Работу выполнил:
Гайдай А. В.
Научный руководитель:
Курносов М. Г.

Методы организации отказоустойчивого выполнения параллельных программ

Докладчик: Гайдай А. В.

Эл. почта: diligent20494@gmail.com
Телефон: +7 (983)-309-84-33

Постановка задачи

- Разработка отказоустойчивой реализации рассматриваемого алгоритма
- Экспериментальное исследование разработанной реализации
- Оценка накладных расходов на организацию отказоустойчивости

СибГУТИ ВС

Актуальность

- Современные высокопроизводительные ВС являются большемасштабными и имеют в своём составе порядка $\mathbf{10^6 - 10^7}$ вычислительных ядер [4]
- Среднее время безотказной работы в таких системах составляет от нескольких **десятков минут** до **пары часов** [1]

Топ 3 ВС из списка www.top500.org

№	Вычислительная система	Кол-во выч. ядер
1	*Sunway TaihuLight*	10,649,600
2	*Tianhe*-2	3,120,000
3	*Titan - Cray XK7*	560,640

СибГУТИ ВС

Контрольные точки восстановления

- Некоторыми исследователями [1] отмечается, что в системах Терафлопсного и Петафлопсного уровней создание согласованной КТ занимает порядка **20-30 минут**

- Методы организации отказоустойчивости, основанные на использовании контрольных точек восстановления, в перспективе становятся малоэффективными [2]

Живучие алгоритмы

- **Живучий алгоритм** в любой момент времени использует все выделенные на решение конкретной задачи исправные ресурсы ВС
- При возникновении программного **отказа** живучий алгоритм адаптируется под новое количество исправных параллельных процессов
- Создание живучих алгоритмов – нетривиальная задача

СибГУТИ ВС

Message Passing Interface

- **Message Passing Interface** (**MPI**, интерфейс передачи сообщений) – это стандарт интерфейса библиотеки передачи сообщений [3]

- **MPI** ориентирован на использование в моделях параллельного программирования, в которых данные передаются из адресного пространства одного процесса в пространство другого по средствам совместных операций

- Две популярные реализации – **MPICH** и **OpenMPI**

СибГУТИ ВС

Параллельный алгоритм решения СЛАУ

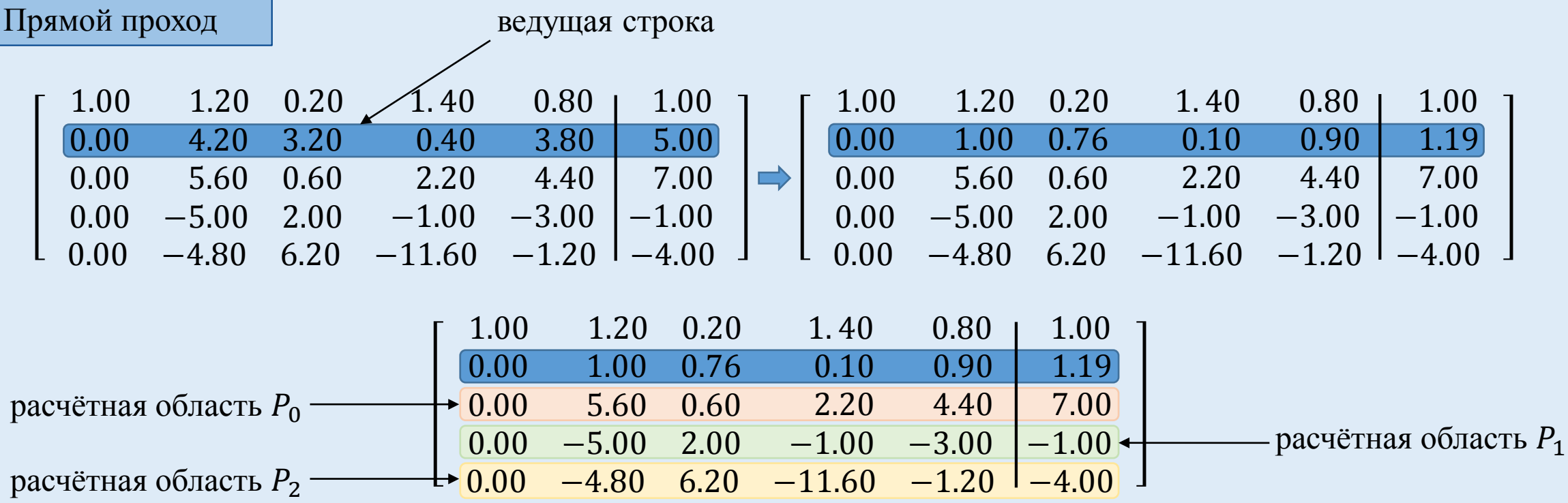
Прямой проход

СибГУТИ

ВС

Параллельный алгоритм решения СЛАУ

Прямой проход



Параллельный алгоритм решения СЛАУ

Обратный проход

- Информационная избыточность повышает надёжность алгоритма

$$\left[\begin{array}{ccccc|c} 1.00 & 1.20 & 0.20 & 1.40 & 0.80 & 1.00 \\ 0.00 & 1.00 & 0.76 & 0.10 & 0.90 & 1.19 \\ 0.00 & 0.00 & 1.00 & -0.45 & 0.18 & -0.09 \\ 0.00 & 0.00 & 0.00 & 1.00 & 0.22 & 2.59 \\ 0.00 & 0.00 & 0.00 & 0.00 & 1.00 & 7.04 \end{array}\right]$$

- Такой способ организации параллельных вычислений выбран для дальнейшего преобразования алгоритма в его живучий аналог

$$x_1 = 1.00 - 1.20x_2 - 0.20x_3 - 1.40x_4 - 0.80x_5$$
$$x_2 = 1.19 - 0.76x_3 - 0.10x_4 - 0.90x_5$$
$$x_3 = -0.09 + 0.45x_4 - 0.18x_5$$
$$x_4 = 2.59 - 0.22x_5$$
$$x_5 = 7.04$$

СибГУТИ ВС

User Level Failure Mitigation

- **User Level Failure Mitigation (ULFM)** – расширение стандарта **MPI**, предоставляющее средства, способствующие возобновлению процесса вычислений после возникновения отказа на этапе выполнения программы [5]

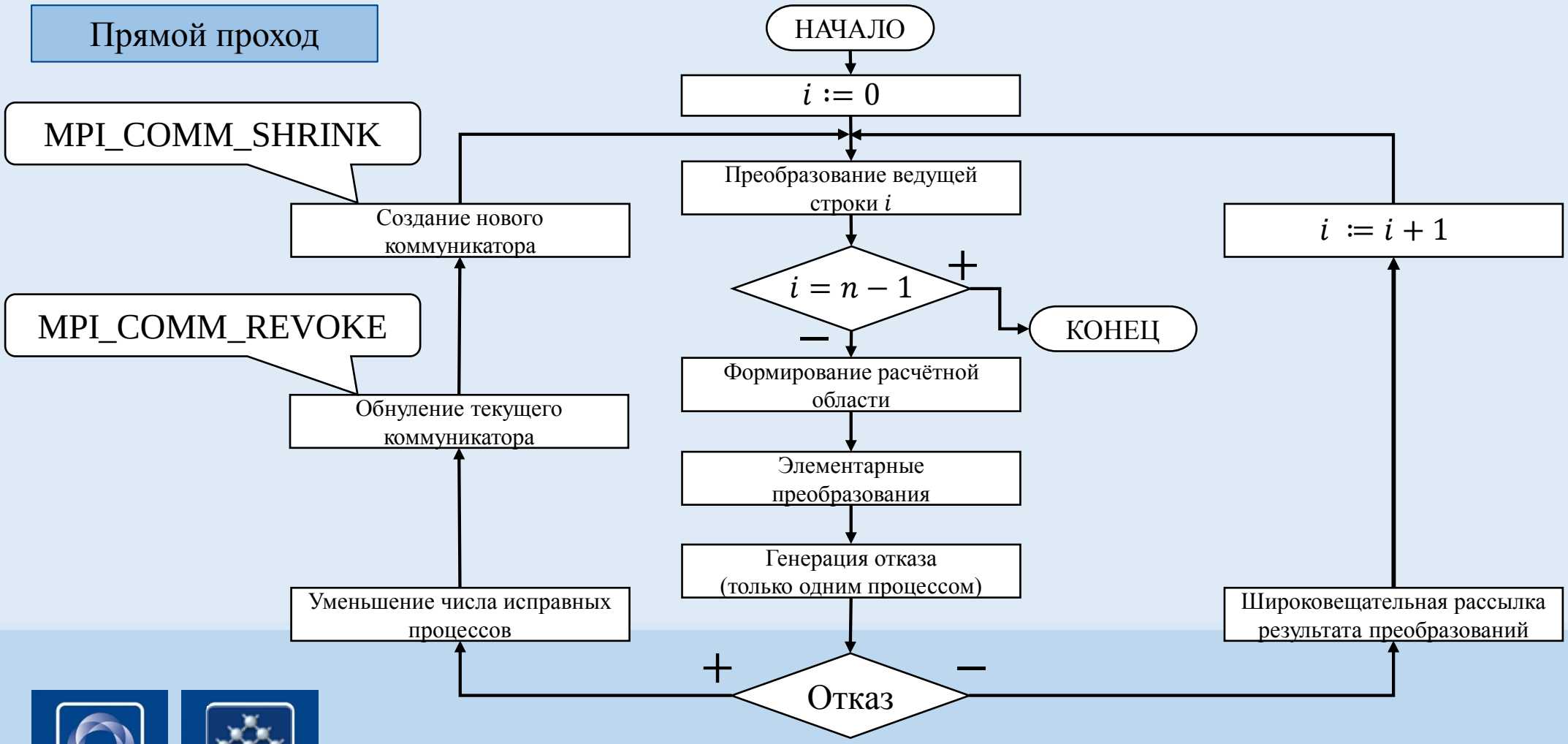
- **MPI_COMM_REVOKE(comm)** – прерывает любые операции на коммуникаторе **comm**

- **MPI_COMM_SHRINK(comm, newcomm)** – создаёт новый коммуникатор **newcomm**, в котором нет отказавших процессов коммуникатора **comm**

СибГУТИ ВС

Живучий алгоритм решения СЛАУ

Прямой проход



СибГУТИ

ВС

Живучий алгоритм решения СЛАУ

Обратный проход

- Информационная избыточность повышает надёжность алгоритма
- Все не отказавшие процессы содержат в памяти вектор значений главных переменных

$$\left[\begin{array}{ccccc|c} 1.00 & 1.20 & 0.20 & 1.40 & 0.80 & 1.00 \\ 0.00 & 1.00 & 0.76 & 0.10 & 0.90 & 1.19 \\ 0.00 & 0.00 & 1.00 & -0.45 & 0.18 & -0.09 \\ 0.00 & 0.00 & 0.00 & 1.00 & 0.22 & 2.59 \\ 0.00 & 0.00 & 0.00 & 0.00 & 1.00 & 7.04 \end{array}\right]$$

$$x_1 = 1.00 - 1.20x_2 - 0.20x_3 - 1.40x_4 - 0.80x_5$$
$$x_2 = 1.19 - 0.76x_3 - 0.10x_4 - 0.90x_5$$
$$x_3 = -0.09 + 0.45x_4 - 0.18x_5$$
$$x_4 = 2.59 - 0.22x_5$$
$$x_5 = 7.04$$

СибГУТИ ВС

Результаты экспериментов

- Эффективность разработанного алгоритма исследовалась на кластере Jet
- Кластер Jet укомплектован 18 вычислительными узлами, управляющим узлом, вычислительной и сервисной сетями связи, а также системой бесперебойного электропитания

Конфигурация вычислительного узла

Системная плата	Intel S5000VSA
Процессор	2 x Intel Xeon E5420
Оперативная память	8 GB
Жесткий диск	SATAII 500GB

Конфигурация управляющего узла

Системная плата	Intel S5000VSA
Процессор	2 x Intel Xeon E5420
Оперативная память	16 GB
Жесткий диск	3 x SATAII 500 GB

СибГУТИ ВС

Результаты экспериментов

СибГУТИ

ВС

Результаты экспериментов

СибГУТИ

ВС

Заключение

- Накладные расходы на организацию отказоустойчивости алгоритма решения СЛАУ методом Гаусса зависят от интенсивности потока отказов и размера входной системы

- Предложенная реализация алгоритма уступает в скорости версии с циклическим распределением строк по процессам, однако основная цель – организация отказоустойчивости, достигнута в полной мере

СибГУТИ ВС

Публикации

1. Гайдай А.В. Алгоритм оптимизации использования мьютексов по результатам предварительного профилирования // Материалы международной научной студенческой конференции (МНСК-2016), Новосибирск, 2016
2. Гайдай А.В. Адаптивный алгоритм операции захвата мьютекса // Российская научно-техническая конференция «инновации и научно-техническое творчество молодёжи», Новосибирск, 2016
3. Гайдай А.В. Оптимизация синхронизации параллельных программ для вычислительных систем с общей памятью // Материалы Двенадцатой Международной Азиатской школы-семинара «Проблемы оптимизации сложных систем», Новосибирск, 2016

СибГУТИ ВС

Работу выполнил:
Гайдай А. В.
Научный руководитель:
Курносов М. Г.

Спасибо за внимание!

Докладчик: Гайдай А. В.

Эл. почта: diligent20494@gmail.com
Телефон: +7 (983)-309-84-33

Источники

1. Cappello, F. Fault tolerance in petascale/exascale systems: Current knowledge, challenges and research opportunities / Cappello F. // International Journal of High Performance Computing Applications. — 2009. — Vol. 23, № 3. — P. 212–226.
2. Hsu, C.-H. A power-aware run-time system for high-performance computting / C.-H. Hsu, W.-C. Feng. // Proceedings of SC|05: The ACM/IEEE International Conference on High-Performance Computing, Networking, and Storage (Seattle, Washington USA November 12 – 18, 2005). — IEEE Press, 2005. — P. 1–9.
3. MPI 3.0 Documentation [Электронный ресурс]. – URL: http://mpi-forum.org/docs/mpi-3.0/mpi30-report.pdf.
4. Top-500 [Электронный ресурс]. – URL: https://www.top500.org/.
5. ULFM [Электронный ресурс]. – URL: http://fault-tolerance.org/category/ulfm/.

СибГУТИ ВС