

# ОРГАНИЗАЦИЯ УДАЛЕННОГО ВЫПОЛНЕНИЯ ТРУДОЕМКИХ РАСЧЕТОВ ПРИ ПРОЕКТИРОВАНИИ СВИС

Д. И. Черемисинов, Л. Д. Черемисинова

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: {cher, cld}@newman.bas-net.by

*Предлагается методика и программные средства, обеспечивающие совместимость по данным и организацию обмена данными между кластерным компьютером и системами автоматизации проектирования для решения трудоемких задач из области логического проектирования.*

## ВВЕДЕНИЕ

Практически все задачи из области теории дискретных автоматов и логического проектирования носят комбинаторный характер и имеют экспоненциальную сложность. В последнее время при решении сложных вычислительных задач все чаще используют многопроцессорные вычислительные системы, наиболее популярным типом которых являются кластеры.

В настоящей работе предлагается сервис грид-системы для выполнения трудоемких расчетов при проектировании СВИС. Сервисный подход для управления работой программой кластерного компьютера привлекателен тем, что ориентирован на использование открытых, универсальных протоколов и интерфейсов. Описывается технология и программные средства, обеспечивающие совместимость по данным и организацию обмена данными между суперкомпьютером кластерного типа и системой автоматизации проектирования для решения трудоемких задач. Предлагаемая технология демонстрируется на примере выполнения этапа верификации результатов проектирования в рамках системы автоматизации логического проектирования заказных КМОП СВИС [1] на основе удаленного решения на суперкомпьютере задачи нахождения корней логических уравнений. Выбор именно этой задачи обусловлен тем, что решение логических уравнений в конъюнктивной (КНФ) или дизъюнктивной нормальной форме (ДНФ) требуется в задачах верификации и синтеза цифровых СВИС.

## I. ОРГАНИЗАЦИЯ ВЫЧИСЛЕНИЙ НА СУПЕРКОМПЬЮТЕРЕ КЛАСТЕРНОГО ТИПА

Большинство кластерных компьютеров (как и используемый суперкомпьютер семейства СКИФ [2]) имеет архитектуру Беовульф на основе автономных вычислительных узлов, соединенных собственной компьютерной сетью, и использования Linux в качестве операционной системы. Управляющая машина (front-end node) является файл-сервером для вычислительных узлов, консолью кластера и шлюзом во внешнюю сеть. Остальные узлы кластера использу-

ются только для выполнения исполнимого кода MPI-программы, которая копируется в память каждого из узлов, выделенных управляющей машиной для ее выполнения. Узлы включаются в работу одновременно, работают параллельно, распознавая и выполняя свои участки MPI-программы и обмениваясь сообщениями при необходимости. Пользователь запускает MPI-программу через управляющую машину кластера, которая доступна по сети через протокол Интернет. Входные данные программы должны быть заранее размещены в файловой системе, туда же MPI-программой помещается результат, доступный после окончания ее работы.

Для запуска заданий на кластере используется система управления заданиями PBS (Portable Batch System), работающая на управляющей машине кластера. Запуск MPI-программы выполняется с помощью запускающего скрипта (командного файла PBS), в котором указывается запускаемая программа и требуемые ресурсы. Для связи с окружающей средой управляющая машина кластера использует OpenSSH сервер, предоставляющий шифрование сеансов связи по компьютерным сетям с использованием протокола SSH.

Выполняемый удаленно метод решения логических уравнений реализован MPI-программой «Вычислитель» для кластерного компьютера. Исходными данными для программы служит описание КНФ в текстовом виде или в формате DIMACS, который позволяет компактно представлять редкие троичные матрицы, каковые получаются при формальной верификации схемных описаний. Результатом выполнения программы является текст, содержащий режим и длительность решения, а также выполняющий вектор, если КНФ выполнима.

## II. АРХИТЕКТУРА РАСПРЕДЕЛЕННОЙ ПРОГРАММЫ

Грид-сервис для удаленного выполнения трудоемких расчетов при проектировании СВИС представляет собой надстройку над обычными веб-сервисами, имеет типовую структуру грид-систем для высокопроизводительных вычисле-

ний и представляет собой трехслойную систему: программа «Вычислитель» для суперкомпьютера, сервис связи и агент пользователя. Агент пользователя взаимодействует с «Вычислителем» через сервис связи. Пользователь может управлять грид-сервисом посредством такого агента как веб-браузер или прямо из САПР СБИС.

Грид-сервис построен на основе открытых стандартов и межплатформенного связующего программного обеспечения (middleware), являющегося посредником между агентом пользователя и программами, выполняемыми на удаленных компьютерах. Сервис связи базируется на технологии интеграции веб-служб протоколом SOAP для обмена структурированными сообщениями в распределенной вычислительной среде. SOAP основан на языке XML и расширяет протокол прикладного уровня. Агенты, взаимодействующие по этому протоколу, имеют простое поведение: запрос/ответ. Агент-отправитель SOAP отправляет посредством HTTP-запроса XML-сообщение и получает результат в HTTP-отклике от агента-получателя.

Протокол работы грид-сервиса состоит из следующих шагов: 1) транспортировать файл с анализируемой КНФ в файловую систему на управляющей машине кластера; 2) запустить программу «Вычислитель» решения систем логических уравнений на суперкомпьютере; 3) вернуть файл с результатами решения на клиентский компьютер.

Клиентская часть грид-сервиса взаимодействует с программой на суперкомпьютере посредством интерфейса Middleware UNICORE, который имеет трёхслойную архитектуру: пользовательский уровень, сервер UNICORE и суперкомпьютер. Пользовательский уровень и UNICORE сервер связаны открытой сетью и взаимодействуют через протокол SSL, который для установления клиент-серверного соединения использует криптографию с открытым ключом, так что компонент пользовательского уровня должен иметь секретный ключ шифрования.

Интерфейс UNICORE ориентирован на пакетную обработку заданий в суперкомпьютере, которые описывают работу, подлежащую выполнению на суперкомпьютере в формате представления заданий UNICORE. В задании указывается, какие данные импортируются (интерфейсом UNICORE) перед выполнением программы в файловую систему кластерного компьютера и то, что нужно экспортировать после того, как задание завершится. Используемая версия Unicore 6 использует стандарты, основанные на XML: SOAP для связи пользовательского уровня и сервера, JSDL для представления работы, SAML для аутентификации и авторизации между пользовательским уровнем и сервером и XACML для аутентификации и авторизации между сервером и суперкомпьютером.

### III. ГРИД-СЕРВИС В МАРШРУТЕ ПРОЕКТИРОВАНИЯ СБИС

Методика запуска на кластерном компьютере программ решения комбинаторных задач, возникающих в процессе проектирования, отработана на примере разработанного в лаборатории логического проектирования ОИПИ НАН Беларуси программного комплекса ЭЛС «Энергосберегающий логический синтез» [1]. Одним из ключевых этапов проектирования в рамках комплекса является верификация проектных решений на всех шагах проектирования. Одним из реализованных в рамках ЭЛС методов верификации является формальная верификация, основанная на сведении задачи к проверке выполнимости КНФ, отражающей структуру сравниваемых описаний. Основные вычислительные затраты падают, как правило, на анализ выполнимости этой КНФ, именно поэтому он выполняется с привлечением средств кластерного компьютера.

Так как набор проектных операций в системе ЭЛС на уровне интерфейса пользователя фиксирован, то для интеграции грид-сервиса решения логических уравнений программа верификации была модифицирована. После получения КНФ в процессе выполнении операции верификации в среде комплекса ЭЛС вместо вызова внутренней программы решения логических уравнений (SAT-Solver Minisat) выполняются следующие действия: анализируемая КНФ в формате DIMACS транспортируется на управляющую машину кластера; запускается программа «Вычислитель» на суперкомпьютере; файл с результатами решения возвращается в систему ЭЛС. Если файл содержит найденный вектор (что говорит о выполнимости КНФ), то делается вывод о неэквивалентности (или нереализуемости) сравниваемых описаний, иначе описания эквивалентны.

С целью испытания предлагаемого подхода к распределенному решению задачи верификации в среде программного комплекса ЭЛС был верифицирован ряд структурных описаний, возникающих в процессе проектирования. О размерности решаемых задач можно судить по следующему примеру: верифицируемые схемы состояли из 1923 элементов КМОП библиотеки, КНФ разрешения в этом случае зависела от 13103 переменных и содержала 38635 дизъюнктов. Общее ограничение программы «Вычислитель» на предельный размер КНФ составляют около 109 дизъюнктов.

1. Бибило, П. Н. Автоматизация логического синтеза КМОП схем с пониженным энергопотреблением / П. Н. Бибило, Л. Д. Черемисинова, С. Н. Кардаш и др. // Программная инженерия. – 2013. – № 8. – С. 35–41.
2. Абрамов, С. М. Принципы построения суперкомпьютеров семейства «СКИФ» и их реализация / С. М. Абрамов, Н. Н. Парамонов, В. В. Анищенко, С. В. Абламейко // Информатика. – 2004. – № 1. – С. 89–106.