

БАЛАНСИРОВКА НАГРУЗКИ В МУЛЬТИАГЕНТНЫХ СИСТЕМАХ НА ОСНОВЕ КРЕДИТА

Жудин Никита Андреевич

студент

Приходько Татьяна Александровна

доцент кафедры вычислительных технологий

Кубанский государственный университет, г. Краснодар

Аннотация. В данной статье рассматривается применение динамической подсистемы балансировки нагрузки на основе кредита. Из-за уникальных особенностей программного агента дисбаланс нагрузки неизбежен со временем. Таким образом, балансировка нагрузки необходима в качестве критически важной системной операции. Миграция агентов является одной из стратегий, принятых для поддержки балансировки нагрузки в мультиагентных системах [3]. Одним из используемых критериев для выбора мигрирующего агента является так называемый кредит [1]. Система распределяет кредит каждому агенту и решает, какого агента необходимо перенести, используя значение этого кредита.

Ключевые слова: мультиагентная система, балансировка нагрузки, подсистема балансировки нагрузки на основе кредита.

Рассмотрим основные проблемы, которые могут возникнуть в процессе эксплуатации распределенной сети: различная нагрузка агентов, множество задач [2], потребление больших системных ресурсов. Гибридная система предложена в [4] для решения каждого типа проблемы в зависимости от ситуации. У каждого узла есть координаторы балансировки нагрузки, которые собирают информацию о своем расположении. С помощью собранной информации они могут решить, какие действия целесообразны для решения текущих проблем. Хотя модель балансировки нагрузки в [2] фокусируется на второй проблеме, они используют распределение задач для перераспределения нагрузки на агента.

Предлагаемая система балансировки нагрузки нацелена на использование характеристик агента для создания автономной системы.

Система проходит три этапа. На первом этапе перед запуском строится модель множественной линейной регрессии для расчета параметров уравнений регрессии как для политики выбора, так и для политики местоположения. Полученные линейные модели используются всеми узлами посредством их операций. Затем выполняется множественная линейная регрессия. Одну и ту же линейную модель можно использовать для нескольких

прогонов системы при условии, что параметры агентов и системы совпадают.

Учитывая переменную y и количество переменных $X_1, \dots, X_p$, связанные с y , можно применять линейный регрессионный анализ для количественной оценки силы взаимосвязи между y и X_j , а именно, какой X_j может вообще не иметь отношения с y , и для определения того, какие подмножества X_j содержат избыточную информацию о y , таким образом, когда один из них известен, а другие больше не являются информативными.

К этому предполагаемому отношению добавляется e_i , чтобы отразить влияние всего остального на Y_i , кроме $X_1, \dots, X_p$. Общая формула для уравнения регрессии имеет вид:

$$Y = b_0 + b_1 X_1 + b_2 X_2 + \dots + b_p X_p + e_i, \text{ где}$$

X_i – переменные регрессоры. Y_i – зависимые переменные или переменные отклика, или предсказанными переменными.

Второй этап – это процесс сбора необходимых значений агентов и системных параметров, эти значения используются на следующем этапе для оценки полученных уравнений, полученных на первом этапе. Третий этап – это оперативная оценка этих политик для выбора подмножества локальных агентов для миграции. Он также выбирает подмножество системных узлов для получения этих агентов.

Агентов можно разделить на административных и рабочих. Каждый административный агент предназначен для управления нагрузкой локального узла и принятия решения о миграции для достижения балансировки нагрузки, в то время как рабочие агенты отвечают за выполнение пользовательских задач.

Предлагаемая модель балансировки нагрузки фокусируется на трех политиках: сбор информации, политика выбора и размещения.

Данная политика основана на событиях. В частности, узел, который обновляет свое локальное значение нагрузки из-за локального события, оценит свое значение Credit Location (LC), которое представляет локальное значение нагрузки, и отправит его другим агентам координаторам балансировки нагрузки (КБН) в системе. В каждом из событий значение локальной нагрузки изменяется, увеличиваясь или уменьшаясь в зависимости от ситуации. Состояние загрузки определяется путем сравнения значения LC с пороговым значением нагрузки, обновленное значение LC отправляется другим агентам КБН, чтобы они обновили свой список узлов-кандидатов.

Политика выбора определяет какая задача или агент переносятся всякий раз, когда в этом есть необходимость. Каждому агенту присваивается числовое значение, называемое «Совокупный выборный критерий» (СВК). Значение СВК указывает на склонность агента оставаться нетронутым. Для агента, чем выше его значение СВК, тем выше его шанс не попасть под миграцию.

Предполагается, что значение СВК агента зависит от двух типов параметров:

Агент-зависимые параметры:

1. вычислительная нагрузка (W);
2. коммуникационная нагрузка (U);
3. размер агента ($S = 0$ (маленький), $S = 1$ (средний));
4. приоритет агента ($P_1, P_2 = 0, 0$ (низкий); $P_1, P_2 = 1, 0$ (средний); $P_1, P_2 = 1, 1$ (высокий)).

Системно-зависимые параметры:

1. надежность канала связи между узлами ($H_1, H_2 = 0, 0$ (высокая); $H_1, H_2 = 1, 0$ (средняя); $H_1, H_2 = 1, 1$ (низкая));
2. наличие необходимого ресурса на исходном узле (A);
3. нагрузка узла ($L_1, L_2 = 0, 0$ (высокая); $L_1, L_2 = 1, 0$ (средняя); $L_1, L_2 = 1, 1$ (низкая)).

Используя операцию множественной линейной регрессии, соберем все упомянутые факторы в одно уравнение:

$$CBK = b_0 + b_1W + b_2U + b_3R + b_4L_1 + b_5L_2 + b_6H_1 + b_7H_2 + b_8P_1 + b_9P_2 + b_{10}S$$

Политика расположения определяет на какой целевой компьютер будет перенесен выбранный агент. Выбор узла назначения основан на LC, рассчитанном агентом КБН.

Основным и единственным фактором для выбора пункта назначения является его локальная нагрузка. В этом разделе опишем факторы, влияющие на локальную нагрузку.

Загрузка локального узла зависит от загрузки агентов, работающих на нем. Нагрузка агента, выполняющегося на машине, определяется как сумма его вычислительной нагрузки и коммуникационной нагрузки на единицу времени [1].

Таким образом, нагрузка на локальный узел может быть смоделирована следующим образом:

$$L_k = W_k + U_k, \text{ где}$$

W_k – суммарное значение вычислительной нагрузки работающих агентов на компьютере k . U_k – общее значение коммуникационной нагрузки работающих агентов на компьютере k . L_k – общая локальная нагрузка на машину k .

В случае разной пропускной способности вышеописанная сумма делится на нее для каждой машины.

Поскольку суммарный вес вычислительной нагрузки и коммуникационной нагрузки может различаться в зависимости от некоторых параметров агента и системы, выполним множественную линейную регрессию.

Поэтому полное уравнение LC можно записать в виде:

$$LC_k = b_0 + b_1W_k + b_2U_k, \text{ где}$$

W_k – сумма вычислительной нагрузки всех работающих агентов на узле k . U_k – сумма коммуникационной нагрузки всех работающих агентов на узле k .

Система состоит из семи агентов. Агент, который выполняет все вычисления и находится на вычислительных узлах, это Рабочий агент. Это единственный тип агента, который подвержен миграции. Второй тип – агент координатора балансировки нагрузки (КБН), который находится на каждом вычислительном узле. Агент КБН подготавливает список агентов-кандидатов и список узлов-кандидатов и обновляет их при необходимости. Агент КБН работает в сотрудничестве с хост-агентом и агентом-отправителем для выполнения процесса миграции [5].

Третий тип – это хост-агент, который отслеживает количество живых рабочих агентов на локальном узле. На каждом узле должен быть один хост-агент. Четвертый тип – это Агент-отправитель, который отвечает за доставку агента переноса на выбранный узел. Каждый узел имеет своего собственного агента-отправителя, который ожидает подтверждения от перенесенного агента, как только он достигает своего узла назначения. Агент-отправитель следит за тем, чтобы агент-мигрант был успешно принят и инициирован на узле назначения. Центральный агент запускает агенты КБН. Во время каждого запуска системы регистрируется статистика изменений рабочей нагрузки и решений о миграции. Шестой тип – пороговый агент; он должен работать на каждом узле, отвечает за сравнение значения локальной нагрузки с заданным пороговым значением для определения состояния локального узла. Последний тип – коммуникационный агент, ответственный за доставку сообщений между различными типами агентов.

Выделенный агент на каждом узле отвечает за локальную координацию действий по балансировке нагрузки. Каждый агент КБН решает, при каких обстоятельствах инициировать миграцию, а также принимать входящие миграции.

После сбора информации о локальных рабочих агентах, находящихся на локальном узле, агент КБН готовит список кандидатов на агентов для миграции. Агент КБН должен сначала удалить локально прерванные и перенесенные агенты из списка кандидатов. Он также вычисляет:

$$W_k = W_k' - W_i$$

$$U_k = U_k' - U_i, \text{ где}$$

W_k, U_k – нагрузки после миграции агента. W_k', U_k' – нагрузки до переноса агента. W_i, U_i – нагрузки перенесенного или прерванного агента.

Затем агент КБН оценивает уравнение СВК для каждого рабочего агента и сортирует их в порядке возрастания их значений СВК. Первым агентом, которого нужно перенести, будет

агент с наименьшим значением СВК.

После полученной информации о нагрузке от активных рабочих агентов на локальном узле, агент КБН сможет рассчитать локальное значение нагрузки, оценив уравнение LC. Затем пороговый агент сравнит это значение с пороговым значением, чтобы определить состояние узловой нагрузки. Пороговый агент отправит обратно состояние загрузки агенту КБН, а также информацию о доступности ресурса. Если состояние локальной загрузки невелико, агент КБН отправит свое значение локальной загрузки и информацию о доступности ресурсов другим КБН, чтобы они добавили его в свой список узлов-кандидатов для выбора места назначения.

Когда агент КБН получает информацию о загрузке от других КБН в системе, он сортирует их в порядке возрастания их значений LC. Если мигрирующий агент запрашивает конкретный ресурс, агент КБН исключит узлы-кандидаты, у которых нет этого ресурса, из списка.

Если мигрирующий агент принят, общая вычислительная и коммуникационная нагрузка входящего агента должна быть добавлена к общей вычислительной и коммуникационной.

В данной статье была представлена схема динамической балансировки нагрузки в мультиагентной системе, основанную на использовании уравнений линейной регрессии для принятия решения о выборе и местоположении. Также использовались характеристики агентов для разработки схемы распределения нагрузки для мультиагентных приложений распределенных систем. Предложенное динамическое балансирование нагрузки использует критерии, основанные на кредитах, как в [1], однако учитывается больше факторов для расчета стоимости кредита для выбора агента и узла, влияние этих факторов изучается с помощью анализа множественной линейной регрессии.

Список литературы:

1. Chow K-P, Kwok Y-K. On load balancing for distributed multiagent computing. IEEE Trans Parallel Distrib Syst 2002;13(8):1153–61.
2. Балансировка нагрузки в распределенных системах [Электронный ресурс]. – Режим доступа: www.intuit.ru/department/algorithms/distrsa/9/, для зарегистрированных пользователей (дата обращения: 17.12.2014).
3. Kim YH, Han S, Lyu CH, Youn HY. An efficient dynamic load balancing scheme for multi-agent system reflecting agent work- load. In: The 12th IEEE international conference on computational science and, engineering; 2009.
4. Ozcan I, Bora S. A hybrid load balancing model for multi-agent systems. In: Innovation in intelligent systems and applications (INSTA); 2011. p. 182–7.
5. Nehra N, Patel RB, Bhat VK. A framework for distributed dynamic load balancing in

heterogeneous cluster. J Comput Sci 2007;3(1):14–24.