ОБ АГЕНТНО-ОРИЕНТИРОВАННОМ ПОДХОДЕ К ИМИТАЦИОННОМУ МОДЕЛИРОВАНИЮ СУПЕРЭВМ ЭКЗАФЛОПСНОЙ ПРОИЗВОДИТЕЛЬНОСТИ

Б.М. Глинский, А.С. Родионов, М.А. Марченко

Введение

Создание суперЭВМ экзафлопсной производительности потребует решения целого ряда технических проблем, связанных с энергопотреблением, памятью большого объема, температурных проблем, обусловленных функционированием от 10 до 100 млн. вычислительных ядер. Другая проблема будет связана с разработкой и применением системного программного обеспечения, ориентированного на миллионы процессорных ядер с гибридной архитектурой. Следующая проблема связана с надежностью и оценкой реальной производительности вычислительной системы с учетом возможных отказов отдельных элементов системы имеющей десятки и сотни миллионов вычислительных ядер. Потребуются алгоритмы, учитывающие стохастические свойства большого числа вычислительных процессов.

Пути решения вышеперечисленных проблем ищет группа экспертов в рамках проекта IESP (International Exascale Software Project), одним из инициаторов которого является Джек Донгарра. По мнению экспертов, системы экзафлопсного уровня потребуют переработки ОС, среды исполнения, компиляторов, библиотек, сред программирования, принципов отказоустойчивости. Если сбои возникают, то выполняемая задача должна уметь их распознавать и уметь восстанавливаться без вмешательства оператора.

Реальные экзафлопсные компьютеры по прогнозам экспертов появятся в районе 2018 года и эксперименты с реальными компьютерами такой производительности в настоящее время невозможны, однако уже сейчас некоторые из вышеперечисленных проблем можно попытаться решить методами имитационного моделирования, которое представляет собой мощное средство анализа сложных систем.

В данной работе рассматривается возможность применения агентно-ориентированной системы имитационного моделирования для решения некоторых проблем, возникающих при создании экзафлопсных компьютеров, содержащих большое количество вычислительных ядер: отработки алгоритмов управления и оптимизации вычислениями в системах экзафлопсной производительности; прогнозирование и предотвращение сбоев и отказов вычислительных ядер.

Общие принципы организации процесса моделирования

При решении поставленных задач имитационное моделирование может применяться по трём сценариям:

- для анализа предлагаемых алгоритмов управления с использованием модели суперЭВМ на ЭВМ меньшей производительности в режиме чистого моделирования;
- для оперативного выбора наилучших решений по управлению счётом задач с использованием эмулятора суперЭВМ на ЭВМ меньшей производительности в режиме реального времени;
- для оперативного выбора наилучших решений по управлению счётом задач на реальной суперЭВМ в режиме реального времени.

Последние два сценария предусматривают реализацию определённой системы принятия решений, т.е. требуют взаимодействия имитационной модели с некоторой системой искусственного интеллекта. Как известно, существует четыре классических способа такого взаимодействия [1,2].

Наиболее очевидным представляется способ "включения" или E-способ (от "Embedding") – в этом случае или экспертная система включается в имитационную модель или, наоборот, имитационная модель входит в экспертную систему. Во втором случае имитационная модель и экспертная система создаются независимо (как самостоятельные программные системы), а для их связи – программа, передающая при необходимости рекомендации от экспертной системы в модель и обратно, результаты моделирования в качестве новых знаний в экспертную систему. Такой способ связи называется P-способом (от "Parallel"). Если экспертная система и имитационная модель предназначены для решения одной задачи и имеют общие данные, то это называется C-способом объединения (от "Cooperation"). Наконец, существует способ, при котором экспертная система находится между пользователем и системой моделирования. В этом случае экспертная система в диалоге с пользователем строит имитационную модель, исполняет ее и интерпретирует результаты для пользователя. Этот способ называется IFE-способом (от "Intelligent Front End").

В последние годы существенное развитие получило новое направление в создании интеллектуальных программных систем, связанное с так называемыми интеллектуальными программными агентами (далее для краткости просто программные агенты) [3]. Технология программных агентов оказалась удобной и для создания систем имитационного моделирования [4]. При этом получилось новое качество взаимодействия искусственного интеллекта и имитационного моделирования, а именно их полное слияние: сами моделирующие агенты наряду с изменением переменных состояния принимают решение о выборе дальнейшего поведения

системы.

Исходя из этого свойства агентно-ориентированного подхода и учитывая признанный факт что “для моделирования распределённых систем наилучшим образом подходит распределённое моделирование, основанное на передаче сообщений” [5], для реализации имитационных моделей был выбран агентный подход.

Агенты функционируют асинхронно по своим законам, взаимодействуя с другими агентами для достижения общих целей. В процессе фукционирования программный агент может изменять как внешнюю среду, так и свое поведение [3],[6]. Распределённый, асинхронный характер поведения *чрезвычайно важен при построении имитационной модели экзафлопсного компьютера, поскольку обеспечить централизованное управление десятками и сотнями миллионов ядер практически невозможно*. Выбору агентного подхода к построению моделей поспособствовал и тот факт, что с точки зрения управляющей агентами системы реальные управляющие агенты и агенты-имитаторы неразличимы, т.е. возможен постепенный переход от чистой модели к полунатурной и далее к реальной системе управления вычислениями.

Особо интересен в теоретическом плане вопрос синхронизации при взаимодействии агентов при реальном моделировании экзафлопсных систем: совершенно очевидно, что классические подходы к синхронизации событий в распределённом моделировании [7] не могут справиться с задачей тотальной синхронизации миллионов и миллиардов событий, планируемых на коротких интервалах времени. При этом локальные имитационные модели, используемые для принятия локальных же решений в реальном масштабе времени, могут выполняться классическим образом с использованием временных календарей событий. Моделирование же экзафлопсных систем в целом, как и реальное управление ими, требует разработки принципиально новых подходов, скорее всего допускающих наличие определённой погрешности в управлении событиями.

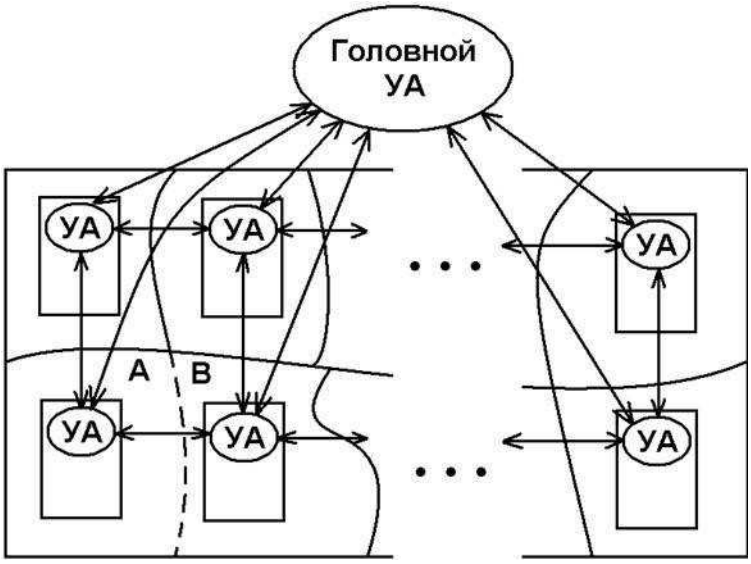

Рис. 1 Двухуровневая схема управления моделью вычислений

Предлагается иерархическая система распределённого управления моделью, переносимая на управление реальной экзафлопсной ВС (двухуровневый вариант представлен на рис. 1). Все вычислительные узлы поделены между областями вычислений, управляемыми своими управляющими агентами (УА). Головной УА (ГУА) распределяет между областями поток больших задач и управляет общими ресурсами. Локальные УА (ЛУА) управляют потоками мелких задач, поступающих непосредственно в управляемые области вычислений, и принимают на счёт большие задачи от ГУА. ЛУА взаимодействуют с соседними ЛУА, сообщая о состоянии используемых ресурсов и, возможно, запрашивая недостающие ресурсы. При отдаче ресурсов соседям ЛУА сохраняет за собой область заданного объема, не допуская их монополизации. Несколько областей могут объединяться для решения одной задачи по требованию ГУА (области A и B на рисунке), при этом по завершению вычислений все соответствующие ЛУА снова получают право управления своими областями. В процессе функционирования ЛУА периодически (а также по требованию) передаёт ГУА информацию о состоянии управляемой области (занятые и свободные ресурсы, перспективы освобождения ресурсов, занятость в совместных с соседними областями вычислениями и т.п.).

Для принятия решений по управлению УА любых уровней могут исполнять вспомогательные имитационные модели со своими собственными локальными календарями событий. При моделировании взаимодействия между областями вычислений может требоваться синхронизация модельного времени с откатом по времени одной или нескольких (зависит от масштаба взаимодействий) из соответствующих подмоделей. Для минимизации числа откатов ГУА периодически запускает программу синхронизации локальных календарей. Запуск этой программы возможен и в экстренном порядке при превышении порога числа откатов за заданный период времени. В реальной системе это соответствует приостановлению потоков новых задач, досчитыванию

запущенных и перезапуску основного процесса управления.

Реализация функциональных агентов моделирующей системы естественным образом зависит от конфигурации моделируемой вычислительной системы, однако общий набор этих агентов достаточно стабилен и с необходимостью включает агенты управления памятью, агенты, соответствующие вычислительным узлам и ядрам и агенты управления внешними устройствами. ЛУА и ГУА, каждый на своём уровне, для реализации функций управления моделью взаимодействуют со вспомогательными агентами среды моделирования, обеспечивающими синхронизацию событий, сбор и хранение данных по поведению модели и средства восстановления состояния модели при сбоях.

Пример организации конкретной модели рассмотрен ниже.

Имитация загрузки экзафлопсной суперЭВМ расчетными задачами

В рамках разрабатываемого агентно-ориентированного подхода предлагается имитировать загрузку вычислительной системы решением тестовых задач с использованием перспективных численных методов.

В настоящее время ведущими специалистами по вычислительной математике высказывается убеждение в том, что в ближайшем будущем в области компьютерного моделирования будут широко применяться вероятностные имитационные модели и методы Монте-Карло (методы численного статистического моделирования) [8]. С одной стороны, это убеждение основано на том, что вероятностные имитационные модели дают адекватное описание физических, химических, биологических и др. явлений при их рассмотрении «из первых принципов». С другой стороны, алгоритмы метода Монте-Карло, реализующие вероятностные модели, допускают возможность эффективного распараллеливания в виде распределенного статистического моделирования. Можно поэтому надеяться, что внедрение экзафлопсных суперЭВМ придаст существенный импульс использованию статистического моделирования в качестве одного из основных инструментов компьютерной имитации.

Под численным статистическим моделированием обычно понимают реализацию с помощью компьютера вероятностной модели некоторого объекта с целью оценивания изучаемых интегральных характеристик на основе закона больших чисел. При этом чем больше объем выборки, составленной из смоделированных независимых реализаций, тем выше точность оценивания, причем статистическая погрешность убывает обратно пропорционально квадратному корню от числа реализаций. Основным инструментом для моделирования сложных вероятностных распределений на компьютере является подходящий генератор базовых случайных чисел, дающий выборочные значения случайной величины, равномерно распределенной в интервале от 0 до 1 [8].

Предлагаемый агентно-ориентированный подход к построению имитационной модели экзафлопсного компьютера, не требующий глобального централизованного управления, хорошо соответствует методике распределенного статистического моделирования [9,10]. При распараллеливании статистического моделирования вычисление независимых реализаций случайной величины распределяется по свободным вычислительным ядрам. Объем памяти, доступной каждому вычислительному ядру, и его быстродействие должны быть достаточными для эффективного моделирования реализаций. Такой алгоритм распределенного статистического моделирования, при правильно выбранном параллельном генераторе базовых случайных чисел, масштабируется практически на неограниченное число ядер [9].

Количество ядер, участвующих в расчетах, влияет на совокупный объем выборки из смоделированных независимых реализаций. В случае отказа какого-либо вычислительного ядра, участвующего в статистическом моделировании, пропавшие данные могут быть скомпенсированы расчетами на других ядрах. Таким образом, распределённое статистическое моделирование представляет собой расчетный метод, хорошо защищённый от отказов вычислительных узлов системы.

Целесообразно осуществлять периодическую пересылку результатов промежуточного осреднения реализаций, независимо полученных на загруженных ядрах, на выделенные ядра, объединенные в иерархическую структуру. Выделенные ядра будут периодически получать переданные им данные и осреднять их, передавая затем результаты на ядро, соответствующее вершине иерархической структуры (см. рис. 2). Рассчитанные на последнем ядре осредненные значения будут соответствовать выборке, полученной совокупно на всех ядрах. Отметим, что при распределении нагрузки расположение загружаемых ядер в топологии вычислительной системы не играет большой роли, поскольку обмен данными между ядрами по необходимости осуществляется достаточно редко.

Распределенное статистическое моделирование на разных вычислительных ядрах производится в асинхронном режиме. Отправка и получение результатов статистического моделирования также осуществляется в асинхронном режиме. К расчетам возможно подключать свободные вычислительные ядра по мере их освобождения от других задач и эффективно учитывать получаемые на них результаты.

Остановка распределенного статистического моделирования на всех загруженных ядрах (или на части загруженных ядер) возможна в любой момент времени и его возобновление можно начать в удобное время, причем данные расчетов, полученные до остановки, будут эффективно учтены при возобновлении счета. Повторный расчет приведет к получению реализаций случайной величины, независимых в статистическом

смысле от реализаций, полученных до остановки.

Предлагаемый способ распределенного статистического моделирования требует правильного выбора базового генератора случайных чисел. Такой генератор должен обладать а) «астрономически» большим периодом, б) возможностью независимого получения потоков базовых случайных чисел на процессорах. Кроме того, полученные таким образом случайные числа в совокупности должны удовлетворять чрезвычайно строгим статистическим тестам на случайность и многомерную равномерность, а также должны быть проверены путем решения сложных тестовых задач [8, 9]. Разработка такого генератора и обоснование возможности его использования для распределенного статистического моделирования на экзафлопсных суперЭВМ представляет собой фундаментальную проблему, которую необходимо решать.

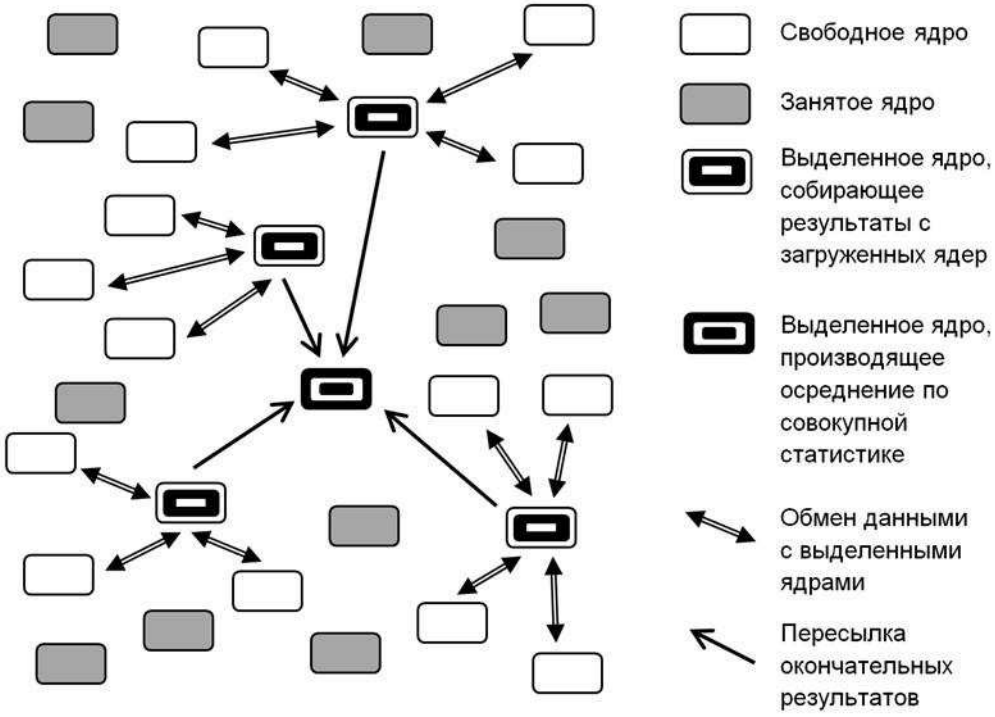

Рис.2. Схема, поясняющая передачу нагрузки на свободные вычислительные ядра и обмен данными с выделенными ядрами при распределенном статистическом моделировании.

В связи с вышесказанным, требуется разработка и имитационное моделирование системы управления заданиями, позволяющей в режиме реального времени:

а) осуществлять мониторинг свободных ресурсов и подключать их к расчетам;
б) назначать выделенные ядра с целью получения и осреднения данных с загруженных ядер, производящих независимое статистическое моделирование;
в) осуществлять передачу начальных данных на загружаемые вычислительные ядра и выдавать задания на начало счета;
г) планировать и контролировать передачу данных в иерархической структуре выделенных ядер с целью получения окончательного результата осреднения.

Моделирование загрузки экзафлопсной супер-ЭВМ можно осуществлять на основе отображения на ее архитектуру существующих параллельных алгоритмов, использующих распределенное статистическое моделирование. Параметры загрузки системы (объем данных, загрузка процессоров, использование коммуникационной сети) целесообразно оценивать на основе реальных расчетов. С этой целью может быть использована статистика о загрузке процессоров и коммуникационной сети при решении различных задач на кластерах ЦКП ССКЦ ИВМиМГ СО РАН, в частности, при использовании библиотеки PARMONC, предназначенной для распределенного статистического моделирования приложений метода Монте-Карло, обладающих большой вычислительной трудоемкостью [10].

При решении масштабных задач существенное влияние могут оказать возможные отказы вычислительных узлов. Следующий раздел посвящен подходам к моделированию таких ситуаций.

Динамическая система предотвращения сбоев на основе мультиагентного моделирования

Рассмотрим вариант реализации мультиагентного моделирования для предотвращения сбоев и отказов вычислительных узлов. Динамическая система предотвращения сбоев состоит из агентов различного назначения: агента датчика вычислительного узла; агента датчика имитационной модели; агента анализа; агента распределения; агента загрузки/отключения; экспертной системы, содержащей свод правил поведения системы

для предотвращения сбоев вычислительных узлов.

Каждый агент выполняет свою функцию для достижения цели, а совместно реализуют динамическую систему предотвращения сбоев и отказов вычислительных узлов. Архитектура предлагаемой системы представлена на рис. 3.

Рассмотрим назначение основных узлов системы и принцип ее работы относительно N-го вычислительного узла.

Агент-датчик вычислительного узла (ВУ) осуществляет сбор информации о температурном режиме, нагрузке и состоянии линий связи между узлами и другой информации, характеризующей состояние узла.

Агент-датчик имитационной модели (ИМ) осуществляет мониторинг объектов имитационной модели во время имитационного прогона, моделируя температурный тренд узла, вычислительную нагрузку на узел, интенсивность обмена между узлами и другие параметры. В процессе работы он может учитывать аналогичные данные по другим вычислительным узлам.

Агент-анализа взаимодействует с агентами – датчиками вычислительного узла и имитационной модели и принимает решение об исправности данного вычислительного узла, возможности отключения или перераспределения нагрузки этого узла, на другие вычислительные узлы, руководствуясь правилами из экспертной системы. В его функцию также входит выдача информации для агента-распределителя о возможности включения данного узла.

Агент-распределения получает информацию от датчика анализа, и если принято решение о снижении вычислительной нагрузки на данный узел или его отключении, то он выясняет на каком ближайшем узле нагрузка минимальна и какую часть нагрузки можно передать на другой узел. Этот агент взаимодействует с другими агентами распределения имитационной модели и также действует по правилам экспертной системы. В функцию данного агента также входит возможность включения вычислительного узла при нормализации контролируемых параметров и извещение ближайших агентов-распределителей о готовности вычислительного узла принять дополнительную нагрузку.

Агент-загрузки/отключения осуществляет перенос нагрузки на выбранный узел. В случае отключения узла информация об этом сохраняется в агенте-распределения и она учитывается другими агентами-распределения.

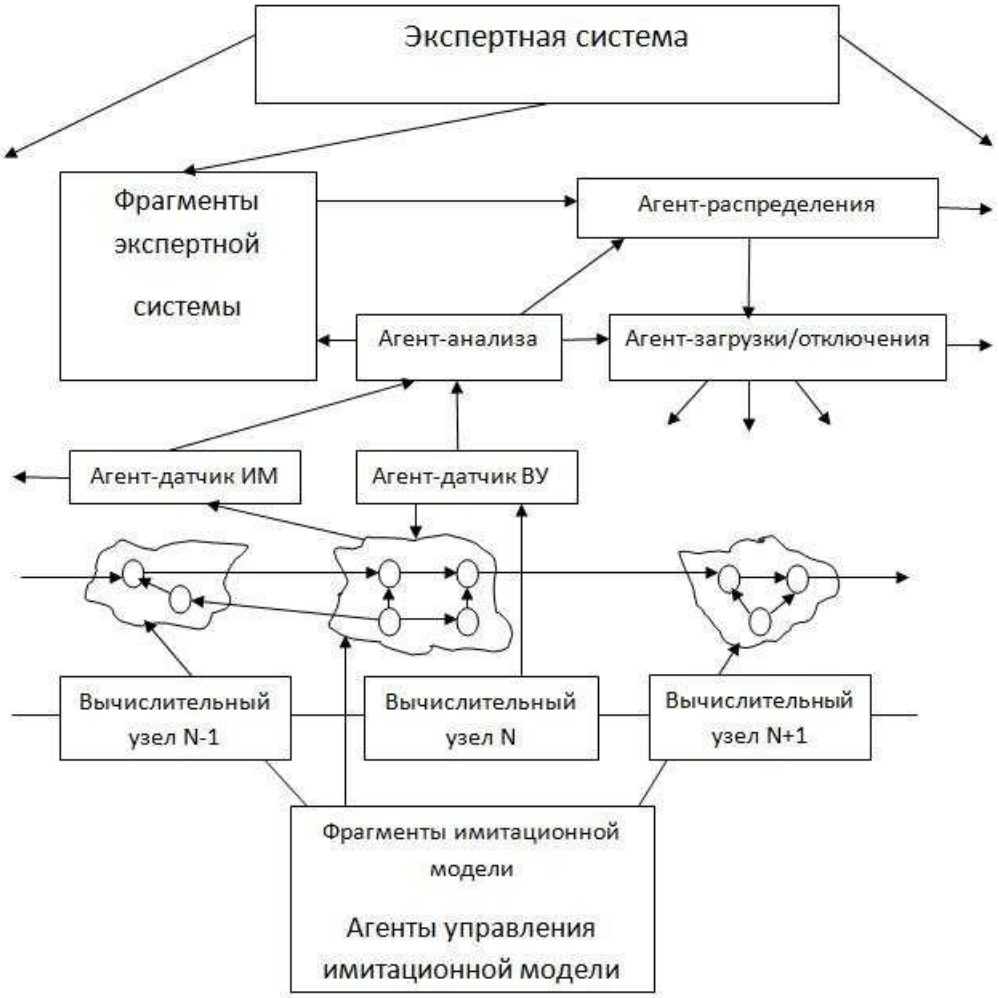

Рис 3. Агентно-ориентированная система предотвращения сбоев и отказов вычислительных узлов

Рис 3. Агентно-ориентированная система предотвращения сбоев и отказов вычислительных узлов

Вычислительный узел экзафлопсного компьютера вероятнее всего будет гибридным, состоящим из многоядерного CPU-процессора и GPU-процессора, либо другого ускорителя. Описание модели функционирования этого узла представляет самостоятельную задачу и в данной работе затрагиваться не будет.

Одной из ключевых проблем при реализации данной системы является разработка правил, которые будут заложены в экспертной системе для оценки текущей ситуации в каждом вычислительном узле. Свод правил должен, например, учитывать температурный тренд вычислительного узла, соответственно задавать температурный диапазон, при котором возможны отказы, и выдавать рекомендацию на разгрузку этого узла, либо отключение при достижении критического уровня. Аналогичные правила должны быть заложены в экспертной системе и для других параметров вычислительного узла, регистрируемых агентом-датчиком ВУ.

Итак, система должна функционировать следующим образом: для каждой поступающей из очереди задачи инициируется система имитационного моделирования (при этом часть ресурса вычислительной системы будет отдано для моделирования); запускается задача и одновременно имитационная модель, функционирование которой было описано выше.

В качестве примера применения в данной системе моделирования агента-датчика рассмотрим возможности системы мониторинга IDC (Input Data Center) [11]. Данный модуль выполняет функцию сбора информации со всего кластера, построенного на серверах HP ProLiant BL460c. В частности, контролируются следующие параметры:

1. **температура**, с помощью датчиков APC, расположенных внутри шкафа и сенсоров в лезвиях, доступ к которым осуществляется с использованием интерфейса «iLO» (Integrated Lights-Out). Отметим, что полученная информация позволяет контролировать данный параметр внутри каждого узла.
2. **загрузка узла**, данные, доступные в логах CMU (Cluster Management Utility) и характеризуют загрузку процессоров и оперативной памяти узлов. А также, что немаловажно для агент-анализа, общее время работы системы после последней операции отключения или

перезагрузки.

Таким образом, реализация подобной системы предотвращения сбоев хотя и потребует определенного дополнительного ресурса, однако существенно повысит надежность функционирования экзафлопсного компьютера.

Заключение

В работе предложена концепция организации агентно-ориентированной системы имитационного моделирования экзафлопсных суперЭВМ. Система предназначена для решения ряда проблем, возникающих при создании компьютеров, содержащих большое количество вычислительных ядер, в частности, для прогнозирования возможных сбоев и отказов вычислительных ядер и отыскания способов их предотвращения; для отработки алгоритмов управления вычислениями и исследования способов их оптимизации.

Агентно-ориентированный подход к моделированию систем экзафлопсной производительности был выбран потому, что для имитации поведения распределённых систем наилучшим образом подходит именно распределённое моделирование, основанное на передаче сообщений. Распределённый, асинхронный характер поведения вычислительной системы чрезвычайно важен при построении имитационной модели экзафлопсного компьютера, поскольку обеспечить централизованное управление десятками и сотнями миллионов ядер практически невозможно.

В работе рассматривается методология моделирования загрузки экзафлопсной суперЭВМ на основе отображения на ее архитектуру задач распределенного статистического моделирования. Сформулированы базовые принципы построения соответствующей системы управления заданиями. При таком моделировании параметры загрузки вычислительной системы (объем данных, загрузка вычислительных ядер, использование коммуникационной сети) могут оцениваться на основе реальных расчетов, проводимых на кластерах ЦКП ССКЦ при ИВМиМГ СО РАН.

В работе рассмотрены принципы создания системы мультиагентного моделирования для предотвращения сбоев и отказов вычислительных узлов. Такая система состоит из агентов различного назначения и экспертной системы, содержащей свод правил поведения вычислительной системы. Одной из ключевых проблем является разработка правил, которые будут заложены в экспертной системе для оценки текущей ситуации в каждом вычислительном узле. Например, свод правил должен учитывать температурный тренд вычислительного узла и соответственно задавать температурный диапазон, при котором возможны отказы; выдавать рекомендацию либо на разгрузку данного узла, либо на его отключение при достижении критического температурного уровня.

В дальнейшем рассматриваемый агентно-ориентированный подход предлагается применять при имитации загрузки экзафлопсной суперЭВМ на основе использования других перспективных параллельных численных методов.

ЛИТЕРАТУРА:

1. O'Keefe R. Simulation and expert systems – a taxonomy and some examples // Simulation. – 1986. – Vol. 46, № 1. – P. 10–15.
2. А.С. Родионов Интеллектуальное моделиро-вание – новое направление в системах имитации (обзор последних публикаций) // Экспертные системы и базы данных, - Новосибирск: ВЦ СО АН СССР, 1988. С.19-35.
3. M. Wooldridge «Introduction to MultiAgent Systems» // England: JOHN WILEY & SONS, LTD, 2002.
4. T. Oren, L. Yilmaz. On the Synergy of Simulation and Agents: An Innovation Paradigm Perspective// International Journal of Intelligent Control and Systems. Vol. 14, No. 1, March 2009, P. 4-19.
5. R.L. Bargodia, K.M. Chandy, J. Misra «A Message-based approach to discrete-event simulation» // IEEE Trans. on Soft. Eng. - 1987. - Vol. 13, № 6. - P. 654--665.
6. Ю.Г. Карпов «Имитационное моделирование систем. Введение в моделирование на Any-Logic 5» // БХВ_Петербург, С.Петербург, 2005.
7. E. Niewiadomska-Szynkiewicz, A. Sikora «Algorithms for Distributed Simulation - Comparative Study» // PARELEC 2002: Warsaw, Poland, P. 261-266.
8. М.А. Марченко, Г.А. Михайлов «Распределенные вычисления по методу Монте-Карло» // Автоматика и телемеханика, 2007, Вып. 5, - С.157–170.
9. М.А. Марченко «PARMONC - библиотека программ для распределенных вычислений по методу Монте-Карло» [Электронный ресурс] - Сайт ССКЦ КП СО РАН: http://www2.sscc.ru/SORAN-INTEL/paper/2011/parmonc.htm
10. Г.А. Михайлов, А.В. Войтишек «Численное статистическое моделирование. Методы Монте-Карло» – М.: Издательский центр «Академия», 2006.
11. Д.В. Гордиенко «Разработка и моделирование системы управления энергопотреблением кластерных вычислительных систем» // Труды межд. конф. «Высокопроизводительные параллельные вычисления на кластерных системах» (ВПВКС 2009).- Владимир 2009.