

**Ответы на замечания по статье**  
**«АВТОМАТИЗИРОВАННОЕ ПРОЕКТИРОВАНИЕ И ИСПОЛНЕНИЕ**  
**ЭФФЕКТИВНЫХ ПРОГРАММ ДЛЯ ЧИСЛЕННЫХ АЛГОРИТМОВ»**

Автор выражает благодарность рецензенту за ценные замечания и рекомендации, способствующие улучшению текста статьи.

<b>№ п/п</b>	<b>Замечание рецензента</b>	<b>Ответ авторов (отметка о выполнении и/или комментарий)</b>
<b>Рецензент В</b>		
1.	<p>В ответе на замечание 1 остается непонятным, откуда же возьмется какая-то другая реализация алгоритма, не использующая весь ресурс параллелизма алгоритма, которую предполагается выполнять на конкретной вычислительной системе. Если есть способ автоматического ее получения, было бы очень интересно с ним ознакомиться. Если же нет, то не очень понятно, чем при этом полезна Q-эффективная реализация.</p>	<p>Выполнено. Пояснения по данному замечанию внесены в текст статьи (последний абзац раздела 2). Q-эффективная программа использует весь ресурс параллелизма алгоритма. Но когда она выполняется на ПВС, то возможно, что из-за нехватки вычислительных ресурсов ПВС не все операции будут выполняться по мере их готовности к выполнению. В результате какие-то из операций, готовых к выполнению, будут выполнены позднее, чем могли бы быть выполнены при достаточных вычислительных ресурсах, а это значит, что будет выполняться реализация алгоритма, не использующая весь ресурс параллелизма алгоритма. Несмотря на данный факт, Q-эффективная реализация полезна тем, что, как пояснял автор в ответе на замечание 1 первого раунда рецензирования, разрабатывать и использовать Q-эффективные программы, т.е. программы, выполняющие Q-эффективные реализации, целесообразно, так как они имеют лучшие динамические характеристики при выполнении на ПВС, т.е. являются эффективными.</p>

№ п/п	Замечание рецензента	Ответ авторов (отметка о выполнении и/или комментарий)
2.	<p>В ответе на замечание 2 (а также в тексте статьи) по-прежнему нет определения того, что автор понимает под эффективностью.</p> <p>Известно несколько определений этого понятия, и какое из них тут используется, неясно.</p>	<p>Выполнено.</p> <p>Пояснения по данному замечанию внесены в текст статьи (предпоследний абзац раздела 2).</p>
3.	<p>В ответе на замечание 3 (а также в тексте статьи) по-прежнему неясно, что за "специфику" имеют программы, получающиеся с помощью автоматизированного проектирования, что при этом возникает проблема их исполнения.</p>	<p>Выполнено.</p> <p>Пояснения по данному замечанию внесены в текст статьи (добавлены предложения 2 и 3 в абзац 4 раздела 4). В качестве формата файла программы используется разработанный специально для данной программной системы бинарный формат, достаточно подробно описанный в статье. В этом и заключается специфика программы для нашей программной системы. Чтобы исполнять такие программы, являющиеся результатом работы компилятора, потребовалось разработать программное обеспечение, которое было названо виртуальной машиной. Мы не утверждаем, что все «программы, получающиеся с помощью автоматизированного проектирования», должны иметь специфику, из-за которой возникает проблема их исполнения. Однако пример решения, предложенного для нашей программной системы, показывает, что специфика программ возможна.</p>
4.	<p>Ответ на замечание 5 что-то разъясняет, но не в тексте статьи, к тому же эта терминология по-прежнему кажется не слишком удачной.</p>	<p>Пояснение по поводу данного замечания в текст статьи не внесено, так как редакция журнала не предложила это сделать.</p> <p>Однако, учитывая данное замечание, автором в тексте статьи сделана замена слова «потеряна» на слово «не учтена».</p>

№ п/п	Замечание рецензента	Ответ авторов (отметка о выполнении и/или комментарий)
5.	<p>Утверждение в ответе на замечание 6 "Схема сдваивания обеспечивает минимальное время вычисления цепочки" требует, как минимум обоснования, особенно если говорить именно про время - вполне возможны архитектуры компьютеров, где это будет не так.</p>	<p>Выполнено.  Пояснения по данному замечанию внесены в текст статьи (добавлены после определения 6 в раздел 2).  Одно из обоснований утверждения "Схема сдваивания обеспечивает минимальное время вычисления цепочки" следующее. Рассмотрим все множество алгоритмов, вычисляющих цепочку длины <math>n</math>, образованную некоторой ассоциативной операцией, из которой состоит множество операций <math>Q</math>. Высота алгоритмов составляет не менее <math>\lceil \log n \rceil</math> и не более <math>n - 1</math>. При этом высота алгоритма, использующего схему сдваивания, равна <math>\lceil \log n \rceil</math>, т.е. является минимальной. Предположим, что программы, реализующие алгоритмы, имеют одинаковую вычислительную инфраструктуру, т.е. условия разработки и выполнения. Тогда меньше всего времени для выполнения последовательных инструкций понадобится программе для реализации алгоритма, использующего схему сдваивания. Следовательно, в соответствии с законом Амдала программа, реализующая алгоритм с использованием схемы сдваивания, будет иметь время выполнения не больше, чем каждая из остальных программ.</p>

№ п/п	Замечание рецензента	Ответ авторов (отметка о выполнении и/или комментарий)
6.	<p>В ответе на замечание 7 идет ссылка на энциклопедию AlgoWiki, однако там понятия "высота" и "ширина" применяются к ярусно-параллельной форме алгоритма, применение же их просто к алгоритму не вполне корректно, либо нужно давать точное определение.</p>	<p>Пояснение по поводу данного замечания в текст статьи не внесено, так как редакция журнала не предложила это сделать.</p> <p>Ответ автора на замечание 7 был дан в следующей формулировке:</p> <p>«Следует отметить, что аналогичные понятия, но связанные с информационными графами алгоритмов, введены в [4] и используются в энциклопедии AlgoWiki [9].» Понятия высоты и ширины, применяемые к ярусно-параллельной форме алгоритма, являются аналогами понятий высоты и ширины алгоритма, введенными автором в рамках концепции Q-детерминанта, так как ярусно-параллельная форма описывает алгоритм и является аналогом Q-эффективной реализации. Следует отметить, что Валентин Васильевич Воеводин употреблял в этом случае термин «высота алгоритма», например, в [4] на стр. 196 написано: «Минимальную высоту параллельных форм алгоритма будем называть высотой алгоритма и т.д.» Этот термин встречается и в других работах.</p>
7.	<p>В ответе на замечание 15: "известно заранее, что выполняющие их программы не имели бы времени выполнения лучше, чем Q-эффективные программы" - утверждение не очевидно. Даже если это теоретически так, на конкретной вычислительной системе могут оказаться важными совсем другие факторы, и Q-эффективная программа вполне может оказаться медленнее. Поэтому реальное сравнение с известными и эффективными реализациями помогло бы подтвердить точку зрения автора.</p>	<p>Пояснение по поводу данного замечания в текст статьи не внесено, так как редакция журнала не предложила это сделать.</p> <p>Экспериментальные исследования, описанные в [3, 15], подтверждают утверждение автора.</p>