

# АЛГОРИТМ БАЛАНСИРОВКИ НАГРУЗКИ WEB-ПРИЛОЖЕНИЙ НА ОСНОВЕ ВРЕМЕНИ ОТВЕТА

В современных Web-приложениях одной из важных задач является балансировка нагрузки между набором серверов. При решении этой задачи встаёт вопрос выбора сервера, на который отправить запрос, эффективным решением этой задачи является алгоритм, основанный на времени ответа сервера. Экспериментально выведена зависимость между загруженностью сервера и временем ответа на запросы.

## ВВЕДЕНИЕ

Основой алгоритма балансировки нагрузки web-приложений является детектор отказов, разработанный в 2004 году группой японских ученых из института «Japan Advanced Institute of Science and Technology» [1].

### I. ДЕТЕКТОР ОТКАЗОВ

Детектор отказов – это система, которая отвечает за обнаружение сбоев в распределённой системе. Был выбран современный «начисляемый» детектор отказов. В этом детекторе отказов каждая система (за которыми «следит» детектор отказа) посылает детектору отказа сообщения, эти сообщения говорят о том, что система работает. Значение детектора отказа изменяется в зависимости от текущего состояния системы, и выражается следующей функцией (минимальные значения будут для систем, работающих без отказов):

$$\phi(t_{now}) = -\log_{10}(P_{later}(t_{now} - T_{last})),$$

где,  $T_{last}$  – время, когда было получено последнее сообщение от системы,  $t_{now}$  – текущее время,  $P_{later}$  – вероятность того, что система пришлёт новое сообщение в промежутке от  $T_{last}$  до  $t_{now}$ .

Эта вероятность определяется:

$$P_{later}(t) = \frac{1}{\sigma\sqrt{2\pi}} \int_t^{+\infty} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx = 1 - F(t),$$

где,  $F(t)$  – это кумулятивная функция нормального распределения с математическим ожиданием  $\mu$  и дисперсией  $\sigma^2$ .

Пусть  $\Phi(x)$  будет кумулятивной функцией стандартного нормального распределения  $N(0, 1)$ :

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{x^2}{2}} dx,$$

тогда:

$$F(t) = \Phi(t - \mu\sigma)$$

*Бонда Павел Сергеевич*, магистрант кафедры информационных технологий автоматизированных систем Белорусского государственного университета информатики и радиоэлектроники, [pbondawork@gmail.com](mailto:pbondawork@gmail.com).

*Научный руководитель: Муха Владимир Степанович*, профессор кафедры информационных технологий автоматизированных систем Белорусского государственного университета информатики и радиоэлектроники, доктор технических наук, профессор, [mukha@bsuir.by](mailto:mukha@bsuir.by).

Интуитивно, вычисление значения  $P_{later}(t_{now} - T_{last})$  эквивалентно ответу на вопрос: «имея интервалы получения предыдущих сообщений, и то, что, последнее сообщение было получено в момент времени  $T_{last}$ , какова вероятность получения следующего сообщения, учитывая что прошло  $t_{now} - T_{last}$  времени?»

Данный детектор отказов является основой алгоритма балансировки нагрузки.

### II. БАЛАНСИРОВКА НАГРУЗКИ

Для того чтобы применить данный детектор отказов для балансировки нагрузки, заменим интервалы между сообщениями, которые система посылает детектору отказа, на интервалы ответа сервера измеряемые балансировщиком. Вес сервера будет обратно пропорционален значению изменённого детектора отказа. Сервер с балансировщик выбирает сервера с максимальным весом и отправляет запросы на них.

При этом возникает проблема, если серверу в соответствии с алгоритмом назначен очень низкий вес (по какой-то причине запрос обрабатывался очень долго), новые запросы не будут отправляться на этот сервер, даже если он восстановил работоспособность. Для того чтобы избежать данной ситуации, введём минимальный вес сервера. Значение веса сервера не может опуститься ниже минимального веса.

### III. ВЫВОДЫ

Предлагаемый алгоритм позволяет эффективно балансировать нагрузку, при этом он прост в реализации и универсален.

### Список литературы

1. The /spl phi/ accrual failure detector [Electronic resource] / N. Hayashibara. – School of Information Science, 2004. – Mode of access: <https://ieeexplore.ieee.org/document/1353004>. – Date of access: 12.04.2021.