

МЕТОДЫ ОЦЕНКИ АЛГОРИТМОВ ДИСПЕТЧЕРИЗАЦИИ НТТР-ЗАПРОСОВ

Развитие и внедрение в повседневную жизнь человека информационных технологий обусловило резкое увеличение числа запросов на обработку и увеличение нагрузки на серверы. Поскольку нагрузка постоянно растет, задачи балансировки приобретают все более важное значение для повышения эффективности информационных систем.

Не менее важной задачей является оценка алгоритмов балансировки нагрузки для того, чтобы понимать, насколько эффективен тот или иной алгоритм балансировки нагрузки.

Для анализа эффективности алгоритмов балансировки нагрузки была разработана математическая модель программной системы с микросервисной архитектурой. Эта система включает 3 типа компонентов: клиенты ($C_i, i = \overline{1, \bar{i}}$), серверы ($S_j, j = \overline{1, \bar{j}}$) и балансировщик нагрузки (B).

Балансировщик нагрузки – это программная компонента, которая является промежуточным звеном между клиентами и серверами: он принимает запросы от клиентов и перенаправляет их серверам и, наоборот, принимает ответы от серверов и отправляет соответствующим клиентам. В запросах и ответах, помимо основной информации, содержится дополнительная информация (α и β), которая может быть использована для управления поведением сервера и балансировщика.

Если кратко рассматривать математическую модель, то всю ту программную систему можно описать как четверку:

$$W = \left\langle \left\{ C_j, j = \overline{1, \bar{j}} \right\}, \left\{ K_l, l = \overline{1, \bar{l}} \right\}, B, \left\{ S_i, i = \overline{1, \bar{i}} \right\} \right\rangle, \quad (1)$$

где $C_j, j = \overline{1, \bar{j}}$ – множество клиентов; $K_l, l = \overline{1, \bar{l}}$ – множество категорий запросов (величина, характеризующая относительную трудоемкость обработки запроса); B – балансировщик; $S_i, i = \overline{1, \bar{i}}$ – множество серверов.

Критерием эффективности алгоритма балансировки нагрузки было принято отношение:

$$\frac{M(t)}{L(t)}, \quad (2)$$

где $L(t)$ – количество поступивших в балансировщик запросов; $M(t)$ – количество поступивших в балансировщик ответов к заданному мо-

менту времени t . При этом предполагается, что чем это отношение ближе к 1, тем более эффективным считается алгоритм.

В общем случае, эффективность алгоритма балансировки нагрузки зависит от конфигурации программной системы (количество серверов и их производительность, интенсивность клиентских запросов и их трудоемкость и т. д.) и может изменяться в течение времени.

Для оценки $E_A(t)$ эффективности алгоритма балансировки нагрузки A удобно иметь оценки нижней $\underline{E}(t)$ и верхней $\overline{E}(t)$ границ эффективности алгоритмов балансировки нагрузки, вычисленные при заданных условиях $P = \langle \{C_j, j = \overline{1, \overline{j}}\}, \{K_l, l = \overline{1, \overline{l}}\}, \{S_i, i = \overline{1, \overline{i}}\} \rangle$.

Нижняя граница $\underline{E}(t) \equiv 0$ малоинформативная, поэтому в качестве оценки нижней границы целесообразно выбрать эффективность какого-нибудь известного и простого алгоритма, например, Round Robin [1].

Теоретический максимум $\overline{E}_P(t)$ эффективности на заданном участке времени $[0, t_e]$ при заданных условиях P может быть достигнут алгоритмом $\overline{A}$ в том случае, если он может предсказать моменты времени и трудоемкости поступающих от клиентов запросов, а также производительности серверов и моменты завершения обработки серверами запросов. В этом случае, задача вычисления функции $\overline{E}_P(t)$ сводится к задачам поиска и анализа планов распределения между серверами поступающих от клиентов запросов.

Таким образом, вычисление точных границ $\underline{E}_P(t)$ и $\overline{E}_P(t)$ при заданных условиях P на отрезке времени $[0, 10]$ сводится к трем задачам:

- построение множества $\Pi = \{ \pi_1, \pi_2, \dots, \pi_k \}$ всех рациональных планов;
- анализ функций эффективности $\varepsilon_i(t) = \frac{M_i(t)}{L_i(t)}, i = \overline{1, \overline{k}}, t \in [0, t]$ планов;
- построение функций $\overline{E}_P(t) = \max_i \varepsilon_i(t)$ и $\underline{E}_P(t) = \min_i \varepsilon_i(t)$ для получения точных границ.

Предложенный метод вычисления точных границ эффективности алгоритмов балансировки нагрузки, предполагает, что помимо конфигурации программной системы, известна последовательность (во времени) запросов и их трудоемкость. В реальных условиях может быть известна только конфигурация и в лучшем случае интенсивности запросов по категориям трудоемкости их обслуживания. Алгоритм для вычисления границ имеет экспоненциальную сложность от вре-

мени t и является трудно разрешимым для реальных конфигураций программных систем.

Кроме того, как правило, интерес представляет эффективность программной системы в установившемся режиме, стабилизация работы которой может занимать достаточно продолжительный интервал времени. Все это делает непригодным применение предложенного метода для оценки эффективности алгоритмов балансировки нагрузки в инженерных задачах, когда требуется оценить эффективность существующей программной системы. Оценка эффективности в таких случаях, как правило, осуществляется с помощью нагрузочного тестирования, которое позволяет на основе собранной статистики получить обоснованную оценку.

ЛИТЕРАТУРА

1. Stuti D., Prashant M. Utilizing Round Robin Concept for Load Balancing Algorithm at Virtual Machine Level in Cloud Computing // International Journal of Computer Applications. 2014. Vol. 94. No. 4. P. 23–29.

УДК 004.421

Е. Н. Бондарчик, маг.; Е. А. Блинова, ст. преп. (БГТУ, г. Минск)

МЕТОДЫ И АЛГОРИТМЫ РАСПРЕДЕЛЕНИЯ НАГРУЗКИ ПРЕПОДАВАТЕЛЕЙ КАФЕДРЫ УНИВЕРСИТЕТА

В рамках управления учебным процессом учреждения высшего образования решается одна из наиболее важных задач – задача распределения нагрузки кафедры между профессорско-преподавательским составом оптимальным образом. Для оптимального распределения учебной нагрузки кафедры необходимо иметь возможность моделировать различные ее варианты, изменяя исходные данные. Варьируя распределением планируемой нагрузки кафедры по семестрам, необходимо спланировать оптимальным образом учебный процесс с тем расчетом, чтобы наибольший объем учебной нагрузки по приоритетным типам работ назначался наиболее компетентному преподавателям с учетом равномерной их загрузки в учебном году [1].

Исходными данными для задачи распределения учебной нагрузки между преподавателями кафедры, являются: учебная нагрузка кафедры по читаемым ей дисциплинам; плановое штатное расписание кафедры; фактический штат преподавателей кафедры; критерии и ограничения.