

# Анализ математической модели планировщика WFQ

**Бурлаков Олег Максимович**

*Академия Федеральной службы охраны Российской Федерации г. Орел  
сотрудник*

УДК 004.71

## Аннотация

Данная статья посвящена анализу существующих планировщиков обслуживания пакетов. Подробно рассмотрена математическая модель планировщика, работающего по принципу взвешенного справедливо обслуживания (*Weighed Fair Queuing – WFQ*), а также предложены пути решения задачи оптимизации процесса моделирования планировщика WFQ.

**Ключевые слова:** планировщик WFQ, качество обслуживания, справедливое обслуживание.

## ВВЕДЕНИЕ

В условиях роста потребности в передачи разнородного трафика в больших объемах, более предпочтительным становится использования транспортных сетей с коммутацией пакетов (далее - ТСКП), т.к. данные сети обладают рядом преимуществ по сравнению с транспортными сетями с коммутацией каналов.

Но для использования ТСКП необходимо решить задачи обеспечения и контроля требуемого качества передачи данных. Для решения данного рода задач следует использовать математическое моделирование планировщиков обслуживания пакетов.

## ОБРАБОТКА ПАКЕТОВ В МАРШРУТИЗАТОРЕ

Устройством взаимодействия элементов сети с коммутацией пакетов является маршрутизатор, функционирующим на сетевом уровне эталонной модели взаимодействия открытых систем, предназначенным для приема, обработки и передачи пакетов. Процесс обработки пакетов можно представить с помощью модели на рисунке 1.

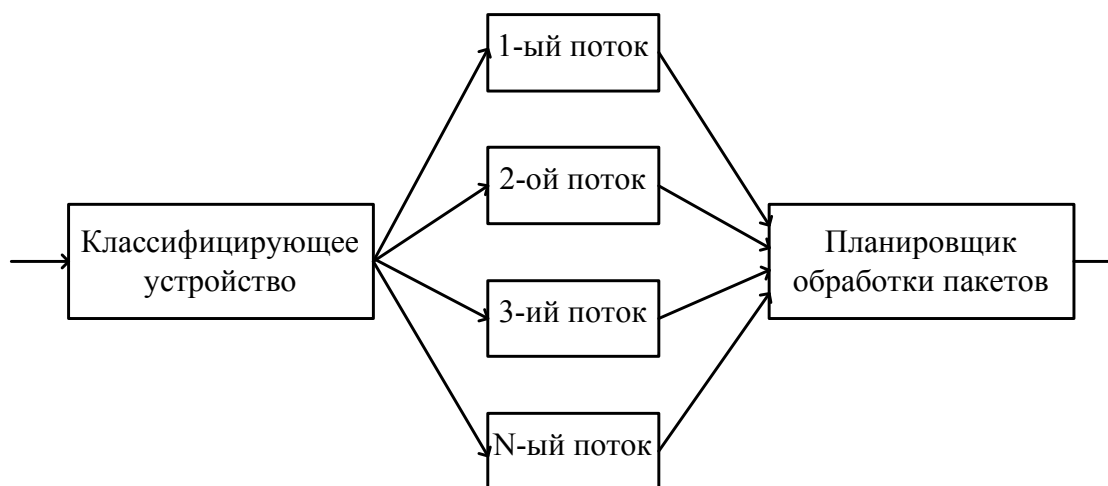


Рисунок 1. Модель процесса обработки пакетов в маршрутизаторе.

Рассмотрим такое устройство как планировщик обработки пакетов. Данное устройство выполняет функцию определения очереди обработки пакетов по какому-либо классифицирующему признаку, определенному классифицирующим устройством. Наиболее простым в реализации будет являться планировщик FIFO (First In First Out). В нем пакеты обрабатываются в соответствии с очередью прибытия. Но при таком виде обслуживания трафик, рассортированный по потокам будет обслужен с одинаковым качеством, которое может не удовлетворять требованиям определенному виду трафика. Планировщик приоритетного обслуживания PQ (Priority Queuing) реализует последовательную обработку пакетов из очереди с высшим приоритетом, затем из очереди с низшим приоритетом. Такой подход позволяет обслуживать потоки с разным качеством, но также имеет свои недостатки, связанные с ухудшением качества низкоприоритетного трафика при увеличении количества высокоприоритетного трафика. Планировщик FQ (Fair Queuing) осуществляет обработку пакетов из каждого потока поочередно. С помощью FQ устранен недостаток, при котором негативное влияние, при нарушении соглашения о качестве обслуживания, не оказывается на другие потоки. Но данный метод не пригоден при необходимости дифференциации требований к качеству для разных потоков. Планировщик WFQ устраняет некоторые недостатки

планировщика FQ. Для каждого потока определяется полоса пропускания с учетом весового коэффициента, назначенного сетевым администратором. Таким образом становится возможность обслуживания различных потоков с различными требованиями к качеству передачи. Однако данный алгоритм имеет высокую вычислительную сложность, что является основной проблемой при его использовании.

## МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ПЛАНИРОВЩИКА WFQ

В данной модели используется:

- $N$  потоков с интенсивностью  $\lambda_i$ , где  $i = 1, 2, 3, \dots, N$ ;
- Общий буфер  $K$ , определяющийся как сумма максимальных очередей каждого потока;
- Весовой коэффициент  $W_i$ , где  $i = 1, 2, 3, \dots, N$ , назначенный каждому потоку.

На рисунке 2 для исследуемой модели показаны пространство состояний и соответствующая диаграмма переходов, при  $N = 2$  и  $K = 3$ .

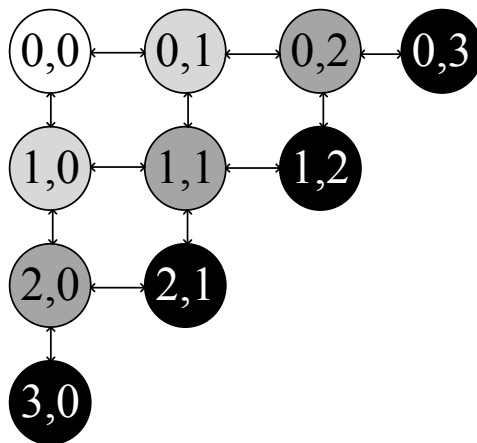


Рисунок 2. Пространство макросостояний и диаграмма переходов, описывающее планировщик WFQ.

Вероятность нахождения системы в каком-либо состоянии обозначим  $P_{i,j}$ , где  $i$  и  $j$  – количество пакетов в первой и второй очереди соответственно. Для расчета вероятностей  $P_{i,j}$ , необходимо составить и решить систему уравнений

статистического равновесия [1]. Для этого необходимо составить матрицу  $Q$ , имеющую вид, представленный на рисунке 3.

	$\pi_{00}$	$\pi_{01}$	$\pi_{10}$	$\pi_{02}$	$\pi_{11}$	$\pi_{20}$	$\pi_{03}$	$\pi_{12}$	$\pi_{21}$	$\pi_{30}$
$\pi_{00}$	$-(\lambda_0 + \lambda_1)$ $B_0$	$\lambda_1$ $A_1$	$\lambda_0$							
$\pi_{01}$	$\mu_1$	$-(\lambda_0 + \lambda_1 + \mu_0)$		$\lambda_1$	$\lambda_0$					
$\pi_{10}$	$\mu_0$ $C_0$		$-(\lambda_0 + \lambda_1 + \mu_1)$ $B_1$		$\lambda_1$	$\lambda_0$				
$\pi_{02}$		$\mu_1$		$-(\lambda_0 + \lambda_1 + \mu_1)$			$\lambda_1$	$\lambda_0$		
$\pi_{11}$		$w_0\mu_0$	$w_1\mu_1$		$-(\lambda_0 + \lambda_1 + w_0\mu_0 + w_1\mu_1)$			$\lambda_1$	$\lambda_0$	
$\pi_{20}$			$\mu_0$ $C_1$			$-(\lambda_0 + \lambda_1 + \mu_0)$ $B_2$			$\lambda_1$	$\lambda_0$
$\pi_{03}$				$\mu_1$			$-\mu_1$ $A_3$			
$\pi_{12}$				$w_0\mu_0$	$w_1\mu_1$			$-(w_0\mu_0 + w_1\mu_1)$		
$\pi_{21}$					$w_0\mu_0$	$w_1\mu_1$			$-(w_0\mu_0 + w_1\mu_1)$	
$\pi_{30}$						$\mu_0$ $C_2$				$-\mu_0$ $B_3$

Рисунок 3. Матрица  $Q$ , описывающая планировщик WFQ.

Матрицы главной диагонали  $B$  описывают переход системы из одного состояния системы в смежное с одинаковым количеством пакетов в очередях, матрицы  $A$  описывают переход системы из одного состояния системы в смежное с большим количеством пакетов в очередях, матрицы  $C$  описывают переход системы из одного состояния системы в смежное с меньшим количеством пакетов в очередях.

Существует такая матрица  $R$ , что  $P_n = P_{n-1}R_n$ , где  $P_n$  – это вектор, элементами которого являются  $P_{i,j}$ , находящиеся в одной области с одинаковым количеством пакетов в очередях. Данный алгоритм был реализован с помощью

программной среды Python. Необходимо отметить основной недостаток данного алгоритма, такой как вычислительная сложность, возникшая из-за присутствия матриц больших размерностей при большом количестве потоков и большом объеме буфера.

## СПОСОБЫ ОПТИМИЗАЦИИ АЛГОРИТМА РЕАЛИЗАЦИИ МАТЕМАТИЧЕСКОЙ МОДЕЛИ ПЛНИРОВЩИКА WFQ

При анализе алгоритма, было выявлено, что основную вычислительную нагрузку создают матричные операции и большое число вложенных циклов, а также наличие рекурсивной функции. Отсюда можно сделать вывод, что для оптимизации алгоритма по скорости работы следует ограничить число вложенных циклов путем модернизации программного кода. Но произвести данную модернизацию не представляется возможным из-за особенностей самого алгоритма расчета. Решением задачи может стать распараллеливание процесса вычисления независимых функций, а именно расчета матриц  $A$  и матриц  $C$ , на различные ядра процессора. Такой подход позволит выиграть по скорости расчета модели. Избавится от рекурсивного алгоритма нахождения матрицы  $R$  можно, производя расчет матриц  $A$ ,  $B$  и  $C$  для одного состояния в области макросостояний с одинаковым числом пакетов в очередях, и последующим его обнулением, что так же даст выигрыш по занимаемой памяти.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Теория телетрафика. Часть 1. Модели и методы теории телетрафика : пособие [Электронный ресурс] / И. А. Саитов, А. Е. Миронов, А. В. Королев, Р. Б. Трегубов, А. Н. Орешин. – Орел : Академия ФСО России, 2013. – 1 электрон. опт. диск.
2. Степанов С.Н. Основы телетрафика мультисервисных сетей. – М.: Эко-Трендз, 2010. – 392 с.: ил.
3. William J. Stewart. The Matrix Geometric/Analytic Methods for Structured Markov Chains. - Department of Computer Science. 177 с.