

Е.А. Корчевская, Л.В. Маркова

*Витебский государственный университет им. П.М. Машерова,
г. Витебск, Беларусь*

РАЗРАБОТКА МОДЕЛЕЙ И АЛГОРИТМОВ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ НА ОСНОВЕ ОБЪЕКТНО-ОРИЕНТИРОВАННОЙ ПАРАДИГМЫ ПРОГРАММИРОВАНИЯ

В настоящее время в образовательном процессе используется структурный подход при создании алгоритмов вычислительной математики. Численные методы представляют собой классическую область для применения вычислительной техники. До появления первых вычислительных средств теория численных методов обгоняла вычислительные возможности, однако со стремительной эволюцией компьютеров и быстрым развитием технологий программирования ситуация менялась. Начиная с некоторых пор, уже вычислительные возможности обгоняют теоретические успехи. Во многих учебниках вопросам программирования численных методов вообще не уделяется места. Понимая важность изложения теории численных методов, нельзя не признать, что вопросы реализации численных методов являются отнюдь не техническими. Исследование авторов ставит своей целью показать, как с помощью современного языка программирования и объектно-ориентированного подхода можно реализовать подавляющее большинство численных методов алгебры, анализа, математической физики, которые могут быть использованы при моделировании естественнонаучных задач.

Прямые методы линейной алгебры – это методы решения систем линейных алгебраических уравнений и методы решения проблемы собственных значений. Известно, что прямые методы теоретически базируются на тех или иных элементарных матричных преобразованиях, которые, в конечном счете, приводят задачу к эквивалентной, но более простой форме, допускающей ее непосредственное решение [1]. Набор таких преобразований сравнительно невелик. Исходя из технологии объектно-ориентированного программирования

эти преобразования могут быть представлены матричными операциями с объектами особого рода – абстрактными матрицами [1]. Важной методологической составляющей реализации технологии объектно-ориентированного программирования в курсе изучения вычислительных методов является определение необходимого набора таких матриц и представление их классами единой матричной иерархии, наряду с основными функциональными матрицами. Далее все прямые методы линейной алгебры рассматриваются как композиции матричных преобразований. При этом их программная реализация основывается на организации обобщенного суперкласса вычислительных методов, применимого ко всем конкретным типам задач линейной алгебры. Значительная часть операций непосредственно реализуется в верхних матричных классах. Тем не менее, переопределение их в конкретных классах с учетом частных математических свойств, особенностей конкретных структур данных позволяет при необходимости добиться наибольшей эффективности программного кода.

Для эффективного применения объектно-ориентированного программирования необходимо также сами методы линейной алгебры рассматривать как реализацию объектов соответствующих классов в их целостной объектной классификации. При этом классификация объектно-ориентированного подхода должна следовать классификации задач линейной алгебры, т.к. результатом решения различных постановок являются объекты разных типов, и это должно отражаться в спецификациях алгоритмических классов. Кроме того, в классификации желательно отразить деление алгоритмов на прямые и итерационные, поскольку принципы их организации существенно отличаются, а это неизбежно приводит к различиям в их программной реализации.

Также предложен объектно-ориентированный подход для моделирования основных методов решения задач численного анализа и математической физики. Для обеспечения возможности проведения расчётов с минимумом затрат времени и наиболее оптимальным использованием вычислительных ресурсов использован способ организации компьютерных вычислений, представляющий собой набор взаимодействующих вычислительных процессов, работающих параллельно.

Выбор технологий является важным предварительным этапом разработки сложных информационных систем. Платформа и язык программирования, на котором будет реализована система, заслу-

живает большого внимания, так как исследования показали, что выбор языка программирования влияет на производительность труда программистов и качество создаваемого ими кода. Среди факторов, влияющих на выбор технологий отметим следующие:

- используемые технологии должны обеспечивать комфортную работу с объектами параллельного программирования;

- выбранный язык программирования должен быть достаточно производительным для соответствия вычислительным целям проекта;

- используемый язык программирования должен быть высокоуровневым и в достаточной степени документированным для обеспечения простоты дальнейшей поддержки проекта.

С учётом вышперечисленных факторов предпочтение отдано языку программирования Rust [2] из-за более передовых подходов к параллельному программированию и контролю за использованием памяти. Также Rust выгодно отличается ориентированностью на написание сетевых клиент-серверных приложений (что в дальнейшем обеспечит возможность расширения разрабатываемой системы) и высокой производительностью, не уступающей языкам семейства C.

Для проектов на языке Rust используется утилита под названием Cargo, и именно на использование этого инструмента была ориентирована разработка программной библиотеки. В его возможности входит автоматизация управления сборкой, запуском и тестированием проектов и управление зависимостями и метаинформацией проекта с использованием специального метафайла.

Создана библиотека, которая содержит три модуля:

- `structs`, отвечающий за реализацию структур хранения и оперирования данными;

- `methods`, представляющий собой базовый набор как параллельных, так и обычных численных методов;

- `benches`, содержащий набор тестов производительности.

Литература

1. Маркова Л.В., Корчевская Е.А., Красоткина А.Н. Вычислительные методы алгебры: практикум. Витебск: ВГУ им. П.М. Машерова, 2013.
2. The Rust Programming Language // Rust Documentation. 2015. URL: <https://doc.rust-lang.org/book>.