Shoe Shine Shop

Блох Александр, Гмыря Михаил, Миронов Сергей, Харчиков Игорь

17 марта 2023 г.

1 Математическая модель

Система состоит из двух серверов, интенсивность обслуживания первого сервера – μ_1, интенсивность обслуживания второго – μ_2. Требования поступают в систему с интенсивностью λ. Каждое требование обслуживается по следующей логике: сначала требование обслуживается на сервере 1, затем, если сервер 2 свободен, требование обслуживается на нем. В противном же случае, если сервер 2 занят, требование «ждет», пока другое требование обслужится на сервере 2, причем сервер 1 в это время считается занятым, и другие требования поступать на него не могут. Схема системы - Рис. 1. Введем марковскую цепь на вероятностном пространстве $(\Omega, F, P(.))$.

1.1 Множество значений

$X_i \in S\ \forall i$, где S состоит из следующих состояний:

$(0,0)$ - состояние, когда оба сервера никого не обслуживают

$(1,0)$ - состояние, когда первый сервер обслуживает требование, второй свободен

$(0,1)$ - состояние, когда первый сервер свободен, второй обслуживает

$(1,1)$ - состояние, когда оба сервера обслуживают требования

$(b,1)$ - состояние, когда второй сервер обслуживает, а первый закончил обслуживание. Однако первый "заблокирован"для поступления нового требования

1.2 Начальное распределение

$p = (1,0,0,0,0)$ или же $P(X_0 = (0,0)) = 1$

1.3 Матрица переходов

Вычислим все вероятности перехода, не равные 0.

$P(X_i = (1,0)|X_{i-1} = (0,0)) = 1$ -

$P(X_i = (0,1)|X_{i-1} = (1,0)) = 1$ - так как второй сервер свободен, после обслуживания на первом сервере требование сразу же переходит на обслуживание на втором сервере, освобождая первый

$P(X_i = (0,1)|X_{i-1} = (b,1)) = 1$ - первое требование обслужилось и ждет, пока освободится второй сервер. Как только он освобождается, требование сразу начинает обслуживаться на втором сервера, освобождая первый

$P(X_i = (1,1)|X_{i-1} = (0,1))$ - это вероятность того, что новое требование придет на первый сервер раньше, чем предыдущее требование обслужится на втором сервере.

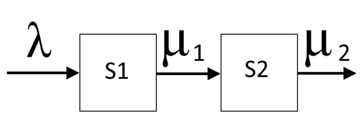

Рис. 1: Схема системы.

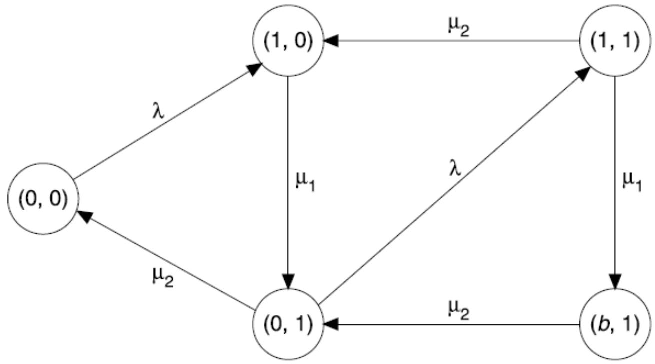

Рис. 2: Диаграмма цепи Маркова для процесса.

Обозначим случайную величину, обозначающую время между приходом двух требований за $Z \sim exp(\lambda)$, а время обслуживания требования на втором сервере за $Y \sim exp(\mu_2)$. Тогда, чтобый найти вероятность $(*)$, нужно вычислить следующую вероятность:

$P(Z < Y) = \int_0^{+\infty} P(Z < Y|Y = y) * \mu_2 e^{-y\mu_2}\, dy = \int_0^{+\infty} (1 - e^{-y\lambda}) * \mu_2 e^{-y\mu_2}\, dy = \frac{\lambda}{\lambda+\mu_2}$

Таким образом,

$P(X_i = (1,1)|X_{i-1} = (0,1)) = \frac{\lambda}{\lambda+\mu_2}$

и

$P(X_i = (0,0)|X_{i-1} = (0,1)) = 1 - \frac{\lambda}{\lambda+\mu_2} = \frac{\mu_2}{\lambda+\mu_2}$

$P(X_i = (b,1)|X_{i-1} = (1,1))$ - это вероятность того, что первый сервер закончит обслуживание требования до того, как закончит второй. $(**)$

Обозначим случайную величину, обозначающую время обслуживания на сервере 1 за $Z \sim exp(\mu_1)$, а время обслуживания требования на втором сервере за $Y \sim exp(\mu_2)$. Тогда чтобый найти вероятности нужно вычислить следующую вероятность:

$P(Z < Y) = \int_0^{+\infty} P(Z < Y|Y = y) * \mu_2 e^{-y\mu_2}\, dy = \int_0^{+\infty} (1 - e^{-y\mu_1}) * \mu_2 e^{-y\mu_2}\, dy = \frac{\mu_1}{\mu_1+\mu_2}$

Таким образом,

$P(X_i = (b,1)|X_{i-1} = (1,1)) = \frac{\mu_1}{\mu_1+\mu_2}$

и

$P(X_i = (1,0)|X_{i-1} = (1,1)) = 1 - \frac{\mu_1}{\mu_1+\mu_2} = \frac{\mu_2}{\mu_1+\mu_2}$

Теперь можем составить матрицу переходных вероятностей:

$$\mathbb{P} = \begin{array}{c} \\ (0,0) \\ (1,0) \\ (0,1) \\ (1,1) \\ (b,1) \end{array} \begin{array}{c} \begin{array}{ccccc} (0,0) & (1,0) & (0,1) & (1,1) & (b,1) \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \frac{\mu_2}{\lambda+\mu_2} & 0 & 0 & \frac{\lambda}{\lambda+\mu_2} & 0 \\ 0 & \frac{\mu_2}{\mu_1+\mu_2} & 0 & 0 & \frac{\mu_1}{\mu_1+\mu_2} \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

2 Вычисления

2.1 Уравнения баланса и система

Для начала стоит отметить, что в данной модели не хватит лишь количества человек в системе для каждого состояния. Поскольку у нас 2 обрабатывающих сервера, то состояния нужно учитывать для обоих. Это возможные 5 состояний системы:

- (0,0) - оба сервера ничего не обратбатывают
- (1,0) - первый сервер обрабатывает запрос, второй ожидает
- (0,1) - второй сервер обрабатывает запрос, первый ожидает

- (1,1) - оба сервера обрабатывают запросы
- (b,1) - первый сервер уже обработал и не может отправить запрос второму, пока тот обрабатывает другой запрос

Исходя из того, что интенсивность прихода равна λ, и интенсивность обработки запросов на первом и втором серверах соответственно μ_1 и μ_2, то получается следующая Марковская цепь состояний нашей модели(Рис. 2).

Следовательно, теперь можем расписать входной и выходной поток для каждого состояния :

- (0,0) : $\lambda * \pi_{00} = \mu_2 * \pi_{01}$
- (1,0) : $\mu_1 * \pi_{10} = \lambda * \pi_{00} + \mu_2 * \pi_{11}$
- (0,1) : $(\mu_2 + \lambda) * \pi_{01} = \mu_1 * \pi_{10} + \mu_2 * \pi_{b1}$
- (1,1) : $(\mu_1 + \mu_2) * \pi_{11} = \lambda * \pi_{01}$
- (b,1) : $\mu_1 * \pi_{11} = \mu_2 * \pi_{b1}$

2.2 Предельные вероятности

Выполняя линейные преобразования с системой, получаем следующие значения:

$$\pi^* = \pi_{00} * \left(1, \frac{\lambda}{\mu_1} + \frac{\lambda^2}{(\mu_1 + \mu_2) * \mu_1}, \frac{\lambda}{\mu_2}, \frac{\lambda^2}{(\mu_1 + \mu_2) * \mu_2}, \frac{\mu_1 * \lambda^2}{\mu_2^2 * (\mu_1 + \mu_2)}\right) \tag{1}$$

Где

$$\pi_{00} = \frac{(\mu_1 + \mu_2) * \mu_2^2 * \mu_1}{\mu_1 * \mu_2^2(\mu_1 + \mu_2) + \lambda * (\mu_1 + \mu_2) * \mu_2^2 + \lambda^2 * \mu_2^2 + \lambda * (\mu_1 + \mu_2) * \mu_1 * \mu_2 + \lambda * \mu_1 * \mu_2 + \mu_1^2 * \lambda^2} \tag{2}$$

2.3 Время ожидания и количество человек в системе

Количество человек:

$$L = P_{01} + P_{10} + 2(P_{11} + P_{b1}) \tag{3}$$

Время ожидания:

$$W = \frac{P_{01} + P_{10} + 2(P_{11} + P_{b1})}{\lambda * (P_{00} + P_{01})} \tag{4}$$

3 Имитационная модель и эксперименты

3.1 Имитационная модель

Мы написали на языке Python симуляцию работы данной системы массового обслуживания. Для реализации мы использовали библиотеку Threading для одновременного отслеживания нескольких клиентов. По ходу выполнения симуляции мы засекаем время смены состояний. Далее таймером замеряются интервалы смены состояний, на их основе вычислены эмпирические аналоги предельных вероятностей. Так же, собрана статистика по количеству человек в системе в моменты смены состояний, и, следуя из нее, время в системе.

3.2 Эксперименты

Для проверки эксперемминтальных выводов, применяются выведенные ранее формулы-следствия уравнений баланса(2, 3, 4) (так же запрограммированы на языке Python).

3.2.1 $\lambda = 1, \mu_1 = 2.5, \mu_2 = 2$

Теоретические значения:

- $\pi_{00} = 0.447$
- $\pi_{10} = 0.218$
- $\pi_{01} = 0.223$
- $\pi_{11} = 0.050$
- $\pi_{b1} = 0.062$

$L = 0.665$
$W = 0.993$

Эксперементальные значения:

- $\hat{\pi}_{00} = 0.446$
- $\hat{\pi}_{10} = 0.226$
- $\hat{\pi}_{01} = 0.225$
- $\hat{\pi}_{11} = 0.045$
- $\hat{\pi}_{b1} = 0.0.58$

$\hat{L} = 0.945$
$\hat{W} = 1.001$

3.2.2 $\lambda = 2, \mu_1 = 1, \mu_2 = 1.5$

Теоретические значения:

- $\pi_{00} = 0.130$
- $\pi_{10} = 0.467$
- $\pi_{01} = 0.173$
- $\pi_{11} = 0.138$
- $\pi_{b1} = 0.092$

$L = 1.101$
$W = 1.819$

Эксперементальные значения:

- $\hat{\pi}_{00} = 0.140$
- $\hat{\pi}_{10} = 0.484$
- $\hat{\pi}_{01} = 0.165$
- $\hat{\pi}_{11} = 0.121$
- $\hat{\pi}_{b1} = 0.089$

$\hat{L} = 1.093$
$\hat{W} = 1.753$

3.2.3 $\lambda = 2.5, \mu_1 = 2, \mu_2 = 1$

Теоретические значения:

- $\pi_{00} = 0.083$
- $\pi_{10} = 0.190$
- $\pi_{01} = 0.208$
- $\pi_{11} = 0.173$
- $\pi_{b1} = 0.346$

$L = 1.436$
$W = 1.976$

Эксперементальные значения:

- $\hat{\pi}_{00} = 0.093$
- $\hat{\pi}_{10} = 0.174$
- $\hat{\pi}_{01} = 0.203$
- $\hat{\pi}_{11} = 0.172$
- $\hat{\pi}_{b1} = 0.358$

$\hat{L} = 1.248$
$\hat{W} = 2.018$

3.2.4 $\lambda = 1, \mu_1 = 2, \mu_2 = 3$

Теоретические значения:

- $\pi_{00} = 0.489$
- $\pi_{10} = 0.293$
- $\pi_{01} = 0.163$
- $\pi_{11} = 0.033$
- $\pi_{b1} = 0.022$

$L = 0.565$
$W = 0.867$

Эксперементальные значения:

- $\hat{\pi}_{00} = 0.460$
- $\hat{\pi}_{10} = 0.314$
- $\hat{\pi}_{01} = 0.170$
- $\hat{\pi}_{11} = 0.035$
- $\hat{\pi}_{b1} = 0.022$

$\hat{L} = 0.870$
$\hat{W} = 0.945$

4 Код

Ссылка на репозиторий: https://github.com/a13xbb/shoe-shine-shop