

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

INFORMATION TECHNOLOGIES



УДК 004.75
<https://doi.org/10.37661/1816-0301-2022-19-4-84-93>

Оригинальная статья
Original Paper

Модель балансировки нагрузки кластерной системы с учетом аппаратных характеристик серверного оборудования

А. Н. Марков

*Центр информатизации и инновационных разработок
Белорусского государственного университета
информатики и радиоэлектроники,
ул. Платонова, 39, Минск, 220013, Беларусь
E-mail: a.n.markov@bsuir.by*

Аннотация

Цели. Поставлены цели модернизировать и дополнить существующую модель балансировки нагрузки в многосерверных системах с учетом аппаратных характеристик серверного оборудования, а также его наиболее загруженных компонентов в кластере сервиса видео-конференц-связи при использовании в образовательных процессах и организации дистанционной формы образования.

Методы. Существующая математическая модель балансировки нагрузки рассмотрена как система массового обслуживания, в которой вводятся штрафы за простой оборудования, а штрафы за ожидание в очереди будут зависеть от загруженности аппаратных компонентов серверов в кластерной архитектуре сервиса видео-конференц-связи.

Результаты. Приведены формулы для вычисления суммарной производительности кластера из n серверов при максимальной и минимальной загрузках аппаратных компонентов серверов в кластере системы видео-конференц-связи.

Заключение. Разработан моделирующий комплекс для проверки математической модели на системе до $n < 10$ серверов в кластере системы видео-конференц-связи. По результатам вычислений моделирующего комплекса сделан вывод о необходимости доработки существующего алгоритма балансировки нагрузки на выбранный сервис видео-конференц-связи BigBlueButton.

Ключевые слова: видео-конференц-связь, сервис BigBlueButton, балансировка нагрузки в кластере, математическая модель балансировки, дистанционное образование

Для цитирования. Марков, А. Н. Модель балансировки нагрузки кластерной системы с учетом аппаратных характеристик серверного оборудования / А. Н. Марков // Информатика. – 2022. – Т. 19, № 4. – С. 84–93. <https://doi.org/10.37661/1816-0301-2022-19-4-84-93>

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Поступила в редакцию | Received 13.07.2022

Подписана в печать | Accepted 26.07.2022

Опубликована | Published 29.09.2022

Cluster system load balancing model with consideration of hardware characteristics of server hardware

Aleksey N. Markov

*Center for Informatization and Innovation Development
of the Belarusian State University
of Informatics and Radioelectronics,
st. Platonova, 39, Minsk, 220013, Belarus
E-mail: a.n.markov@bsuir.by*

Abstract

Objectives. To upgrade and complement the existing load balancing model in multi-server systems, taking into account the hardware characteristics of the server equipment, as well as the most loaded components of the server equipment in the video conferencing service cluster in educational processes and distance education.

Methods. The existing mathematical model of load balancing as a mass exchange system is considered, when significant changes are introduced: penalties for equipment downtime and penalties for waiting in a queue will depend on the load on the server hardware components in the cluster architecture of video conferencing service.

Results. Formulas are given for calculating the total performance of a cluster of n servers with the maximum and minimum load of server hardware components in a videoconferencing system cluster.

Conclusion. A modeling complex has been developed to test the mathematical model on a system of up to $n < 10$ servers in a cluster of a videoconferencing system. Based on the results of calculations of the modeling complex, it was concluded that it is necessary to upgrade the existing algorithm for balancing the load on the selected BigBlueButton video conferencing service.

Keywords: video conferencing, service BigBlueButton, load balancing in a cluster, mathematical model of load balancing, distance education

For citation. Markov A. N. *Cluster system load balancing model with consideration of hardware characteristics of server hardware*. *Informatika [Informatics]*, 2022, vol. 19, no. 4, pp. 84–93 (In Russ.).
<https://doi.org/10.37661/1816-0301-2022-19-4-84-93>

Conflict of interest. The author declares of no conflict of interest.

Введение. В ходе реализации проекта цифровой трансформации образования на этапе перехода к смешанной модели обучения в рамках системы электронного обучения в БГУИР был внедрен сервис видео-конференц-связи BigBlueButton [1, 2]. В процессе интеграции видеосервиса в учебный процесс выявлена проблема повышенной нагрузки на единственный сервер, которая описана в работе [3]. На этапе тестовой эксплуатации эмпирическим путем было установлено, что увеличение аппаратных ресурсов сервера BigBlueButton относительно первоначально заданных параметров не приносит значительного прироста производительности сервиса в целом. Одной из главных причин нестабильности подключений и ненадлежащего качества видеопотока является превышение допустимого числа подключений к сервису. Для выхода из существующей проблемы был развернут сервис балансировки нагрузки системы видео-конференц-связи, решающий проблему распределения пользователей в зависимости от доступности каждого сервера и устранения задержек на этапах подключения к видеоконференции. Стандартный алгоритм балансировки нагрузки не учитывал параметры загрузки серверного оборудования. Встроенный метод балансировки нагрузки на основе алгоритма Round Robin DNS [4] учитывает только доступность ресурса без учета нагрузки на аппаратные компоненты серверов в кластере. Для устранения данного недостатка системы можно взять за основу математическую модель балансировки нагрузки в адаптивном алгоритме управления распределением нагрузки в многосерверных системах [5] и доработать ее с учетом загрузки аппаратных компонентов серверного оборудования в кластере. Другие математические модели для балансировки нагрузки (грид-вычисления) нельзя принять за основу, так как принципиальный под-

ход к балансировке нагрузки в них осуществляется на аппаратном уровне самого сервера (процессора и распределения потоков в операционной системе) без учета количества серверов в кластере.

Критерии качества работы многосерверной системы. Критериями качества многосерверных систем являются затраты, связанные с простоем серверов в кластере и временем ожидания подключения к конференции при наличии очереди на подключение. Оптимизация работы системы заключалась в использовании этих критериев для вычисления таких параметров работы серверов с учетом выбранного оборудования, при которых затраты будут минимальными. В процессе оптимизации подбираются доли распределения входного потока между серверами в кластере α_i и время доступа к ресурсу b_i (выраженное как время обработки одного запроса на вход в конференцию) с учетом аппаратных компонентов серверов при заданном количестве серверов n в кластере, где $i = \overline{1, n}$ – номер сервера в кластере.

Для определения минимальных значений критериев качества кластер выделен как система массового обслуживания типа $M | D | 1$. На основании системы $M | D | 1$ суммарный критерий качества работы системы (суммарные затраты на ожидание в очереди и суммарные затраты на простой сервера в кластере), выраженный через метод множителей Лагранжа, будет иметь следующий вид [5]:

$$\min \left\{ S(n, \lambda_1, \lambda_2 \dots \lambda_n, b_1, b_2 \dots b_n) = \sum_{i=1}^n \left\{ \frac{Pw_i \lambda_i^2 b_i^2}{2(1 - \lambda_i b_i)} + Pd_i \cdot (1 - \lambda_i b_i) \right\} \right\},$$

$$\sum_{i=1}^k \lambda_i = \Lambda, \lambda_i \geq 0, \quad (1)$$

где $i = \overline{1, n}$ – номер сервера в кластере;

n – количество серверов в вычислительном кластере;

λ_i – интенсивность потока запросов на сервер i ;

Λ – суммарная интенсивность потока запросов на кластер видео-конференц-связи, определяющая, на сколько пользователей какое количество запросов на вход в видеоконференцию распределено;

b_i – время обработки одного запроса на вход в конференцию, с;

Pw_i – штраф за ожидание в очереди на подключение к конференции;

Pd_i – штраф за простой сервера в кластере.

Сумма потоков заявок на все серверы равна общему потоку запросов на вход в конференцию, а значение потока запросов на каждый сервер – величина неотрицательная.

Отыскание точки минимума функции позволит определить параметры работы кластера с учетом штрафа за ожидание в очереди и штрафа за простой сервера исходя из выделенных аппаратных ресурсов серверов.

Зависимость величины штрафа за ожидание в очереди от времени доступа к серверу должна быть обратной: чем больше время обработки запроса, учитывающее загруженность сервера, тем «дешевле» должен быть сервер и меньше штраф за ожидание в очереди такого сервера.

Предположим, что при условии организации кластера видео-конференц-связи штраф за ожидание в очереди можно определить выражением

$$Pw_i = \frac{Pr_i}{b_i}, \quad (2)$$

где Pr_i – суммарная загрузка сервера i , выраженная как произведение производительностей всех аппаратных компонентов:

$$Pr_i = (P_{cpu\ max})_i \cdot (P_{ram\ max})_i \cdot (P_{netw\ max})_i \cdot (P_{hdd\ max})_i, \quad i = \overline{1, n}. \quad (3)$$

Здесь $(P_{cpu\ max})_i$ – производительность процессора в сервере i ;

$(P_{ram\ max})_i$ – производительность памяти сервера i ;

$(P_{netw\ max})_i$ – производительность локальной сети сервера в кластере;

$(P_{hdd\ max})_i$ – производительность дискового хранилища сервера.

Учитывая специфику работы серверов и выделенные аппаратные ресурсы, параметрами, не оказывающими влияние на итоговую загруженность системы (при условии использования видео-конференц-связи на выбранном кластере), можно пренебречь согласно анализу графиков нагрузки на аппаратные компоненты [4]:

$$P_{ram\ max} = 1, \quad P_{netw\ max} = 1, \quad P_{hdd\ max} = 1. \quad (4)$$

Между тем предположим, что кластер построен на различных выделенных серверах. Тогда его итоговая загруженность будет тем выше, чем больше загружены оперативная память, сеть и дисковое хранилище.

Для выделенного кластера с учетом аппаратных ресурсов общая загруженность серверов будет зависеть преимущественно от производительности центрального процессора:

$$Pr_i = (P_{cpu\ max})_i, \quad i = \overline{1, n}. \quad (5)$$

Основываясь на методе расчета и анализа теоретической пиковой производительности процессоров [6–9] с изменениями ввиду специфики работы сервиса, предположим, что производительность процессора в серверах рассчитывается следующим образом:

$$(P_{cpu\ max})_i = F_{cpu\ max\ i} \cdot Fl_i \cdot v_i \cdot H_{осн\ i}, \quad (6)$$

где $F_{cpu\ max\ i}$ – турбочастота процессора (для некоторых процессоров это максимальная частота работы). С учетом систем виртуализации выделенные аппаратные ресурсы будут напрямую зависеть от процессоров в сервере i ;

Fl_i – количество операций с плавающей запятой в секунду, выполняемых данной вычислительной системой, на одно ядро процессора, флопс;

v_i – количество ядер в выделенном процессоре сервера i ;

$H_{осн\ i}$ – усредненный коэффициент загрузки процессора сервера i на выполнение основных операций 0,95, показывающий долю от производительности процессора на выполнение задач распределения потока. Оставшаяся доля 0,05 – усредненный коэффициент, определяющий выполнение фоновых задач для сервера i на базе виртуализированной Unix-системы (коэффициенты 0,90 и 0,10 – для сервера на базе системы Windows).

Параметры, влияющие на общую производительность, а именно: величина кеш-памяти, объем физической адресуемой памяти, разрядность внешних шин данных и адреса, разрядность регистров, не учитываются, поскольку напрямую не влияют на показатели производительности процессора при обработке потока запросов в единицу времени.

Простой сервера в кластере (с учетом специфики работы видео-конференц-связи) является естественным процессом в отличие от простоя веб-серверов или серверов баз данных. Пользо-

вателей конференции может быть минимальное количество, и тогда все серверы в кластере не используются. Чем меньше время обслуживания запроса от клиента на доступ к конференции, тем большую итоговую производительность обеспечивает сервер, тем «дороже» время работы такого сервера и тем больше должен быть штраф за его простой. Предположим, что штраф за простой сервера будет рассчитываться исходя из минимальных значений производительности сервера:

$$Pd_i = \frac{Pr(\min)_i}{b_i}, \quad (7)$$

где $Pr(\min)_i$ – минимальная производительность сервера i в кластере, выраженная как произведение минимальной производительности всех аппаратных компонентов серверов:

$$Pr(\min)_i = (P_{cpu\ min})_i \cdot (P_{ram\ min})_i \cdot (P_{netw\ min})_i \cdot (P_{hdd\ min})_i, \quad i = \overline{1, n}. \quad (8)$$

Исходя из анализа графиков работы серверов в кластере и учитывая выделенные аппаратные ресурсы, можно сделать вывод, что параметрами, не оказывающими влияние на производительность системы, можно пренебречь [4]:

$$P_{ram\ min} = 1, \quad P_{netw\ min} = 1. \quad (9)$$

Между тем минимальная производительность системы хранения данных (или жесткого диска в выделенном сервере) будет оказывать влияние на работу сервера в целом. Поскольку $P_{hdd\ min} = 2$, простой дискового массива более критичен по сравнению с рабочим режимом массива в целом, что обусловлено временем доступа (обращения) к данным на диске в массиве.

Если кластер построен на различных выделенных серверах, итоговая минимальная величина учета его производительности будет тем ниже, чем менее производительными будут оперативная память, сеть и хранилище.

Для выделенного кластера с учетом ресурсов общая минимальная производительность сервера i может определяться выражением

$$Pd_i = (P_{cpu\ min})_i (P_{hdd\ min})_i, \quad i = \overline{1, n}. \quad (10)$$

Предположим, что аналогично расчету пиковой производительности минимальную производительность процессора для сервера i можно найти по формуле

$$Pr(\min)_i = (P_{cpu\ min})_i = F_{cpu\ mini} \cdot Fl_i \cdot v_i \cdot H_{фонов\ i} \cdot m, \quad (11)$$

где $m = 3,5$ – поправочный коэффициент весомости простоя самого низкопроизводительного сервера. Физический смысл коэффициента заключается в том, что даже самый низкопроизводительный сервер в кластере должен быть минимально загружен для повышения общей производительности, нежели не быть использован вовсе. Коэффициент подбирается экспериментальным путем с учетом выделенных аппаратных ресурсов серверов в кластере. В случае построения кластера на различных процессорах в серверах коэффициент подбирается исходя из экспериментальных данных по загрузке и простоя каждого сервера в отдельности;

$F_{cpu\ mini}$ – базовая частота процессора сервера i (для некоторых процессоров это минимальная частота работы);

$H_{фонов\ i}$ – усредненный коэффициент учета загрузки процессора сервера i на выполнение фоновых операций 0,2, показывающий долю от производительности одного процессора на выпол-

нение задач распределения потока. Оставшаяся доля 0,8 – усредненный коэффициент, учитывающий основные задачи для сервера на базе системы Unix (коэффициенты 0,3 и 0,7 – для сервера на базе системы Windows).

Задачи распределения входного потока между серверами и выбора производительности серверов. Оптимизация системы серверов, связанная одновременно с распределением входного потока и выбором производительности серверов, является задачей математического программирования с ограничениями, которую аналитически или численно можно решить, используя метод множителей Лагранжа. Для нахождения оптимальных значений α_i и b_i методом множителей Лагранжа необходимо решить систему уравнений, где частные производные по всем переменным равны нулю. С учетом подстановок и выражений функций получим систему уравнений

$$\left\{ \begin{array}{l} -\frac{Pw_1\lambda_1\Lambda b_1^2(\lambda_1 b_1 - 2)}{2(\lambda_1 b_1 - 1)^2} - Pd_1\Lambda b_1 + x = 0; \\ -\frac{Pw_2\lambda_2\Lambda b_2^2(\lambda_2 b_2 - 2)}{2(\lambda_2 b_2 - 1)^2} - Pd_2\Lambda b_2 + x = 0; \\ \dots\dots\dots \\ -\frac{Pw_n\lambda_n\Lambda b_n^2(\lambda_n b_n - 2)}{2(\lambda_n b_n - 1)^2} - Pd_n\Lambda b_n + x = 0; \\ \sum_{i=1}^k \alpha_i - 1 = 0; \\ -\frac{Pw_1\alpha_1^2\Lambda^2 b_1(\alpha_1\Lambda b_1 - 2)}{2(\alpha_1\Lambda b_1 - 1)^2} + Pd_1 \frac{(1 + \alpha_1\Lambda)}{b_1} = 0; \\ -\frac{Pw_2\alpha_2^2\Lambda^2 b_2(\alpha_2\Lambda b_2 - 2)}{2(\alpha_2\Lambda b_2 - 1)^2} + Pd_2 \frac{(1 + \alpha_2\Lambda)}{b_2} = 0; \\ \dots\dots \\ -\frac{Pw_n\alpha_n^2\Lambda^2 b_n(\alpha_n\Lambda b_n - 2)}{2(\alpha_n\Lambda b_n - 1)^2} + Pd_n \frac{(1 + \alpha_n\Lambda)}{b_n} = 0. \end{array} \right. \quad (12)$$

Решение системы уравнений (12) позволит вычислить оптимальное распределение входного потока запросов при оптимальных параметрах производительности серверов, а также сами эти параметры. Данная задача является многомерной. Решения многомерных задач всегда очень сложны, поэтому для простоты изложения предлагается упростить решение системы уравнений (12), разбив его на два этапа.

В качестве примера возьмем задачу оптимизации работы кластера из двух серверов, которая связана с поиском оптимального распределения входного потока между ними и заключается в нахождении минимума критерия качества по величине α_i . С учетом параметров загрузки оборудования задача распределения входного потока между серверами будет иметь следующий вид:

$$\min_{\alpha_i} \left\{ S(\alpha_1, \alpha_2 \dots \alpha_n) = \sum_{i=1}^n \left\{ \frac{Pw_i \alpha_i^2 \Lambda^2 b_i^2}{2(1 - \alpha_i \Lambda b_i)} + Pd_i(b_i) \cdot (1 - \alpha_i \Lambda b_i) \right\} \right\}, \quad \sum_{i=1}^n \alpha_i = 1, \quad \alpha_i \geq 0. \quad (13)$$

Сумма долей входного потока на кластер равна общему потоку заявок на вход в видеоконференции, а доля запросов на каждый сервер в кластере – величина неотрицательная.

Частный случай задачи может быть представлен через систему уравнений. Оптимизация работы кластера является задачей математического программирования. Для первичной оценки системы и решения данной задачи применим метод множителей Лагранжа. Система уравнений для решения задачи распределения потока для двух серверов в кластере с учетом преобразований будет иметь вид

$$\begin{cases} -\frac{Pw_1\alpha_1^2\Lambda^2b_1(\alpha_1\Lambda b_1-2)}{2(\alpha_1\Lambda b_1-1)^2} - Pd_1\Lambda + \frac{Pw_2(1-\alpha_1)^2\Lambda^2b_2((1-\alpha_1)\Lambda b_2-2)}{2((1-\alpha_1)\Lambda b_2-1)^2} + Pd_2\Lambda = 0; \\ x = \frac{Pw_2(1-\alpha_1)^2\Lambda^2b_2((1-\alpha_1)\Lambda b_2-2)}{2((1-\alpha_1)\Lambda b_2-1)^2} + Pd_2\Lambda; \\ \alpha_2 = 1 - \alpha_1. \end{cases} \quad (14)$$

Типовым подходом к решению системы уравнений n с $n - 1$ неизвестными будет использование численных методов. Для решения численными методами написана программа, с помощью которой был осуществлен перебор с заданным шагом (0,001) всех возможных вариантов α_1 и α_2 , удовлетворяющих условиям $\sum_{i=1}^n \alpha_i = 1$, $\alpha_i > 0$. Распределение потока запросов между двумя серверами показано на рис. 1.

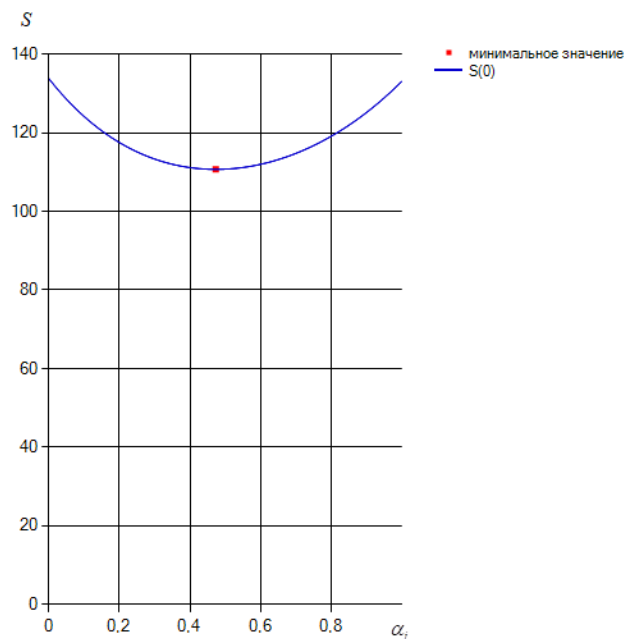


Рис. 1. График распределения потока запросов для кластера из двух серверов

Fig. 1. Diagram of the distribution of the flow of requests for a cluster of two servers

В результате пошаговых вычислений было рассчитано наиболее приемлемое распределение потока запросов в кластере из двух серверов: $\alpha_1 = 0,492$, $\alpha_2 = 0,508$ при $S(\alpha_1, \alpha_2) = 106,254$, что видно из рис. 1.

Оптимизация работы сервера в кластере по времени доступа к ресурсу также является задачей математического программирования. Для решения данной задачи применим метод множи-

телей Лагранжа. Выражение для решения задачи поиска времени доступа для серверов в кластере с учетом преобразований будет иметь вид

$$\min_{b_i} \left\{ S(b_i) = \sum_{i=1}^n \left\{ \frac{Pw_i \alpha_i^2 \Lambda^2 b_i^2}{2(1 - \alpha_i \Lambda b_i)} + Pd_i \cdot (1 - \alpha_i \Lambda b_i) \right\} \right\}. \quad (15)$$

Для кластера, состоящего из одного сервера, $\alpha = 1$. Тогда с учетом преобразований критерий качества по времени доступа к одному серверу будет определяться выражением

$$\min_b \left\{ S(b) = \frac{Pw\Lambda^2 b^2}{2(1 - \Lambda b)} + Pd(1 - \Lambda b) \right\}. \quad (16)$$

Величина времени доступа к серверу b_i , при которой рассчитанное значение критерия качества минимально, будет решением задачи оптимизации критерия качества.

Типовым подходом для решения задачи будет использование численных методов. Для решения численными методами написана программа, с помощью которой был осуществлен перебор с заданным шагом (0,001) всех возможных вариантов b . Для каждого значения времени доступа рассчитана величина критерия качества. Построим график зависимости величины критерия качества работы системы от величины времени доступа к серверу путем вычисления величины критерия качества $S(b)$ для значения времени доступа b , лежащего на отрезке $[0;2]$ (рис. 2).

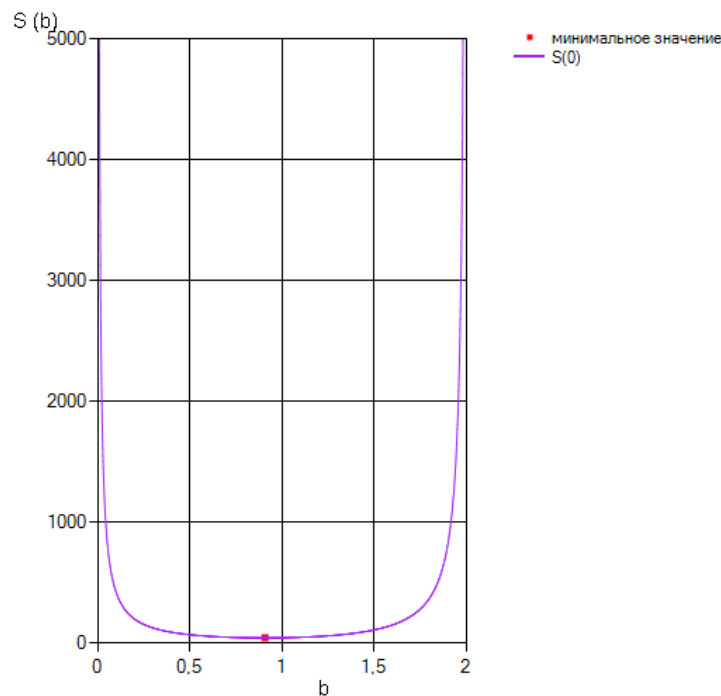


Рис. 2. График распределения производительности для одного сервера

Fig. 2. Diagram of performance distribution for one server

Пошаговые вычисления показали, что наиболее приемлемое значение критерия качества для различных распределений производительности сервера достигается при величине времени доступа $b = 0,908$, $S(b) = 43,768$ при $\Lambda = 0,5$, что соответствует графику на рис. 2.

Аналогично решению задачи для кластера из двух серверов задачу распределения можно решать в программном комплексе для n серверов при условии, что $n < 10$, так как с учетом по-

требностей в видео-конференц-связи на уровне крупных потребителей количество выделенных серверов редко превышает данные значения, а с учетом специфики учреждения образования и количества пользователей системы видео-конференц-связи – не превышает пяти. Ограничения в количестве серверов для кластера могут быть превышены, однако исходя из практики применения систем видео-конференц-связи, следует отметить, что у крупных компаний отсутствует «безграничный» подход к распределению вычислительных ресурсов. Так, зачастую оптимальнее иметь несколько высоконагруженных (более 60 % задействованных аппаратных ресурсов) серверов в кластере системы видео-конференц-связи, чем применять большое количество низконагруженных серверов.

Заключение. Для оптимального распределения больших потоков пользователей видеоконференции на базе кластерной архитектуры построения необходима доработка программного обеспечения балансировщика с учетом аппаратных составляющих серверов в кластере. Решение по распределению потока пользователей должно приниматься на основании загруженности всего кластера, что для системы видео-конференц-связи не всегда возможно ввиду специфики ее работы.

Доработанная математическая модель может быть адаптирована для веб-сервисов, вычислительных сервисов на базе грид-систем, а также для сервисов баз данных с учетом высоконагруженных аппаратных компонентов при работе данных сервисов и модификаций алгоритма распределения интенсивности входного потока запросов на системы.

Список использованных источников

1. Парамонов, А. И. Проблемы дистанционного образования и их прикладные решения в образовательных технологиях / А. И. Парамонов // Высшее техническое образование: проблемы и пути развития = Engineering education: challenges and developments : материалы X Междунар. науч.-метод. конф., Минск, 26 нояб. 2020 г. – Минск : БГУИР, 2020. – С. 182–187.
2. Марков, А. Н. Готовность учреждений высшего образования к цифровой трансформации процессов / А. Н. Марков, С. А. Мигалевич // Цифровая трансформация. – 2021. – № 2. – С. 64–68.
3. Марков, А. Н. Выбор сервиса видео-конференц-связи и его адаптация под учреждение образования / А. Н. Марков, Р. О. Игнатович, А. И. Парамонов // Информатика. – 2021. – Т. 18, № 4. – С. 17–25. <https://doi.org/10.37661/1816-0301-2021-18-4-17-25>
4. Lui, C. DNS and BIND / C. Lui, P. Albitz. – 5th ed. – O'Reilly Media, Inc., 2006. – 642 p.
5. Построение и исследование распределенной информационной системы веб-приложений и сервисов на базе кластерной архитектуры / В. А. Старых [и др.] // Качество. Инновации. Образование. – 2011. – № 12. – С. 91–99.
6. A survey of general-purpose computation on graphics hardware / J. D. Owens [et al.] // Computer Graphics Forum. – 2007. – Vol. 26, no. 1. – P. 80–113.
7. Dongarra, J. J. The LINPACK benchmark: Past, present and future / J. J. Dongarra, P. Luszczyk, A. Petitot // Concurrency and Computation: Practice and Experience. – 2003. – Vol. 15, no. 9. – P. 803–820.
8. High Performance Computing: Technology, Methods and Applications / J. Dongarra [et al.]. – Inc., 2015. – 444 p.
9. Dongarra, J. Adaptive linear solvers and eigensolvers / J. Dongarra // Argonne Training Program on Extreme-Scale Computing. – 2016. – Mode of access: https://extremecomputingtraining.anl.gov/files/2016/08/Dongara_830_AdaptiveLinear.pdf. – Date of access: 08.08.2022.

References

1. Paramonov A. I. *Distance education problems and their applied solutions in educational technologies*. Vysshee tehnikeskoe obrazovanie: problemy i puti razvitiya : materialy X Mezhdunarodnoj nauchno-metodicheskoy konferencii, Minsk, 26 nojabrja 2020 g. [*Engineering Education: Challenges and Developments : Materials of the X International Scientific and Methodological Conference, Minsk, 26 November 2020*]. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radiojelektroniki, 2020, pp. 182–187 (In Russ.).
2. Markov A. N., Migalevich S. A. *Readiness of higher education institutions for digital transformation processes*. Cifrovaja transformacija [*Digital Transformation*], 2021, no. 2, pp. 64–68 (In Russ.).

3. Markov A. N., Ihnatovich R. O., Paramonov A. I. *Choosing a video conferencing service and its adaptation for educational institution*. Informatika [Informatics], 2021, vol. 18, no. 4, pp. 17–25 (In Russ.). <https://doi.org/10.37661/1816-0301-2021-18-4-17-25>

4. Lui C., Albitz P. *DNS and BIND*, 5th edition. O'Reilly Media, Inc., 2006, 642 p.

5. Starykh V., Varyonov D., Ploskov S., Kuznetsov A. *Construction and research of a distributed information system of web applications and services based on cluster architecture*. Kachestvo. Innovacii. Obrazovanie [Quality. Innovation. Education], 2011, no. 12, pp. 91–99 (In Russ.).

6. Owens J. D., Luebke D., Govindaraju N., Harris M., Krüger J., ..., Purcell T. J. A survey of general-purpose computation on graphics hardware. *Computer Graphics Forum*, 2007, vol. 26, no. 1, pp. 80–113.

7. Dongarra J. J., Luszczek P., Petitet A. The LINPACK benchmark: Past, present and future. *Concurrency and Computation: Practice and Experience*, 2003, vol. 15, no. 9, pp. 803–820.

8. Dongarra J., Grandinetti L., Joubert G. R., Kowalik J. *High Performance Computing: Technology, Methods and Applications*. Inc., 2015, 444 p.

9. Dongarra J. Adaptive linear solvers and eigensolvers. *Argonne Training Program on Extreme-Scale Computing*, 2016. Available at: https://extremecomputingtraining.anl.gov/files/2016/08/Dongara_830_AdaptiveLinear.pdf (accessed 08.08.2022).

Информация об авторе

Марков Алексей Николаевич, аспирант, заместитель начальника Центра информатизации и инновационных разработок, Белорусский государственный университет информатики и радиоэлектроники.
ORCID <https://orcid.org/0000-0001-8508-1812>
E-mail: a.n.markov@bsuir.by

Information about the author

Aleksey N. Markov, Postgraduate Student, Deputy Head of the Center for Informatization and Innovative Development, Belarusian State University of Informatics and Radioelectronics.
ORCID <https://orcid.org/0000-0001-8508-1812>
E-mail: a.n.markov@bsuir.by