

# Первая версия Открытой энциклопедии свойств алгоритмов

А.С. Антонов<sup>1</sup>, Вад.В. Воеводин<sup>1</sup>, Вл.В. Воеводин<sup>1</sup>, А.М. Теплов<sup>1</sup>, А.В. Фролов<sup>2</sup>

Московский государственный университет имени М.В. Ломоносова<sup>1</sup>,  
Институт вычислительной математики РАН<sup>2</sup>

Открытая энциклопедия свойств алгоритмов AlgoWiki представляет собой проект создания в сети Интернет wiki-подобной публичной энциклопедии, максимально подробно описывающей свойства и характеристики наиболее используемых математических алгоритмов. Предложенная структура описания свойств алгоритма состоит из двух частей: машинно-независимых свойств алгоритма и особенностями различных реализаций данного алгоритма. Первая версия Открытой энциклопедии свойств алгоритмов AlgoWiki реализована в сети Интернет по адресу <http://algowiki-project.org>. Потенциал разработанной технологии продемонстрирован на описаниях более чем 10 реальных алгоритмов.

## 1. Исследование свойств алгоритмов

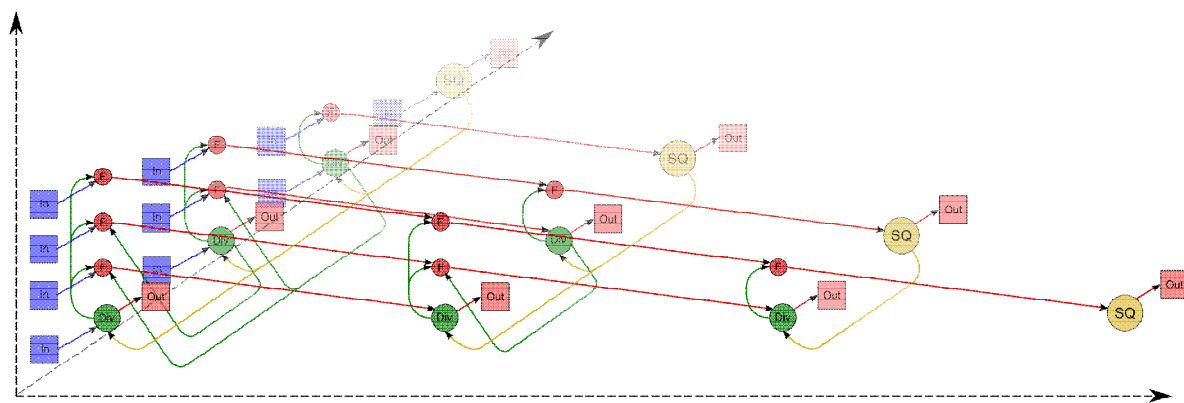
### 1.1 Введение в проблематику

Задача исследования свойств алгоритмов остается весьма актуальной в особенности из-за повсеместного проникновения параллелизма в архитектуру современных вычислительных систем. Для эффективной реализации алгоритма на параллельных вычислительных системах помимо традиционных свойств, требуется исследовать и свойства, связанные с использованием ресурса параллелизма, такие как оценка параллельной сложности алгоритма, сбалансированность параллельных шагов по числу и типу операций и т.д. [1, 2]

Для определения тонких информационных свойств алгоритмов разработан подход, связанный с исследованием графа информационных зависимостей или графа алгоритма [1, 3-7] – единственной графовой модели программ и алгоритмов, содержащей полную информацию об их структуре. Этот подход позволяет определить все необходимые свойства исследуемого алгоритма, как традиционные, так и связанные с использованием параллелизма. Ориентация на использование графа алгоритма гарантирует полноту определения ресурса параллелизма.

Потенциальный параллелизм особенно важен, так как в этом случае появляется возможность описания машинно-независимых свойств программ и их последующее эффективное отображение на различные архитектуры, а также их запись с помощью различных технологий параллельного программирования.

Одно из направлений работ в рамках данного проекта связано с отработкой и стандартизацией методов визуального представления графа алгоритма. Часто граф алгоритма является сложным многомерным объектом, а многие его свойства становятся понятными только при его удачном визуальном отображении. Один из примеров текущего варианта визуализации графа алгоритма приведен на рис. 1.



**Рис. 1.** Граф алгоритма разложения Холецкого с отображением входных и выходных данных

Использование разных типов вычислительных систем (компьютеров с общей и распределенной памятью, разнородных систем, систем, использующих различные типы ускорителей и т.д.) приводит к тому, что применительно к каждому типу компьютеров свойства алгоритмов должны использоваться по-разному. Возникает также ряд свойств, связанных с качеством именно различных параллельных реализаций рассматриваемого алгоритма. Это, например, свойства, описывающие степень локальности данных и вычислений в программе [8-11], или свойства, связанные с исследованием масштабируемости параллельной программы [12-15].

Исследования по определению информационной структуры программ опираются на методы статического анализа программ, т.е. на анализ лишь исходных текстов без какого-либо учета данных времени исполнения. Вместе с этим, у авторского коллектива есть значительный задел и в области исследования динамических характеристик программно-аппаратных сред [16-18], что прекрасно дополняет сведения, полученные в статике. Разработана система мониторинга данных системного уровня, позволяющая для каждого приложения определять как типичные значения характеристик поведения, так и обнаруживать аномалии, сигнализирующие о деградации реальной производительности суперкомпьютерных систем.

В рамках данного проекта разработанная технология будет использована для описания свойств целого множества реальных алгоритмов и их реализаций для классов архитектур – это также является одной из задач проекта. На начальном этапе основной упор делается на алгоритмы линейной алгебры: различные варианты нахождения сумм элементов векторов и скалярного произведения, варианты перемножения матриц, умножения матрицы на вектор, различные алгоритмы решения систем линейных уравнений, алгоритмы решения задач на собственные вектора и значения, варианты задач как с плотными, так и с разреженными структурами. В дальнейшем множество алгоритмов будет расширено.

Предполагается, что выполненное описание свойств реальных алгоритмов ляжет в основу Открытой энциклопедии свойств алгоритмов в сети Интернет. Энциклопедия посвящена свойствам алгоритмов и должна предоставлять возможность коллективной работы всего вычислительного сообщества над ее пополнением и совершенствованием. Несмотря на ориентацию данного проекта в сторону мощных суперкомпьютерных систем, сформулированная проблема актуальна для всех компьютерных платформ, вплоть до мобильных телефонов и планшетов, которые также стали параллельными. Сегодня почти весь компьютерный мир стал параллельным, и другим он уже не станет никогда.

## 1.2 Структура описания свойств алгоритмов

Разработанное описание свойств алгоритмов состоит из двух частей [19]. В первой части описываются собственно алгоритмы и их свойства, а вторая посвящена описанию особенностей их программной реализации с учетом конкретных программно-аппаратных платформ. Такое деление на части сделано для того, чтобы машинно-независимые свойства алгоритмов, которые определяют качество их реализации на параллельных вычислительных системах, были бы вы-

делены и описаны отдельно от множества вопросов, связанных с последующими этапами программирования алгоритмов и исполнения результирующих программ.

## **1 ЧАСТЬ. Описание свойств и структуры алгоритмов: общая часть**

### *1. Общее описание алгоритма*

Общее словесное описание алгоритма является необходимым элементом схемы описания свойств алгоритмов, поскольку дает пользователю информацию как о том, к какому классу схожих по структуре алгоритмов данный алгоритм принадлежит, так и об особенностях алгоритма и объектов, с которыми он работает. По данной информации, а также по описанию математических свойств и структуры входных данных пользователь может однозначно понять суть решаемой задачи и определить, нужно ли ему дальнейшее знакомство с данным алгоритмом.

### *2. Математическое описание алгоритма*

Математическое описание решаемой задачи в виде совокупности формул и соотношений необходимо для строгой и однозначной постановки решаемой задачи для человека, знающего математику.

### *3. Выделение вычислительного ядра алгоритма*

Выделение и описание вычислительного ядра (вычислительных ядер) алгоритма необходимо для понимания того, на какую часть приходится основное время работы алгоритма. Скорее всего, именно эти части будут представлять наибольший интерес для пользователя при дальнейших исследованиях.

### *4. Определение макроструктуры алгоритма*

Многие алгоритмы удобно описывать не целиком, а в виде макроструктуры, ссылаясь на другие описанные алгоритмы в качестве составных частей.

### *5. Описание схемы реализации последовательного алгоритма*

Прежде чем переходить к описанию параллельных вариантов алгоритма, полезно бывает понять, как устроен исходный последовательный вариант. Возможно, некоторым пользователям будет достаточно и его, либо же алгоритм может не иметь эффективной параллельной реализации.

### *6. Определение последовательной сложности алгоритма*

Оценка последовательной сложности алгоритма в виде числа последовательных операций, которые нужно выполнить для получения результата работы алгоритма, позволяет понять, насколько сложным является последовательный вариант алгоритма. Недостаточное количество операций может говорить о том, что задача распараллеливания данного алгоритма не имеет практического смысла.

### *7. Определение информационного графа алгоритма*

Информационный граф (граф алгоритма) является удобным представлением алгоритма при исследовании его структуры, ресурса параллелизма, а также других свойств. Граф алгоритма обладает компактностью за счет параметризации. Визуальное представление графа алгоритма, а также его описание позволяют оценить многие полезные свойства исходного алгоритма.

### *8. Описание ресурса параллелизма алгоритма*

Оценка параллельной сложности алгоритма как числа шагов, за которое можно выполнить данный алгоритм, в предположении доступности неограниченного числа необходимых исполнителей (процессоров, функциональных устройств, вычислительных узлов, нитей и т.п.) дает информацию о минимальном времени, за которое в принципе может быть выполнен данный алгоритм. Описание ресурса параллелизма в виде суперпозиции конечного и массового параллелизма позволяет оценить, каким потенциалом для распараллеливания данный алгоритм обладает.

### *9. Описание входных и выходных данных алгоритма*

Описание объема, структуры, особенностей и свойств входных и выходных данных алгоритма позволяет судить о необходимости использования и необходимых характеристиках дисковых или иных хранилищ данных, а также о характеристиках каналов связи с ними при реализациях данного алгоритма. Предположения относительно диапазона значений входных данных могут повлиять на выбор той или иной реализации алгоритма.

### *10. Описание свойств алгоритма, важных на этапе реализации*

Свойства алгоритма, независимые от программно-аппаратной среды, позволяют судить обо многих особенностях алгоритма, которые необходимо учитывать при реализации на любом целевом суперкомпьютере.

## **2 ЧАСТЬ. Описание свойств и структуры алгоритмов: программная реализация**

### *1. Особенности реализации последовательного алгоритма*

При переходе к реализациям полезно сначала оценить особенности и варианты реализации алгоритма в виде последовательной программы, которые влияют на эффективность ее выполнения. Если последовательный вариант по какой-то причине является неэффективным на данной вычислительной платформе, то с большой вероятностью параллельная реализация также будет неэффективна.

### *2. Описание локальности данных и вычислений*

Локальность работы с данными в программе является одним из базовых факторов, определяющих эффективность ее работы. Приводимые результаты оценки локальности работы с данными для конкретных реализаций должны помочь оценить влияние данного фактора и способствовать выбору оптимального варианта.

### *3. Возможные способы и особенности реализации параллельного алгоритма*

Способов реализации обнаруженного в алгоритме ресурса параллелизма может быть много. Информация из данного раздела должна помочь выбрать тот вариант реализации, который будет наиболее эффективным.

### *4. Масштабируемость алгоритма и его реализации*

Масштабируемость является ключевой характеристикой при использовании высокопроизводительных вычислительных систем. Всесторонний анализ этого фактора должен помочь более эффективному использованию имеющихся вычислительных ресурсов.

### *5. Динамические характеристики и эффективность реализации алгоритма*

Анализ различных динамических характеристик различных реализаций алгоритма должен помочь определению реальных причин снижения производительности параллельного приложения, определению узких мест алгоритма, особенностей реализации и отображения на конкретную вычислительную платформу.

### *6. Выводы для классов архитектур*

Выделение позитивных и негативных факторов по отношению к конкретным классам архитектур и определение вариантов оптимизации программы для различных суперкомпьютеров дает информацию, необходимую как для выбора целевой вычислительной платформы, так и для оптимизации программы под уже имеющийся суперкомпьютер.

### *7. Существующие реализации алгоритма*

Для многих алгоритмов накоплен большой багаж реализаций для самых разных целевых вычислительных платформ. Информация из данного раздела должна помочь ориентироваться в этом множестве реализаций, чтобы выбрать из них наиболее подходящую для конкретного пользователя.

## **2. Первая версия Открытой энциклопедии**

### **2.1 Технологическая основа**

Первая версия Открытой энциклопедии свойств алгоритмов AlgoWiki [19, 20] реализована в сети Интернет по адресу <http://algowiki-project.org>. В качестве основы реализации был выбран программный механизм MediaWiki [21], реализующий систему управления контентом в стиле так называемой «вики»-технологии. Вики-технология предоставляет возможность создать сайт в сети Интернет, структуру и содержимое которого могут самостоятельно изменять пользователи с помощью предоставленных инструментов. Для форматирования текста и вставки различных объектов предлагается специальная разметка. На базе этой технологии построены Википедия [22] и многие другие сайты.

MediaWiki является свободно распространяемым программным средством с открытым кодом и написано на языке PHP. Для хранения данных используется база данных MySQL.

MediaWiki предлагает свой формат разметки текста, называемый викитекст, который поддерживает гиперссылки для создания ссылок между страницами и является более простым, наглядным и безопасным, чем язык HTML. Его использование позволяет создавать и редактировать страницы пользователям, не знакомым с такими технологиями, как XHTML и CSS.

Все варианты редактируемой пользователем страницы записываются в базу данных MySQL. Это дает возможность администратору легко аннулировать произведенные изменения в случаях как непреднамеренной порчи содержимого, так и вандализма или спама.

Для некоторых специальных целей (отображения формул, генерации pdf, специального форматирования и т.д.) используются специальные расширения, написанные на PHP независимыми разработчиками. Таких расширений разработано большое количество, их использование позволяет администраторам и пользователям получить возможность адаптировать сайт под собственные запросы.

Для представления на сайте математических формул используется расширение MathJax версии 0.7. Оно позволяет показывать формулы в форматах TeX и MathML путем вызова JavaScript-инструментария MathJax.

Для поддержки многоязычности (в текущей реализации – русского и английского языков) установлено расширение Interwiki версии 2.0. Interwiki – это средство организации ссылок между различными вики-сайтами в сети Интернет. Оно предоставляет возможность избежать вставки в текст полного адреса страницы, ограничившись внутренней ссылкой. При этом ссылка формируется из двух частей: к префиксу удаленного вики-сайта добавляется название удаленной страницы.

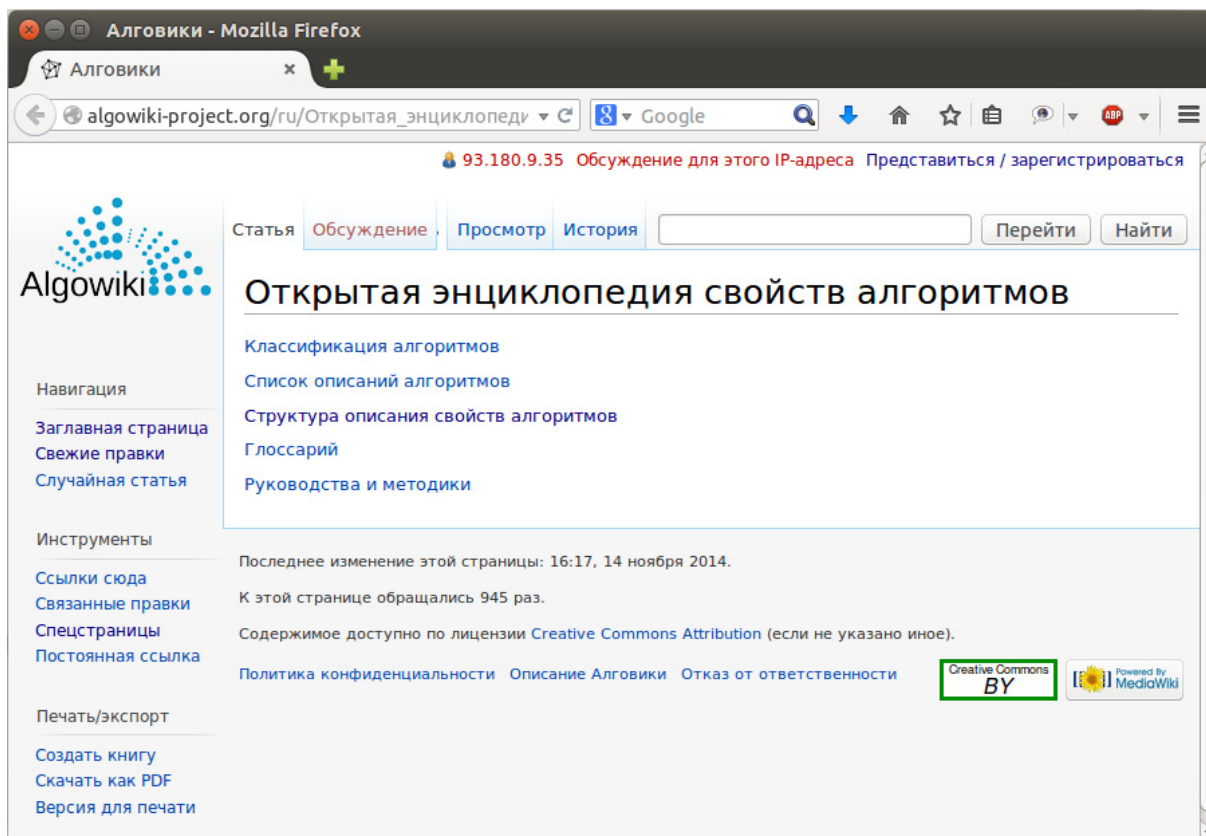
Для предоставления возможности конвертации страниц сайта в формат pdf установлено расширение Collection версии 1.5. К сожалению, получаемый файл в формате pdf иногда выглядит не идеально. Из-за того, что формулы генерируются в виде картинок и затем вставляются в итоговый файл вместе с текстом, на месте стыка текста и формул наблюдаются артефакты: не соблюдаются интервалы, математические символы немного «плавают» и т.д. На данный момент исследуется возможность использовать для этой цели другие расширения.

Для предоставления возможности сравнения различных версий одной и той же страницы установлено расширение Wikidiff2.

Для подсветки синтаксиса языков программирования установлено расширение SyntaxHighlight версии 1.0.8.4.

## 2.2 Организация сайта

В настоящее время пользователь по ссылке <http://algowiki-project.org> попадает на русскоязычную страницу Открытой энциклопедии свойств алгоритмов AlgoWiki (рис. 2). Это временное решение, оно связано с тем, что на первом этапе все описания алгоритмов выполняются на русском языке. В дальнейшем предполагается одновременное наполнение как русскоязычной, так и англоязычной версий энциклопедии. Сейчас для явного доступа в англоязычной части можно использовать адрес <http://algowiki-project.org/en>. Для связывания какой-либо страницы со страницей из другой части сайта (например, русскоязычной страницы с англоязычной), в конце исходного кода помещается конструкция вида `[[en:Main Page]]`, после этого в боковой панели слева, в разделе «На других языках» появится ссылка «English». Для того чтобы просто вставить в текст гиперссылку на страницу из другой части сайта, необходимо написать `[[en:Main Page]]`. Примерно таким же образом многоязычность реализована и в Википедии.



**Рис. 2.** Главная страница Открытой энциклопедии свойств алгоритмов AlgoWiki

Для того чтобы в русскоязычной и англоязычной частях не требовалось дублировать общие элементы описаний (такие, например, как рисунки), введен специальный раздел сайта, называемый вики-складом (<http://algowiki-project.org/pool>), предназначенный для хранения файлов, которые потенциально могут использоваться в обеих частях. Чтобы вставить на страницу изображение, загруженное на вики-склад, использовать Interwiki не нужно, достаточно просто написать `[[File:my_image.png]]`. Система сконфигурирована так, что при отсутствии локального файла, она ищет его на вики-складе. При этом оставлена возможность загружать файлы локально, а при конфликте имен действует приоритет локального хранилища.

Все три раздела (русскоязычный, англоязычный и вики-склад) поддерживают единые учетные записи. То есть, достаточно один раз зарегистрироваться на любом из разделов, чтобы потом работать со всеми.

Главная часть Открытой энциклопедии свойств алгоритмов AlgoWiki находится в разделе «Классификация алгоритмов». В этом разделе доступ ко всем описаниям алгоритмов производится по группам, соответствующим типам выполняемых операций.

Для того чтобы всеми пользователями однозначно понимались термины, применяемые в описаниях алгоритмов, введен специальный раздел «Глоссарий». Пример статей из Глоссария можно увидеть на рис. 3. По принятым на данный момент соглашениям ссылка на термин в Глоссарии вставляется на первое вхождение данного термина в редактируемой странице.

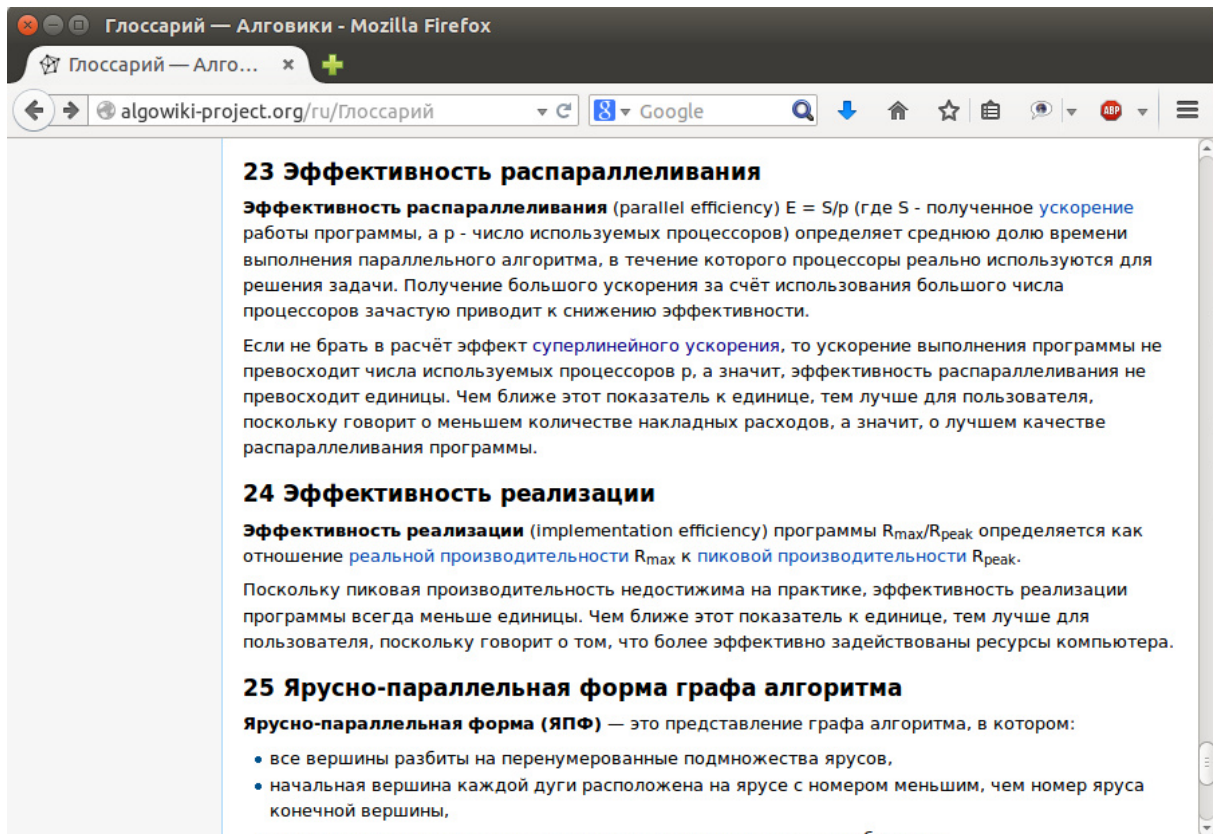


Рис. 3. Фрагмент Глоссария Открытой энциклопедии свойств алгоритмов AlgoWiki

Для того чтобы по одной схеме заполнять все разделы описания свойств разных алгоритмов, началось формирование раздела «Руководства и методики», куда заносятся пошаговые инструкции по получению данных для соответствующих разделов описаний.

Само описание свойств конкретного алгоритма строится по приведенной в разделе 1.2 структуре. В начале описания помещается карточка алгоритма, в которую выносятся основные свойства, характеризующие данный алгоритм. Информация из карточки алгоритма должна позволить читателю быстро оценить, насколько данный алгоритм может быть ему полезен. На рис. 4 приведено начало описания разложения Холецкого из Открытой энциклопедии свойств алгоритмов AlgoWiki с показанной карточкой алгоритма. В дальнейшем планируется дополнять информацию в карточке алгоритма экспертными оценками (например, «хорошо реализуется только на компьютерах с общей памятью», «позволяет эффективно использовать графические ускорители» и т.д.) а также данными об эффективности наиболее популярных реализаций.

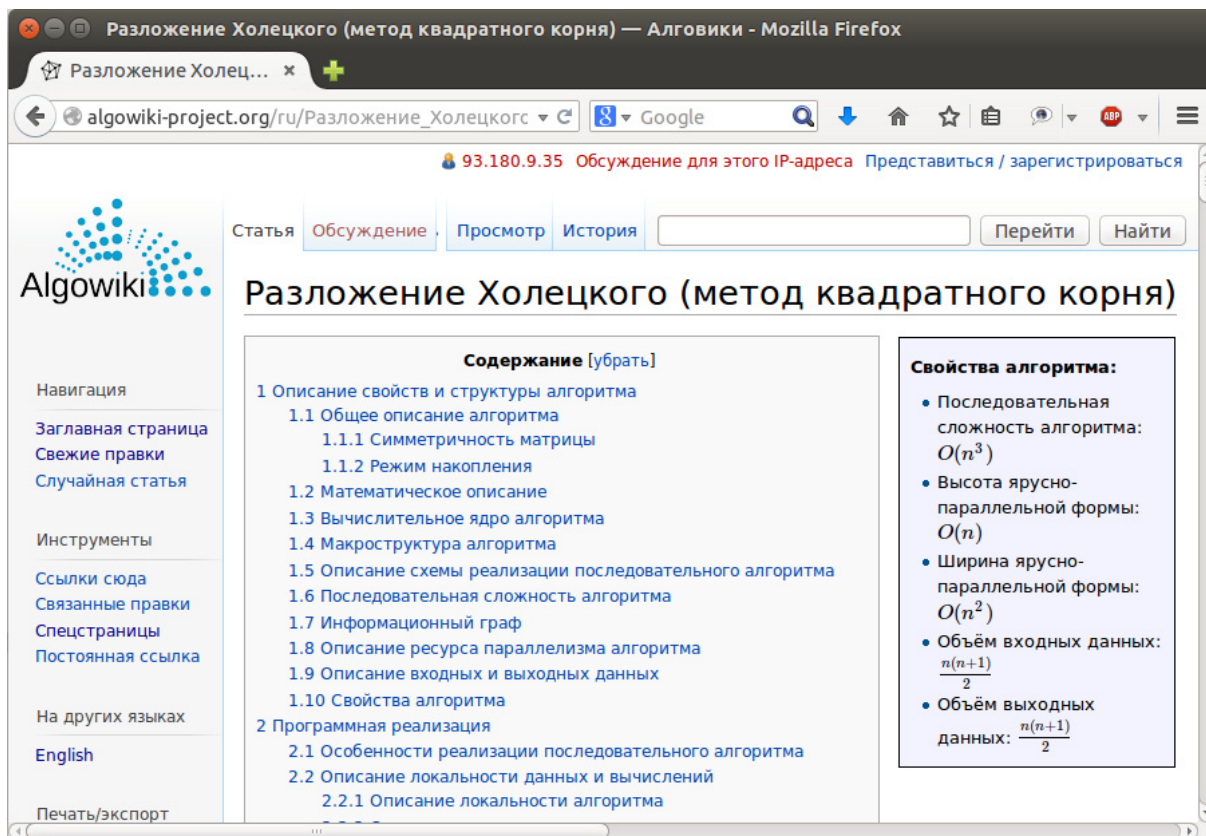


Рис. 4. Начало описания разложение Холецкого с карточкой алгоритма

Некоторые алгоритмы являются вариантами реализации какого-то общего метода решения математической задачи. В этом случае представляется разумным вводить дополнительный верхний уровень описания, а именно страницу с описанием общего метода решения, с которой идут ссылки на описания реализующих данный метод алгоритмов. В некоторых случаях представляется достаточным подробно описать некий базовый алгоритм, а на странице описания метода указать отличия других вариантов реализации алгоритмов от базового. Так, сейчас в рамках текущего варианта Открытой энциклопедии свойств алгоритмов AlgoWiki для метода Холецкого выполнено подробное описание базового точечного вещественного варианта для плотной симметричной положительно-определенной матрицы, а далее на странице метода Холецкого будут описываться отличия блочного вещественного варианта для плотной симметричной положительно-определенной матрицы, варианта для плотной комплексно-симметричной матрицы и т.д.

В конце любых описаний Открытой энциклопедии свойств алгоритмов AlgoWiki приводится список используемых источников. Также предполагается возможность указания авторства людей, осуществивших это описание.

### 2.3 Организация работы с сайтом

Для начала работы с сайтом пользователь должен зарегистрироваться. На данный момент регистрация свободная, но в будущем планируется введение какого-либо режима подтверждения регистрации. Сразу после получения доступа пользователь может создавать и редактировать страницы. Почти все действия по структурированию и обработке содержимого делаются пользователями вручную. Для разметки редактируемой страницы можно использовать вики-текст, вставляя вручную или с помощью встроенного редактора соответствующие теги.

Поскольку описание структуры алгоритмов предполагает широкое использование формул, для их набора используются конструкции LaTeX, заключенные в теги  $...$ . Для задания выключенных формул (тех, которые пишутся с новой строки и центрируются), используется конструкция  $...$  (начинается с двоеточия).



Для создания новой страницы достаточно на ссылающейся странице ввести конструкцию вида [[Новая страница]] и перейти по автоматически получившейся ссылке. Есть специальные конструкции для установления ссылки на существующую страницу, для вставки рисунков и т.д. Для загрузки рисунка нужно использовать ссылку «Загрузить файл» в левой колонке. Лучше всего рисунки загружать на вики-склад.

Пользователю предоставляется возможность как создать pdf-файл из текущей страницы (нажав ссылку «Скачать как PDF» в левой колонке), так и выбрать конкретные страницы, из которых он хочет собрать pdf-файл (нажав ссылку «Создать книгу» в левой колонке).

Все последние произведенные действия с сайтом можно посмотреть по ссылке «Свежие правки» в левой колонке сайта.

### **3. Начальное наполнение Открытой энциклопедии**

#### **3.1 Классификация алгоритмов**

В разделе «Классификация алгоритмов» делается попытка разложить все описываемые алгоритмы по тематическим разделам. Предполагается, что эта классификация будет со временем видоизменяться и пополняться по мере подключения к проекту специалистов, описывающих алгоритмы из других научных областей. Ничему также не противоречит, если какие-то алгоритмы будут входить сразу в несколько разделов.

На момент написания статьи классификация алгоритмов имеет такой вид:

1. Векторные операции
2. Умножение матрицы на вектор
3. Матричные операции
4. Разложения матриц
  - 4.1 Треугольные разложения
  - 4.2 Унитарно-треугольные разложения
  - 4.3 Разложения на унитарные и хессенберговы матрицы
  - 4.4 Разложения на унитарные и диагональные матрицы
5. Решение систем линейных уравнений
6. Тесты производительности компьютеров
7. Преобразование Фурье
8. Алгоритмы на графах
9. Алгоритмы поиска
10. Алгоритмы сортировки
11. Вычислительная геометрия
12. Компьютерная графика
13. Криптографические алгоритмы
14. Нейронные сети
15. Алгоритмы оптимизации
16. Алгоритмы теории игр
17. Алгоритмы моделирования квантовых систем
  - 17.1 Алгоритмы моделирования квантовых вычислений
18. Другие алгоритмы

#### **3.2 Список алгоритмов, взятых для начального наполнения**

Для начального наполнения Открытой энциклопедии свойств алгоритмов AlgoWiki был выбран ряд алгоритмов от весьма простых до достаточно сложных и содержательных. Этот набор в будущем должен послужить образцом при включении в проект новых специалистов, описывающих свои предметные области.

На момент написания статьи в разной степени готовности на сайте представлены следующие алгоритмы:

- Разложение Холецкого (метод квадратного корня)
- Суммирование сдвигами
- Равномерная норма вектора, вещественная версия, последовательный вариант
- Равномерная норма вектора, вещественная версия, последовательно-параллельный вариант
- Скалярное произведение векторов, вещественная версия, последовательный вариант
- Скалярное произведение векторов, вещественная версия, последовательно-параллельный вариант
- High Performance Conjugate Gradient (HPCG) benchmark
- Linpack benchmark
- Сумма вектора и произведения матрицы на другой вектор, вещественная версия, последовательный вариант, плотная матрица
- Сумма вектора и произведения матрицы на другой вектор, вещественная версия, последовательный вариант, разреженная матрица
- Последовательно-параллельный метод суммирования
- Последовательно-параллельный метод нахождения всех частных выражений для ассоциативных операций
- Схема Горнера, вещественная версия, последовательный вариант
- Сдвиг аргументов многочлена по схеме Горнера, вещественная версия, последовательный вариант
- Решение правой двухдиагональной СЛАУ, вещественная версия, последовательный вариант
- Решение правой двухдиагональной СЛАУ с единичной диагональю, вещественная версия, последовательный вариант
- Решение левой двухдиагональной СЛАУ, вещественная версия, последовательный вариант
- Решение левой двухдиагональной СЛАУ с единичной диагональю, вещественная версия, последовательный вариант
- Разложение трехдиагональной матрицы, вещественная версия, последовательный вариант (первая стадия прогонки)
- Разложение трехдиагональной матрицы, вещественная версия, последовательный вариант без корней (первая стадия прогонки)
- Разложение трехдиагональной матрицы, вещественная версия, последовательный вариант с корнями (первая стадия прогонки)
- Быстрое преобразование Фурье для степеней двойки
- Умножение плотных матриц
- Умножение плотной матрицы на вектор
- Метод Гаусса решения СЛАУ (прямой ход)
- Метод Гаусса решения СЛАУ (обратный ход)
- Квадратурные (кубатурные) методы численного интегрирования по отрезку (многочленному кубу)

## 4. Заключение

Основная задача, которая ставится в данном проекте – это разработка фундаментальных основ и формализация процесса отображения алгоритмов на архитектуру параллельных вычислительных систем. Решение данной задачи позволит получить ответ на многие вопросы, такие как:

- Каковы фундаментальные основы процесса разработки эффективного программного обеспечения?
- Что в реальности влияет на эффективность?

- Что здесь можно воспринять из сделанного ранее, чтобы не повторять одну и ту же работу снова и снова?
- Как сохранить инвестиции (время, финансы...), вкладываемые в разработку программного обеспечения для высокопроизводительных вычислительных систем?

Создаваемая Открытая энциклопедия свойств алгоритмов AlgoWiki не имеет прямых аналогов в мировой практике. Проблема тут не в том, чтобы сделать и опубликовать энциклопедию, работающую на принципах вики-технологий. Проблема в том, что нет не только адекватного контента, но даже понимания, как подобный контент можно было бы формировать.

Первая версия Открытой энциклопедии свойств алгоритмов AlgoWiki не является окончательной и будет модифицироваться и совершенствоваться. Однако на ней отрабатываются те принципы, которые хочется заложить в окончательный вариант энциклопедии, наполняемой мировым вычислительным сообществом.

Данные о динамических характеристиках конкретных реализаций алгоритмов, описанных в рамках Открытой энциклопедии свойств алгоритмов AlgoWiki, получены на суперкомпьютерах СКИФ МГУ «Чебышев» [23] и «Ломоносов» [24].

Исследование выполнено в Московском государственном университете имени М.В. Ломоносова за счет гранта Российского научного фонда (проект N14-11-00190).

## Литература

1. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. СПб.: БХВ-Петербург, 2002. 608 с.
2. Гергель В.П. Высокопроизводительные вычисления для многопроцессорных многоядерных систем.- М.: Издательство Московского университета, 2010. 544 с.
3. Воеводин В.В. Математические основы параллельных вычислений// М.: Изд. Моск. ун-та, 1991. 345 с.
4. Воеводин В.В. Информационная структура алгоритмов// М.: Изд. Моск. ун-та, 1997. 139 с.
5. Воеводин Вл.В. Теория и практика исследования параллелизма последовательных программ// Программирование. 1992. N 3. С. 38–53.
6. Воеводин Вл.В. Статический анализ и вопросы эффективной реализации программ// Вычислительные процессы и системы (Под ред. Г.И. Марчука), М.: «Физматлит», 1993. N 9. С. 249–301.
7. Вл.В.Воеводин. Описание входных и выходных данных фрагментов программ// Вестник Московского университета. Серия 15. 1997. N 1. С. 41–44.
8. Воеводин Вад.В. Характеристики типовых алгоритмических структур // Вестник Нижегородского университета, 1(2):181–189, 2011.
9. Воеводин Вад.В. Визуализация и анализ профиля обращений в память // Вестник Южно-Уральского государственного университета. Серия Математическое моделирование и программирование, 17(234):76–84, 2011.
10. Воеводин Вад.В., Швец П.А. Метод покрытий для оценки локальности использования данных в программах // Труды XV Международной суперкомпьютерной конференции «Научный сервис в сети Интернет: все грани параллелизма», Абрау-Дюрсо, 2013. С. 176–180.
11. Воеводин Вл.В., Воеводин Вад.В. Спасительная локальность суперкомпьютеров // Открытые системы. 2013. № 9. С. 12–15.
12. Антонов А.С., Теплов А.М. Исследование масштабируемости программ с использованием инструментов анализа параллельных приложений на примере модели атмосферы NH3D // Вестник Южно-Уральского государственного университета. Серия «Вычислительная математика и информатика», Т. 2, N 1, 2013. С. 5–16.
13. Антонов А.С., Теплов А.М. Использование данных системного мониторинга для определения факторов, уменьшающих масштабируемость приложения // Научный сервис в сети Ин-

- тернет: многообразие суперкомпьютерных миров: Труды Всероссийской научной конференции (22–27 сентября 2014 г., г.Новороссийск). М.: Изд-во МГУ, 2014. С. 87–96.
14. Антонов А.С., Теплов А.М. О практической сложности понятия масштабируемости параллельных программ // Высокопроизводительные параллельные вычисления на кластерных системах (HPC 2014): Материалы XIV Международной конференции (г.Пермь, 10–12 ноября 2014 г.)-Пермь: Издательство Пермского национального исследовательского политехнического университета, 2014. С. 20–27.
  15. Теплов А.М. Об одном подходе к сравнению масштабируемости параллельных программ // Вычислительные методы и программирование: Новые вычислительные технологии (Электронный научный журнал). Том 15, выпуск 4, 2014. С. 697–711.
  16. Никитенко Д.А. Комплексный анализ производительности суперкомпьютерных систем, основанный на данных системного мониторинга // Вычислительные методы и программирование: Новые вычислительные технологии (Электронный научный журнал), том 15. С. 85–97.
  17. Антонов А.С., Жуматий С.А., Никитенко Д.А., Стефанов К.С., Теплов А.М., Швец П.А. Исследование динамических характеристик потока задач суперкомпьютерной системы // Вычислительные методы и программирование: Новые вычислительные технологии (Электронный научный журнал), том 14, № 2. С. 104–108.
  18. Адинец А.В., Брызгалов П.А., Воеводин В.В., Жуматий С.А., Никитенко Д.А., Стефанов К.С. JobDigest – подход к исследованию динамических свойств задач на суперкомпьютерных системах // Вестник Уфимского государственного авиационного технического университета, том 17, № 2 (55). С. 131–137.
  19. Воеводин Вл.В. Algowiki: открытая энциклопедия свойств алгоритмов // Вычислительные методы и программирование: Новые вычислительные технологии (Электронный научный журнал). (в печати)
  20. Воеводин Вл.В. Суперкомпьютер «Ломоносов», прикладные пакеты и особенности алгоритмов // Сборник докладов конференции СКТП–2014. Крыловский государственный научный центр. СПб, 2014. С. 25–28.
  21. MediaWiki, <http://www.mediawiki.org>
  22. Wikipedia, <https://www.wikipedia.org>
  23. Антонов А.С. СКИФ МГУ – основа Суперкомпьютерного комплекса Московского университета // Вторая Международная научная конференция «Суперкомпьютерные системы и их применение» (SSA'2008): доклады конференции (27–29 октября 2008 года, Минск).- Минск: ОИПИ НАН Беларуси, 2008. С. 7–10.
  24. Владимир Воеводин, Сергей Жуматий, Сергей Соболев, Александр Антонов, Петр Брызгалов, Дмитрий Никитенко, Константин Стефанов, Вадим Воеводин. Практика суперкомпьютера «Ломоносов» // Открытые системы, N 7, 2012. С. 36–39.