Лабораторная работа 3. Моделирование стохастических процессов

В данном разделе использованы материалы [2].

3.1. Предварительные сведения. СМО $M|M|1$

$M|M|1$ — однолинейная СМО с накопителем бесконечной ёмкости. Поступающий поток заявок — пуассоновский с интенсивностью λ. Времена обслуживания заявок — независимые в совокупности случайные величины, распределённые по экспоненциальному закону с параметром μ.

Система дифференциальных уравнений Колмогорова:

$$\begin{cases} p_0'(t) = -\lambda p_0(t) + \mu p_1(t), \\ p_i'(t) = -(\lambda+\mu) p_i(t) + \lambda p_{i-1}(t) + \mu p_{i+1}(t), \quad i \geqslant 1, \end{cases}$$

$p_i(t) = P\{\nu(t) = i\}$ — вероятность того, что в момент времени t в системе находится i заявок.

Стационарные вероятности удовлетворяют СУР:

$$\begin{cases} 0 = -\lambda p_0 + \mu p_1, \\ 0 = -(\lambda+\mu) p_i + \lambda p_{i-1} + \mu p_{i+1}, \quad i \geqslant 1. \end{cases} \tag{3.1}$$

Уравнение локального баланса:

$$\lambda p_i = \mu p_{i+1}, \quad i \geqslant 0,$$

с условием нормировки: $\sum\limits_{i=0}^{\infty} p_i = 1$.

Решение уравнения (3.1):

$$\begin{cases} p_i = p_0 \rho^i, \quad i \geqslant 0, \quad \rho = \dfrac{\lambda}{\mu}, \\ p_0 = \dfrac{1}{\sum\limits_{i=0}^{\infty} \rho^i} = \left[\dfrac{1}{1-\rho}\right]^{-1} = 1-\rho, \quad \rho < 1 \end{cases}$$

Окончательно стационарное распределение числа заявок:

$$p_i = (1-\rho)\rho^i, \quad \rho < 1,$$

$\rho = \frac{\lambda}{\mu}$ — загрузка системы.

Характеристики системы:

Коэффициент использования системы: $u = 1 - p_0 = \rho$.

Стационарное среднее число заявок в системе: $N = \sum\limits_{i=0}^{\infty} i p_i = \dfrac{\rho}{1-\rho}$.

Стационарное среднее число заявок в очереди:

$$Q = \sum_{i=0}^{\infty}(i-1)p_i = \frac{\rho^2}{1-\rho}.$$

Стационарное среднее время ожидания начала обслуживания:

$$w = \frac{\rho}{\mu(1-\rho)}.$$

Стационарное среднее время пребывания заявки в системе:

$$v = \frac{1}{\mu(1-\rho)}.$$

3.2. Предварительные сведения. СМО $M|M|n|R$

$M|M|n|R$ — однолинейная СМО с накопителем конечной ёмкости R. Поступающий поток заявок — пуассоновский с интенсивностью λ. Времена обслуживания заявок — независимые в совокупности случайные величины, распределённые по экспоненциальному закону с параметром μ.

Система дифференциальных уравнений Колмогорова:

$$\begin{cases} p_0'(t) = -\lambda p_0(t) + \mu p_1(t), \\ p_i'(t) = -(\lambda + i\mu)p_i(t) + \lambda p_{i-1}(t) + (i+1)\mu p_{i+1}(t), \quad i = \overline{1, n-1}, \\ p_i'(t) = -(\lambda + n\mu)p_i(t) + \lambda p_{i-1}(t) + n\mu p_{i+1}(t), \quad i = \overline{n, n+R-1}, \\ p_{n+R}'(t) = -n\mu p_{n+R}(t) + \lambda p_{n+R-1}(t), \end{cases}$$

$p_i(t) = P\{\nu(t) = i\}$ — вероятность того, что в момент времени t в системе находится i заявок.

Стационарные вероятности удовлетворяют СУР:

$$\begin{cases} 0 = -\lambda p_0 + \mu p_1, \\ 0 = -(\lambda + i\mu)p_i + \lambda p_{i-1} + (i+1)\mu p_{i+1}, \quad i = \overline{1, n-1}, \\ 0 = -(\lambda + n\mu)p_i + \lambda p_{i-1} + n\mu p_{i+1}, \quad i = \overline{n, n+R-1}, \\ 0 = -n\mu p_{n+R} + \lambda p_{n+R-1}. \end{cases} \tag{3.2}$$

Уравнения локального баланса:

$$\lambda p_i = (i+1)\mu p_{i+1}, \quad i = \overline{1, n-1},$$
$$\lambda p_i = n\mu p_{i+1}, \quad i = \overline{n, n+R-1}$$

с условием нормировки: $\sum\limits_{i=0}^{\infty} p_i = 1$.

Решение уравнения (3.2):

$$\begin{cases} p_i = \dfrac{\rho^i}{i!} p_0, \quad i = \overline{1, n-1}, \\ p_i = \dfrac{\rho^i}{n! n^{i-n}} p_0, \quad i = \overline{n, n+R}, \\ p_0 = \left[\sum\limits_{i=0}^{n-1} \dfrac{\rho^i}{i!} + \dfrac{\rho^n}{n!} \dfrac{1 - \left(\frac{\rho}{n}\right)^{R+1}}{1 - \frac{\rho}{n}} \right]^{-1}, \end{cases}$$

$\rho = \dfrac{\lambda}{\mu}$ — загрузка системы.

При $n = 1$: $p_i = \dfrac{1-\rho}{1-\rho^{R+2}} \rho^i, \quad i = \overline{0, R+1}$.

Характеристики системы: Стационарная вероятность немедленного обслуживания заявки: $P_{w=0} = p_0 \sum\limits_{i=0}^{n-1} \dfrac{\rho^i}{i!}$.

Стационарная вероятность потери заявки: $\pi = p_{n+R} = \dfrac{\rho^{n+R}}{n! n^R} p_0$.

Стационарное среднее число заявок в очереди:

$$Q = \sum_{i=1}^{R} i p_{n+i} = \frac{\rho^{n-1}}{(n-1)!} \frac{1 + r\left(\frac{\rho}{n}\right)^{R+1} - (R+1)\left(\frac{\rho}{n}\right)^R}{\left(\frac{n}{\rho} - 1\right)^2} p_0.$$

3.3. Реализация модели на NS-2

```
# создание объекта  Simulator
set ns [new Simulator]

# открытие на запись файла out.tr для регистрации событий
set tf [open out.tr w]
$ns trace-all $tf

# задаём значения параметров системы
set lambda 30.0
set mu     33.0

# размер очереди для M|M|1 (для M|M|1|R: set qsize R)
set qsize     100000

# устанавливаем длительность эксперимента
set duration  1000.0

# задаём узлы и соединяем их симплексным соединением
# с полосой пропускания 100 Кб/с и задержкой 0 мс,
# очередью с обслуживанием типа  DropTail
set n1 [$ns node]
set n2 [$ns node]
```

```
set link [$ns simplex-link $n1 $n2 100kb 0ms DropTail]

# наложение ограничения на размер очереди:
$ns queue-limit $n1 $n2 $qsize

# задаём распределения интервалов времени
# поступления пакетов и размера пакетов
set InterArrivalTime [new RandomVariable/Exponential]
$InterArrivalTime set avg_ [expr 1/$lambda]
set pktSize [new RandomVariable/Exponential]
$pktSize set avg_ [expr 100000.0/(8*$mu)]

# задаём агент UDP и присоединяем его к источнику,
# задаём размер пакета
set src [new Agent/UDP]
$src set packetSize_ 100000
$ns attach-agent $n1 $src

# задаём агент-приёмник и присоединяем его
set sink [new Agent/Null]
$ns attach-agent $n2 $sink
$ns connect $src $sink

# мониторинг очереди
set qmon [$ns monitor-queue $n1 $n2 [open qm.out w] 0.1]
$link queue-sample-timeout

# процедура finish закрывает файлы трассировки
proc finish {} {
  global ns tf
  $ns flush-trace
  close $tf
  exit 0
}

# процедура случайного генерирования пакетов
proc sendpacket {} {
  global ns src InterArrivalTime pktSize
  set time [$ns now]
  $ns at [expr $time +[$InterArrivalTime value]] "sendpacket"
  set bytes [expr round ([$pktSize value])]
  $src send $bytes
}

# планировщик событий
$ns at 0.0001 "sendpacket"
$ns at $duration "finish"

# расчет загрузки системы и вероятности потери пакетов
set rho [expr $lambda/$mu]
set ploss
 [expr (1-$rho)*pow($rho,$qsize)/(1-pow($rho,($qsize+1)))]
puts "Теоретическая вероятность потери = $ploss"
```

```
set aveq [expr $rho*$rho/(1-$rho)]
puts "Теоретическая средняя длина очереди = $aveq"

# запуск модели
$ns run
```

3.4. График в GNUplot

В каталоге с проектом создайте отдельный файл, например, `graph_plot`:

```
touch graph_plot
```

Откройте его на редактирование и добавьте следующий код, обращая внимание на синтаксис GNUplot:

```
#!/usr/bin/gnuplot -persist

# задаём текстовую кодировку,
# тип терминала, тип и размер шрифта
set encoding utf8
set term pdfcairo font "Arial,9"

# задаём выходной файл графика
set out 'qm.pdf'

# задаём название графика
set title "График средней длины очереди"

# задаём стиль линии
set style line 2

# подписи осей графика
set xlabel "t"
set ylabel "Пакеты"

# построение графика, используя значения
# 1-го и 5-го столбцов файла qm.out
plot "qm.out" using ($1):($5) with lines
        title "Размер очереди (в пакетах)",\
     "qm.out" using ($1):($5) smooth csplines
        title " Приближение сплайном ", \
     "qm.out" using ($1):($5) smooth bezier
        title " Приближение Безье "
```

Сделайте файл исполняемым. После компиляции файла с проектом, запустите скрипт в созданном файле `graph_plot`, который создаст файл `qm.pdf` с результатами моделирования (рис. 3.1).

Рис. 3.1. График поведения длины очереди