3. Bogey C., de Cacqueray N., Bailly C., Self-adjusting shock-capturing spatial filtering for high-order non-linear computations // AIAA 2008–2968, 14th AIAA/CEAS Aeroacoustics Conference (29th AIAA Aeroacoustics Conference, 5–7 May 2008, Vancouver, British Columbia Canada.

4. http://www.energy.kth.se/proj/projects/Markus%20Joecker/STCF.

А.Н. Фирсов

Пермский государственный университет

ОБЕСПЕЧЕНИЕ ОТКАЗОУСТОЙЧИВОСТИ В ГЕТЕРОГЕННЫХ РАСПРЕДЕЛЕННЫХ СИСТЕМАХ

В настоящее время все острее встают вопросы обеспечения отказоустойчивости информационных систем (ИС). Этому есть несколько причин. Во-первых, системы с каждым годом усложняются, включают в себя все большее количество компонентов, а следовательно, вероятность того, что в одном из компонентов системы произойдет сбой, увеличивается. Это может привести к отказу всей системы, если она не спроектирована с учетом такой возможности. Во-вторых, информационные системы проникают во все области жизнедеятельности, люди все больше полагаются на ИС, все больше зависят от них, поэтому последствия отказа ИС становятся все более и более критичными, если не катастрофическими.

Византийская модель сбоев [1] (или модель произвольных сбоев) является наиболее общей, включающей в себя все остальные, моделью. Она не накладывает никаких ограничений на действия сбойных процессоров и, благодаря этому, позволяет моделировать такие типы сбоев, как отказы оборудования, потеря связи, неисправность оборудования, программные сбои, ошибки операторов и действия злоумышленников.

Особый интерес представляют асинхронные распределенные системы, так как они не накладывают никаких временных ограничений и благодаря этому идеально подходят для работы в сети Интер-

нет. Наиболее универсальной моделью вычислительного узла является машина с конечным числом состояний (state machine).

В данной работе рассматриваются асинхронные распределенные системы, которые, во-первых, устойчивы к произвольным отказам, во-вторых, позволяют выполнять прикладные программы, требующие сохранять свое состояние между обращениями к ним, в-третьих, обеспечивают отказоустойчивость на программном уровне без необходимости в какой-либо специальной аппаратуре и, в-четвертых, делают все это прозрачно для прикладных программ.

В настоящее время существует несколько таких систем: Rampart [2], SecureRing [3], BFT [4] и т.д. Все они реализуют единственный известный на сегодня программный метод защиты от произвольных сбоев – активную репликацию. Суть этого метода заключается в том, что параллельно выполняются нескольких копий (или, как их еще называют, реплик) одной и той же программы. Отличие между этими системами в основном заключается в стеке протоколов, который обеспечивает согласованную работу всех реплик.

Существующие системы обладают несколькими недостатками, корнем которых является то, что эти системы плохо приспособлены для работы в гетерогенных сетях. Под гетерогенностью сети здесь понимается различная надежность включенных в эту сеть компьютеров, различная скорость и задержка при обмене информацией между компьютерами, различная производительность. Именно такой сетью является Интернет, для работы в котором и создавались приведенные выше отказоустойчивые системы, однако то, что они не учитывают неоднородность, приводит к неэффективной работе этих систем:

- Предоставляемый ими стек протоколов фиксирован, а как показано в работе [6,7], в разных ситуациях эффективным оказывается применение разных протоколов.

- Не предоставляется алгоритм для определения оптимального значения параметров работы системы (частота создания контрольных точек, время ожидания у ненадежных детекторов сбоев и т.д.). А следует заметить, что значения этих параметров существенно влияют на эффективность работы системы [6, 7].

• Не дается оценка надежности получаемого решения. А именно в этом показателе наиболее заинтересован пользователь отказоустойчивых систем, так как абсолютная надежность недостижима.

• Не определяется вклад каждого процессора в общий вычислительный процесс. Создатели приведенных выше систем такую задачу перед собой и не ставили, но, как продемонстрировано платформой BOINC [5], это может оказаться очень полезным.

Подход к обеспечению отказоустойчивости в гетерогенных системах. Корнем всех проблем, описанных выше, является то, что, существующие системы обеспечения отказоустойчивости не получают никакой информации о среде, в которой они работают. Под средой выполнения здесь подразумевается следующее:

1. Сеть, на основе которой работает отказоустойчивая система. Информация о сети включает:

а) производительность вычислительных узлов;

б) вероятности отказов вычислительных узлов;

в) задержки при передаче сообщений между вычислительными узлами;

г) скорость передачи информации по каналам связи.

2. Прикладная программа, бесперебойное выполнение которой требуется обеспечить. О ней необходима следующая информация:

а) время, требующееся на создание контрольной точки;

б) частота групповых рассылок.

3. Относительная ценность ресурсов. Работа алгоритма обеспечения отказоустойчивости требует ресурсов. Можно выделить три из них, которые являющиеся наиболее важными:T_o – общее время работы программы, T_p – количество единиц процессорного времени, потраченного на выполнение программы всеми процессорами, V – количество переданных сообщений. Не существует алгоритма обеспечения отказоустойчивости, который является предпочтительным по всем параметрам. Как правило, каждый из протоколов позволяет оптимизировать один параметр за счет другого, например, сэкономить процессорное время за счет передачи большего числа сообщений. Поэтому вопрос об использовании того или иного протокола зависит от степени важности

того или иного ресурса. А эту степень важности может определить только пользователь или система балансировки.

В общем виде относительная степень важности ресурсов задается вектором $(r_1, r_2, ..., r_N)$, компоненты которого должны удовлетворять двум условиям:

1) $r_i \geq 0, i = \overline{1, N}$;

2) $\sum_{i=1}^{N} r_i > 0$.

Имея данные о сети и прикладной программе, можно оценить, сколько ресурсов каждого вида будет потреблять тот или иной алгоритм обеспечения отказоустойчивости в зависимости от параметров его функционирования [7]. Общую же ресурсоемкость R алгоритма A можно вычислить по формуле

$$R^A = \sum_i r_i \cdot R_i^A,$$

где R_i^A – количество потребленных единиц ресурса i при использовании алгоритма обеспечения отказоустойчивости A ($T_o{}^A \equiv R_1^A$, $T_p{}^A \equiv R_2^A$, ...).

При имеющейся оценке ресурсоемкости протоколов обеспечения отказоустойчивости вопрос о поиске оптимальных значений параметров этих протоколов сводится к задаче многомерной глобальной оптимизация функции $R^* = \min_X(R(X))$, где X – вектор параметров некоторого протокола.

Выбор же оптимального алгоритма обеспечения отказоустойчивости сводится к оценке ресурсоемкости всех имеющихся алгоритмов с оптимальными параметрами и выбору того, который обеспечивает минимальную ресурсоемкость.

Кроме как для определения наилучшего алгоритма обеспечения отказоустойчивости и его оптимальных параметров для каждой конкретной ситуации, данные о среде исполнения могут быть использованы в алгоритмах обеспечения отказоустойчивости напрямую. К таким алгоритмам относятся асинхронный протокол надежной групповой рассылки с переменным числом

активных реплик[8] и алгоритм группировки вычислительных узлов для повышения надежности [9].

3. Реализация и использование. С целью практического применения идей, методов и алгоритмов, изложенных выше, была реализована распределенная система SFT (Statistical Fault-Tolerance), обеспечивающая устойчивость к произвольным отказам. Более подробное описание архитектуры и некоторых деталей реализации приведено в [10].

При использовании системы SFT принципиально важной является среда, в которой она работает. В зависимости от среды выполнения преимущества от использования системы SFT по сравнению с другими системами могут быть как бесконечно большие (в теории), так их может и не быть вовсе.

В общем случае действует правило: чем больше неоднородность среды выполнения, тем больше получаемый выигрыш. Если же среда гомогенна, то выигрыша в ресурсоемкости не будет, а накладные расходы на сбор информации о среде выполнения могут привести и к отрицательным результатам. Отсюда следует вывод, что систему SFT следует использовать в гетерогенных сетях.

Список литературы

1. Lamport L., Shostak R. and Pease M. The Byzantine Generals Problem. // ACM Transactions on Programming Languages and Systems –July 1982. –Vol. 4, No. 3. – P. 382–401.

2. Reiter M. The Rampart Toolkit for Building High-Integrity Services // Selected Papers from the International Workshop on Theory and Practice in Distributed Systems –Springer-Verlag, 1994. –P. 99–110

3. Kihlstrom K. P., Moser, L. E. and Mellar-Smith, P. M. The SecureRing group communication system // ACM Transactions on Information and System Security (TISSEC), –November 2001. –Vol. 4, No. 4. –P. 371–406.

4. Castro M. Practical Byzantine Fault Tolerance // Massachusetts Institute of Technology, 2001. PhD thesis. -172 p.

5. Anderson, D.P., Christensen, C. and Allen, B. Designing a runtime system for volunteer computing // Proceedings of the 2006 ACM/IEEE conference on Supercomputing –2006. –Article No. 126

6. Фирсов А.Н. Оптимизация на основе статистических данных асинхронной распределенной системы, устойчивой к произвольным отказам // Параллельные вычислительные технологии (ПаВТ-2009): тр. междунар. науч. конф. (Нижний Новгород, 30 марта – 3 апреля 2009 г.). – Челябинск: Изд-во ЮУрГУ, 2009. – С. 765–771.

7. Фирсов А.Н. Оценка эффективности некоторых оптимизаций протоколов надежной и атомарной групповой рассылки // Вестн. Перм. гос. ун-та. Математика. Механика. Информатика. 2009. – С. 161–168.

8. Фирсов А.Н. Асинхронный протокол надежной групповой рассылки с переменным числом активных реплик // Математика программных систем: межвуз. сб. науч. тр. / Перм. ун-т. – Пермь, 2008. – С. 166– 177.

9. Фирсов А.Н. Повышение надежности гетерогенных распределенных отказоустойчивых систем за счет группировки вычислительных узлов. Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: тр. междунар. суперкомпьютерной конф. (20–25 сентября 2010г., г. Новороссийск). – М.: Изд-во МГУ, 2010. – С. 628–634.

10. Фирсов А.Н. Архитектура распределенной системы устойчивой к произвольным отказам // Высокопроизводительные параллельные вычисления на кластерных системах: материалы IX Междунар. конф.-семинара. – Владимир : Изд-во Владимир. гос. ун-та, 2009. – С. 396–400.