МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования   
**«Национальный исследовательский   
Нижегородский государственный университет им. Н.И. Лобачевского»**

**(ННГУ)**

**Институт информационных технологий, математики и механики**

**Кафедра: МОСТ**

Направление подготовки: «Фундаментальная информатика и информационные технологии»

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА**

**Тема:**

**«TODO»**

Допущена к защите Выполнил:

студент группы

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Гладков Дмитрий Олегович

ученая. степень, ученое звание, ф.и.о.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ подпись подпись

Научный руководитель:

к.т.н. Сысоев Александр Владимирович

ученая степень, ученое звание, ф.и.о.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись

Рецензент:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ученая степень, ученое звание

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись

Нижний Новгород  
2018

Оглавление

[Оглавление 2](#_Toc483333009)

[Аннотация 2](#_Toc483333010)

[Введение 2](#_Toc483333011)

[Цель работы 2](#_Toc483333012)

[Обзор предметной области 2](#_Toc483333013)

[Мультимасштабное картографирование и генерализация картографических объектов 2](#_Toc483333014)

[Свойства и задачи мультимасштабного картографирования 2](#_Toc483333015)

[Автоматический процесс генерализации 2](#_Toc483333016)

[Методы оценивания алгоритмов генерализации 2](#_Toc483333017)

[Алгоритмы генерализации 2](#_Toc483333018)

[Алгоритм генерализации, использующий автоматическую сегментацию и упрощение полученных сегментов с помощью алгоритма Дугласа-Пейкера 2](#_Toc483333019)

[Приведение ломаной к равнозвенному виду 2](#_Toc483333020)

[Сегментация 2](#_Toc483333021)

[Вычисление фрактальной размерности каждого сегмента 2](#_Toc483333022)

[Упрощение 2](#_Toc483333023)

[Сглаживание сплайнами 2](#_Toc483333024)

[Описание программного комплекса 2](#_Toc483333025)

[Клиентское приложение 2](#_Toc483333026)

[Серверное приложение 2](#_Toc483333027)

[Литература 2](#_Toc483333028)

Аннотация

За последние десятилетия были разработаны и представлены несколько высокоскростных сетей, каждая из которых отличается от других показателями производительности, коммуникационными особенностями и возможностями. Однако инициативы по разработке новых сетевых решений сопровождаются увеличением расхождений в коммуникационных интерфейсах или “языках”, используемых каждой сетью. Соответственно, переносимость приложений на различных коммуникационных интерфейсах стала темой обширных исследований. В качестве основного средства для достижения переносимости приложений широко использовались такие модели программирования, как Sockets, Shared Memory, Message Passing Interface (MPI), Symmetric Hierarchical MEMory (SHMEM), Global Arrays (GA), Charm++ и другие.

В данной работе исследуются различные подходы для создания программной модели, которая может быть использована различными параллельными программными моделями, такими как MPI, SHMEM и другими, в различных высокоростных сетевых средах (например, InfiniBand, Omni-Path Architecture, Ethernet, а также решения, основанные на Ethernet – iWARP, RoCE). В частности, исследования в данной работе посвящены:

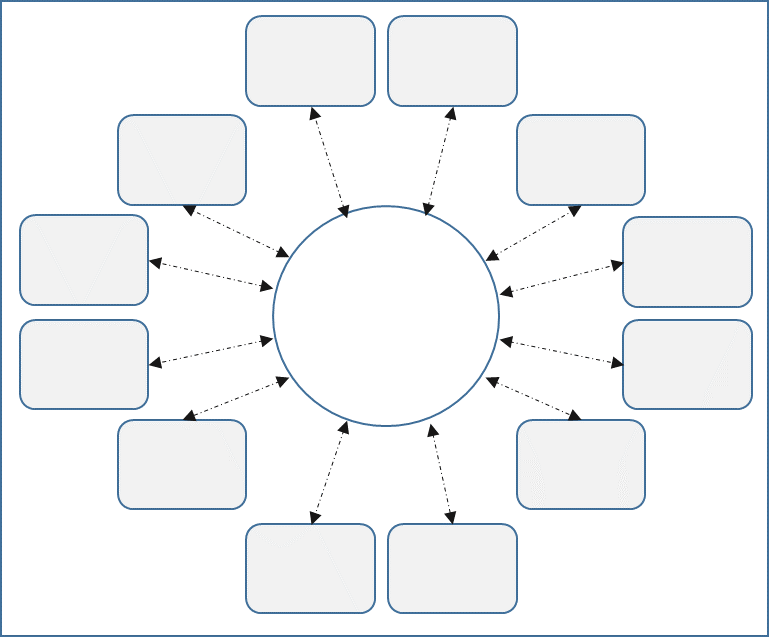
1. анализу существующих программных моделей для параллельного программирования и их потребностей коммуникационным интерфейсам;
2. рассмотрению сущестующих и широкоиспользуемых программных интерфейсов для работы с высокоскоростными сетями и возможности построения единой программной модели, способной обеспечить переносимость приложений;
3. анализу ограничений перспектиных коммуникационных интерфейсов для мультисетевого программирования, рассмотрение и разработка возможностей для обеспечения наибольшей эффективности сетевого стека для параллельных программных моделей.

Введение

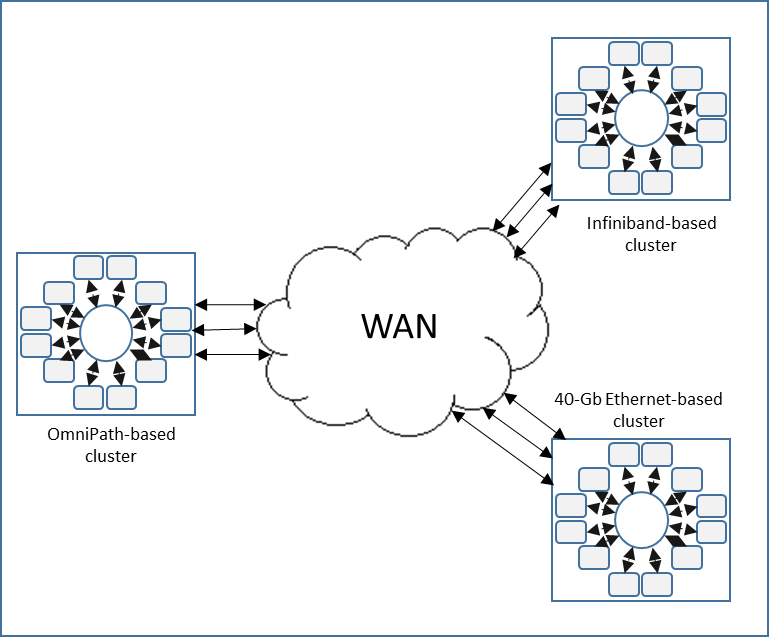
За последние несколько десятилетий были достигнуты большие успехи в области компьютерных технологий. Компьютеры стали повсеместными во всех областях жизнедеятельности человека, будь то промышленность или исследования, и теперь оказывают влияние на каждый аспект человечской жизни. В начале 70-х, когда обычные компьютеры набирали популярность среди исследователей, было понятно, что возникла, необходимость в более мощных машинах, которые могли бы решить проблемы, которые слишком сложные для обычных компьютеров. Это осознание привело к развитию суперкомпьютеров – передовых и мощных машин, состоящих из нескольких обрабатывающих подсистем. Cray-1, разработанный Cray Research (ныне, Cray Inc.), был одним из таких мощных суперкомпьютеров. Развитие суперкомпьютеров и компьютерных технологий сопутствовало экспоненциальному росту требований приложений. Высокая стоимость проектирования, разработки и поддержки таких суперкомпьютеров для обеспечения высокой производительности приложений требовала, чтобы исследователи искали альтернативу этим суперкомпьютерам в виде кластерных систем (кластеров).

Кластеры состоят из недорогих коммерческих, готовых к использованию без дополнительных модификаций компьютеров (Commercial off-the-shel - COTS), соединенных между собой посредством сети. ПК в кластере взаимодействуют друг с другом по сети, чтобы вести себя как суперкомпьютер. Кластеры становятся все более популярными в различны сферах применения, главным образом, из-за высокой производительности и стоимости. Такие системы теперь могут быть спроектированы для достижения различного уровня производительности из-за увелечения производительности производимых процессоров, памяти и сетевых технологий.

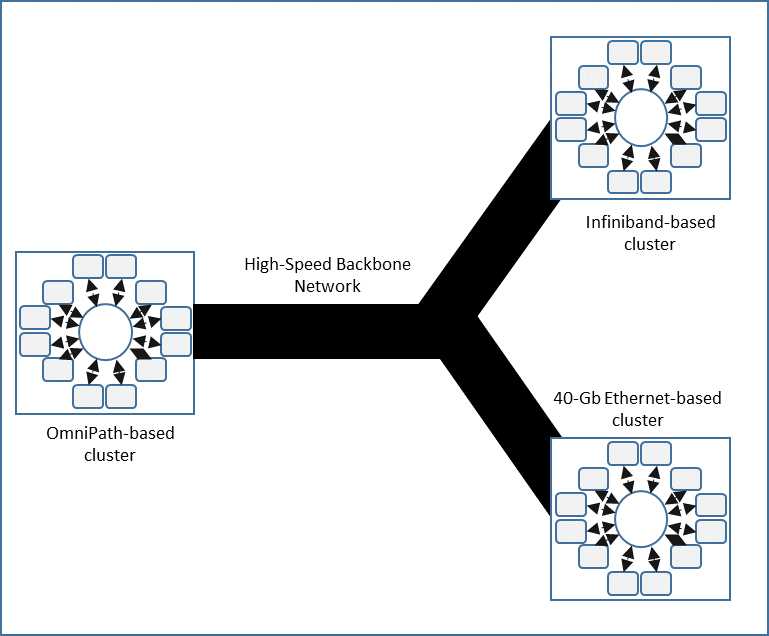
Поскольку системы на основе кластеров опирается на опирается на возможности нескольких недорогих компьютеров, взаимодействующих друг с другом по сети, способность сети (аппаратное обеспечение, а также связанное с ним программное обеспечение) является важным компонентом в его эффективности и масштабируемости. На Рис. 1.1 показаны типичные среды, используемые параллельными и распределенныи вычислительными приложениями. Среды могут варьироваться от локальный сетей внутри одного кластера кластера до нескольких различных кластеров, подключенных через WAN или высокоскоростную магистральную сеть.



(а)



(b)



(c)

Рис. 1.1. Типичные коммуникационные среды для распределенных и параллельных приложений:

(a) Связь внутри локальной сети кластера

(b) Межкластерная связь по WAN

(c) Межкластерная связь по высокоскоростной магистральной сети

За последние несколько лет, на рынке был представлен ряд высокоскоростных сетей, включая 1/10/40/100-гигабитный Ethernet, InfiniBand, Myrinet, Omni-Path и т.д. С появлением таких сетей, коммуникационные издержки в кластерных системах переходят от самой сети к сетевым протоколам на стороне отправителя или получателя. Сетевые протоколы предыдущего поколения, такие как TCP/IP, полагаются на ядо операционной системы для обработки сообщений. Это вызывает множество копирований и переключений контекста ядра на критического пути обработки сообщения. Таким образом, накладные расходы на связь были высокими. В течение последних нескольких лет исследователи рассматривают альтернативы для увеличения производитльности коммуникаций в кластерах, посредством разработки протоколов низкого уровня с низкими задержками и высокой пропускной способностью, таких как FM и GM для Myrinet, EMP для Gigabit Ethernet и т.д. Чтобы стандартизовать эти решения, в конце 90-х годов была предложена архитектура виртуального интерфейса (VIA), но он не смог добиться успеха. В последние годы в индустрии была стандартизована InfiniBand Architecture для разработки кластеров нового поколения. Все это приводит к сокращению отставания между функциональными возможностями физической сети и тем, что получают конечные пользователи.

Тем не менее, каждая новая сеть и протоклы пользовательского уровня предоставляют пользователю новые API (Application Programming Interface) или “язык” для взаимодействия с пользователем. Хотя эти новые API могут быть эффективно использованы для разработки новых приложений, они могут оказаться непригодными для уже существуюющих приложений, которые были разработаны несколько лет назад. Разработчики приложений при написании приложений нацелены на переносимость на различные будущие платформы и сети. Модели программирования, такие как MPI и Sockets, широко признаны в качестве приемлемого подхода для достижений такой переносимости.

В течение последних нескольких лет MPI является де-факто стандратом для научных приложений. С другой стороны, интерфейс Sockets является наиболее широко используемой моделью программирования для традиционных научных приложений, коммерческих приложений, файловых систем и т.д.

Цель работы

Имея ввиду поставленные проблемы Sockets интерфейса, которые не позволяют раскрыть полностью возможности сетей, и проблемы специализированных API, разработанных для конкретных сетей, которые ограничивают переносимость приложений, в данном дипломном проекте предлагается рассмотреть перспективные подходы к созданию унифицированного интерфейса для различных параллельных технологий со следующими возможностями:

1. Позволять приложениям не только работать непосредственно в различных высокоскоростных сетях без каких-либо изменений, но также иметь возможность извлекать наилучшую производительность из сети.
2. Иметь возможность извлекать выгоду от предоставляемых способностей сетей, не ограничевая производительность и переносимость при переходе на другие сетевые технологии.
3. Обеспечивать надежность передачи данных, высокую масштабируемость приложений.

Цель данной работы состоит в том, чтобы:

1. Исследовать один из современных программных интерфейсов, который предоставляет высокопроизводительное, многофункциональное, переносимое промежуточное ПО для существующих параллельных моделей, используемых для высокопроизводительных вычислений, таких как MPI, GlobalArrays, SHMEM и другие.
2. Рассмотреть и предложить возможные решения для текущих недостающих функциональных возможностей исследуемого ПО, которые мешают обеспечивать низкие задержки, высокую пропускную способность, требуемую масштабируемость на известных сетевых технологиях.
3. Провести исследования, экспериментальные доказательства состоятельности и возможности дальнейшего применения предложенных решений на примере, широко используемого программного интерфейса программирования выскопроизводительных приложений, MPI.

Обзор предметной области

Заключение

Литература

Приложение