллельных вычислительных схем.

Серьёзные перспективы развития МПО связаны с переходом к уровню баз активных знаний, которые охватывают классифицированную информацию по литературным источникам, программным разработкам и по анализу потенциального рынка создаваемых продуктов. Огромные имеющиеся объёмы памяти и возможности быстрой интеллектуальной обработки данных, в том числе с осуществлением логических выводов, обеспечивают перспективу достижения качественно нового уровня информатизации человеческой деятельности, по сравнению с тем, который имелся, например, в ранних версиях википедии или в существовавшем когда-то отечественном фонде алгоритмов и программ (ГОСФАП).

Главная цель интеллектуализации суперкомпьютерного моделирования – машинное обучение, которое должно базироваться на законах математической логики и технологических средствах принятия решений. Основные проблемы, возникающие при создании наукоёмкого программного обеспечения, рассматривались в работе [40]. Самый простой принцип обучения заключается в методе проб и ошибок. Однако глубокое изучение сложнейших многопараметрических процессов и явлений (с выявлением их внутренних свойств и построением прогнозов) возможно главным образом с применением классических алгоритмов оптимизации для решения обратных задач. Строго говоря, рассматривается не сам объект, а его модель, в силу чего такой подход называется «суррогатным моделированием». При этом определяется некоторый целевой функционал, который необходимо минимизировать, как правило, при заданных ограничениях на параметры задачи.

При исследовании сложных систем формулируется уравнение состояния, которое может описываться большим набором дифференциальных и/или интегральных соотношений. В многомерных случаях одно вычисление целевого функционала означает решение ресурсоёмкой прямой задачи, а в процессе направленных поисков оптимальных параметров таких расчётов может быть очень много (сотни, тысячи и более). Теория нелинейной оптимизации (условной или безусловной) насчитывает большое количество современных алгоритмов [23]. Например, в прогнозах погоды актуальная задача усвоения измеряемых данных эффективно решается с помощью сопряжённых уравнений [41, 42]. Данный класс задач экстремально сложен; они могут быть решены в интерактивном режиме путём организации вычислительных сеансов с коррекциями моделей и направлений поиска в пространстве параметров, где большое значение имеет опыт и искусство специалиста-«моделлера».

Тематика машинного обучения включает многочисленные фундаментальные и прикладные аспекты создания и внедрения наукоёмкого интегрированного вычислительного окружения, многие из которых ещё

ждут глубоких исследований [43–45]. Кроме того, среди разнообразных вопросов создания больших программных проектов следует упомянуть такие актуальные, как организация вычислений в «облачных сетях» [46] и менеджмент «софтверных» продуктов [47].

Проблематика ИВО, конечно, тесно связана с разработками систем автоматизации проектирования (САПР, англоязычные аналоги CAD, CAE, а также PLM [48]), которые составляют сейчас мировой многомиллиардный рынок и претерпевают конвергенцию с продуктами математического моделирования. Примерами успешных инноваций здесь являются компании CADFEM (дистрибьютор широко распространённых в России и Европе продуктов ANSYS) и многие другие разработчики прикладных программных комплексов на основе FEA (конечно-элементный анализ). Для данного сегмента приложений исходные данные и результаты расчётов следует представлять в виде заводской инженерно-конструкторской документации, что позволяет включать этапы моделирования непосредственно в производственный цикл.

Другое характерное направление интегрированного вычислительного окружения – системы моделирования, связанные с обработкой огромных объёмов измеряемых данных, которые могут быть неточными, неполными и противоречивыми. Примеры – прогноз погоды и климата, различные спутниковые наблюдения и т.д.

Что касается создания и применения учебных модификаций системы GURU, её потенциальные возможности в образовательном процессе трудно переоценить. Причём это касается не только таких дисциплин, как вычислительная и прикладная математика, методология моделирования и информационные технологии. Речь может идти о коренном пересмотре преподавания для самых разных научных и инженерных направлений на основе изучения процессов и явлений с наглядной иллюстрацией существующих зависимостей результатами численных экспериментов. Здесь возникает много новых дидактических вопросов, поскольку наличие огромных возможностей суперкомпьютерной педагогики открывает дорогу к формированию идеального учителя с сочетанием искусственного и естественного интеллекта. Реализация машинного обучения в образовании очевидным образом осуществляется путём формирования надстроек для штатного контента ИВО.

Нельзя не сказать о принципиальной особенности архитектуры GURU: на смену исторически сложившемуся «зоопарку» из проблемно ориентированных разработок, неизбежно развиваемых методом «заплаток», предлагается создать с чистого листа, но на основе критического анализа положительного и отрицательного опыта наукоёмких разработок, интегрированного вычислительного окружения нового поколения, которое было бы ориентировано на перспективные современные методы и технологии, с учётом огромных достижений суперкомпьютерных систем и искусственного интеллекта за последнее десятилетие.

Другая важная черта предлагаемой концепции – системный подход, то есть комплексность, полнота

и целостность планируемой разработки, включая все аспекты создания, эксплуатации, развития и обучения. В этом смысле ИВО представляет собой сложную систему, которая сама является достойным объектом для дальнейших исследований. Его предметная область включает такие понятия, как формализованные представления моделей, алгоритмов, технологических инструментов и т.д. Рассмотренные методо- и проблемно ориентированные разноуровневые классификации процессов математического моделирования могут служить отправной точкой в таком аналитическом подходе.

МЕТОДОЛОГИЧЕСКИЕ ВОПРОСЫ СОЗДАНИЯ ИНТЕГРИРОВАННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И МАШИННОГО ОБУЧЕНИЯ

Особенность текущего «постпетафлопсного» периода развития вычислительно-информационных наук и технологий связана с тем, что они выпустили из бутылки «интеллектуального джина», который потенциально способен в разы повысить темпы роста производительности труда и внутреннего валового продукта (ВВП). А значит, необходимо создать математическое и программное обеспечение нового поколения для решения сверхбольших задач суперкомпьютерного моделирования и массово внедрить его в производственную, социальную, образовательную и гуманитарную сферы. Образно говоря, это означает переход от парадигмы разовых узконаправленных разработок в стиле «шире круг броуновского движения», нацеленных на краткосрочное использование, к созданию промышленных производств высококонкурентных отечественных программных изделий с длительным жизненным циклом. Они должны базироваться на оптимальном планировании и анализе рынков с учётом приоритетных направлений научно-технологического развития страны. Такая экосистема предполагает стимулирование создания инициативных продуктов благодаря эффективной инструментальной среде вроде интеллектуального конструктора LEGO.

Необходимо подчеркнуть, что предлагаемая архитектура GURU не является абстрактной конструкцией структурированного набора кодов; она представляет собой содержательный комплекс функциональных модулей, инструментов и научных данных, формируемых на основе системного анализа огромного практического опыта реализации методов и технологий для различных стадий математического моделирования. В зависимости от целей и ресурсов команды пользователей предлагаемая экосистема выдвигает или эксклюзивные решения, или базовые конфигурации с гибкими возможностями расширения и развития. Это касается как разработок новых продуктов, так и готовых расчётных средств, а также машинных обучающих курсов или мастер-классов.

Важно отметить, что тенденция стремительного одновременного роста вычислительно-информационных ресурсов и интеллектуальной мощи суперкомпьютеров согласованным образом порождает общую математическую проблему – переход к высокоточным алгоритмам с активным использованием функцио-

нально-аналитических преобразований. Дело в том, что машинная производительность определяется арифметическими и информационными операциями, а пересылка данных не только на порядки замедляет вычисления, но и вносит основной вклад в энергозатраты эксплуатации ЭВМ, а это сейчас главный экономический фактор. Поэтому перед разработчиками стала необходимость новых разработок с кардинальным уменьшением используемой памяти, что означает акцент на методы высокого разрешения, связанные со сложными формульными преобразованиями. Десятки лет назад это было препятствием в силу человеческого фактора, но сейчас с этими трудностями легко справляется искусственный интеллект, располагающий системами автоматизации аналитических преобразований.

Рассмотренные концептуальные и технологические инновации наукоёмкого МПО призваны преодолеть

в определённом смысле кризис программирования и выровнять темпы развития «софтвера» и «хардвера». Простые соображения подсказывают, что динамика этого процесса нелинейная с положительным ускорением, и здесь получают преимущества те государства, которые раньше вступают на путь опережающего развития. При этом в условиях стихийного рынка менее развитые страны навсегда отстают от мирового прогресса. Чтобы избежать пессимистического сценария, необходимо регулирующее воздействие государства – на уровне правительственных решений. Есть потребность в креативной межведомственной инфраструктуре с участием ведущих академических коллективов, госкорпораций и коммерческих компаний, творческий потенциал которых должен обеспечить в короткие сроки создание МПО-отрасли качественно нового научно-технологического уровня и её устойчивое развитие.

СПИСОК ЛИТЕРАТУРЫ/REFERENCES

1. LeCun Y., Bengio Y., Hinton G. Deep learning // *Nature*. 2015, vol. 521, pp. 436–444.
2. Ershov A.P., Marchuk G.I. Man-machine Interaction ш Solving a Certain Class of Differential Equations // *Proceed. IFIP Congress*. New York, 1965. Pp. 550–556.
3. Marchuk G.I. On the automatically construction of algorithms // *Applicate Math*. 1965, vol. 10, no. 3, pp. 255–267.
4. Marchuk G.I., Il'in V.P. Parallel computations in grid methods for solving mathematical physics problems // *In: Proceedings of the IFIP Congress 1980*. Tokyo–Amsterdam, 1980. Pp. 671–676.
5. ЛОГОС: Пакет программ инженерного анализа и суперкомпьютерного моделирования. <http://www.loges.vniief.ru>
LOGOS: A software package for engineering analysis and supercomputer modeling. <http://www.loges.vniief.ru> (In Russ.)
6. Flow Vision. <http://www.flowvision.ru>
7. ANSYS. <http://www.ansys.com>
8. Comsol Multiphysics. <http://www.comsol.ru>
9. Hammond G.E.H., Lichtner P.C., Mills R.T. Evaluating the performance of parallel subsurface simulators. An illustrate example with PFLOWTRAN // *Water Resources Research*. 2014, vol. 50, pp. 208–228. DOI: 10.1002/2012WRD13483
10. FENICs. <http://www.fenicsproject.org>
11. *Бартенев Ю.Г., Ерзунов В.А., Карпов А.П. и др.* Комплекс библиотек параллельных решателей SLAE LPARSOL версии 3 // XV Международная конференция «Супервычисления и математическое моделирование» / Под ред. Р.М. Шагалиева. Саров: РФЯЦ-ВНИИЭФ, 2014. С. 102–110.
Bartenev Yu.G., Erzunov V.A., Karpov A.P. et al. Complex of libraries of parallel solvers SLAE LPARSOL version 3 // XV International Conference “Supercomputing and Mathematical Modeling” / Ed. R.M. Shagaliev. Sarov: RFNC-VNIIEF, 2014. Pp. 102–110. (In Russ.)
12. PETSc: Portable Extensible Toolkit for Scientific Computation. <https://petsc.org/release/>
13. HYPRE: Scalable Linear Solvers and Multigrid Methods. <https://computing.llnl.gov/projects/hyre-scalable-linear-solvers-multigrid-methods/>
14. NETLIB. <http://www.netlib.org>
15. *Dongarra J.* List of freely available software for linear algebra on the web (2006). <http://netlib.org/utk/people/JackDongarra/la-sw.ht>
16. Bastian P., Blatt M., Dedner A., et al. The Dune Framework: Basic Concepts and Recent Developments // *Computers and Mathematics with Applications*. 2020. DOI.org/10.1016/j.camwa.2020.06.007
17. OpenFOAM. <https://www.openfoam.com/>
18. INMOST: A Toolkit for Distributed Mathematical Modeling. <https://www.inmost.org>
19. Ilyin V.P. The Conception, Requirements and Structure of the Integrated Computational Environment. 4 th Russian Supercomputing Days 2018. Revised Selected Papers. 653665 (Communications in Computer and Information Science). Vol. 965. Springer-Verlag GmbH and Co., 2028.

20. Ильин В.П. Программирование ближайшего будущего: концепция и прагматика // Вестник Российской академии наук. 2023. Т. 93. № 2. С. 150–161. DOI: 10.31857/S086958732302007X
Ilyin V.P. Programming the Near Future: Concept and Pragmatics // *Herald of the Russian Academy of Sciences*. 2023, vol. 93, no. 2, pp. 150–161. DOI: 10.31857/S086958732302007X (In Russ.)
21. Ilyin V.P. Mathematical Modeling. Part 1. Continuous and Discrete Models. Novosibirsk, SBRAS Publ., 2017.
22. Ilyin V.P. On an Integrated Computational Environment for Numerical Algebra // *Springer Nature Switzerland AG 2019* / L. Sokolinsky, M. Zymbler (eds.). PCT 2019. V. CCIS 1063. Pp. 91–106.
23. Ilyin V.P. The Integrated Computational Environment for Optimization of Complex Systems // *Proceed. of the 15th International Asian School-Seminar*, 2019.
24. Ilyin V.P. Integrated Computational Environment for Grid Generation Parallel Technologies // *Springer Nature Switzerland AG 2020* / L. Sokolinsky, M. Zymbler (eds.). 2020. V. CCIS 1263. P. 5868.
25. Бутюгин Д.С., Ильин В.П. СНЕВЫСНЕВ: принципы автоматизации построения алгоритмов в интегрированной среде для сеточных аппроксимаций начально-краевых задач. Труды Международной конференции ПАВТ'2014. Челябинск: Изд-во ЮУрГУ, 2014. С. 4250.
Butyugin D.S., Ilyin V.P. СНЕВЫСНЕВ: Principles of automation of algorithm construction in an integrated environment for grid approximations of initial-boundary value problems. *Proceedings of the International Conference PAVT'2014*. Chelyabinsk: SUSU Publishing House, 2014. P. 4250. (In Russ.)
26. Ilyin V.P. Parallel intelligent computing in algebraic problems, in *Parallel Computational Technologies*. PCT 2021. *Communications in Computer and Information Science* / Ed. by L. Sokolinsky and M. Zymbler. Vol. 1437, pp. 108–117. Springer, Cham., 2021.
27. Ilyin V.P. Artificial Intelligence Problems in Mathematical Modeling // *Springer Nature Switzerland AG 2019* / V. Voevodin, S. Sobolev (eds.). RuSCDays 2019. CCIS 1129. 505516.
28. Akhmed-Zaki D., Lebedev D., Malyshkin V., Perepelkin V. Automated Construction of High-Performance Distributed Programs in LuNA System // *Parallel Computing Technologies*. PaCT 2019. *Lecture Notes in Computer Science*. V. 11657. Springer Cham. 3–9. <https://doi.org/10.1007/978-3-030-25636-4i>
29. Ilyin V.P. The conception and architecture of the knowledge base of the systems of linear algebraic equations. *Proceed. of IX Conference "Knowledge–Ontology–Theory"*. Novosibirsk: ISI SBRAS, 2023. Pp. 143–154.
30. Antonov A., Dongarra J., Voevodin V. AlgoWiki Project as an Extension of the Top500 Methodology // *Supercomput. Frontiers and Innovations*. 2018, vol. 5, no. 1, pp. 4–10.
31. Zagorulko Yu., Zagorulko G. Architecture of extensible tools for development of intelligent decision support systems // *New Trends in Software Methodologies, Tools and Techniques*. Proc. of the 10th SoMeT-11. Hamido Fujita (ed.). Amsterdam: IOS Press, 2011. Pp. 457–466.
32. Goswami S., Bora A., Yu Y., Karniadakis G.E. Physics-informed deep neural operator networks // *Machine Learning in Modeling and Simulation: Methods and Applications*. Springer, 2023. Pp. 219–254.
33. Fanaskov V.S., Oseledets I.V. Spectral neural operators // *Doklady Mathematics*. 2023, vol. 108, pp. 226–232.
34. MOLE. <http://mole-docs.readthedasio/en/latest>
35. NEMESIS. <http://github.com/jdranjou/Hard>
36. FREEFEM. <http://freefem.org>
37. Kleppe A. *Software Language Engineering: Creating Domain Specific Language Using Metamodels*. N.Y.: Addison Wesley, 2008.
38. DVM Systems. <http://www.keldush.ru/dvm>
39. Aleeva V. Designing Parallel Programs on the Base of the Conception of τ -Determinant // *Supercomputing. RuSCDays 2018. Communications in Computer and Information Science (CCIS)*. 2019, vol. 965 / V. Voevodin, S. Sobolev (eds.). Springer, Cham. 565577.
40. Weinan E. Machine Learning and Computational Mathematics // *Commun. Comput. Phys.* 2020, vol. 28, no. 5, pp. 1639–1670. <https://doi.org/10.4208/cicp.oa-2020-0185>
41. Марчук Г.И. Сопряжённые уравнения и анализ сложных систем. М.: Наука, 1992.
Marchuk G.I. Adjoint equations and analysis of complex systems. Moscow: Nauka, 1992. (In Russ.)
42. Агошков В.И., Залесный В.Б., Шутяев В.П. Сопряжённые уравнения и методы вариационного усвоения данных наблюдений в задачах геофизической гидродинамики // *Актуальные проблемы вычислительной математики и математического моделирования* / Под ред. В.П. Дымникова. Новосибирск: Изд-во СО РАН, 2025. С. 9–36.
Agoshkov V.I., Zalesny V.B., Shutyaev V.P. Adjoint equations and methods of variational assimilation of observational data in problems of geophysical hydrodynamics // *Actual problems of computational mathematics and*