

3.2. Основы теории стохастических сетей массового обслуживания

В этом подразделе приведены основные выкладки из теории систем массового обслуживания, которые будут применены в предложенных в учебном пособии моделях. Рассмотрены три вида систем массового обслуживания (СМО). К первому относятся классические СМО с неограниченной очередью типа $M/M/1$ и $M/M/m$ [1], причём первая буква M означает, что поток заявок на входе СМО является простейшим. Вторая буква M означает, что времена обслуживания в приборах распределены по экспоненциальному закону, третья показывает количество обслуживающих приборов в СМО. Вторым видом – СМО типа $M/M/1/K$, где K означает конечность длины очереди. К третьему виду относятся СМО с приоритетным обслуживанием заявок. Кроме того здесь приведены основные параметры и характеристики стохастических сетей массового обслуживания.

3.2.1. Параметры стохастических сетей

Рассмотрим произвольную разомкнутую сеть массового обслуживания, содержащую n систем обслуживания (СМО) S_1, S_2, \dots, S_n , и единственный источник заявок, из которого они с вероятностью P_{0i} могут поступать в любую из СМО (рис.3.1).

Внутри сети заявки перемещаются случайным образом и могут покинуть её из любой СМО с вероятностью P_{i0} . Такие сети называются стохастическими.

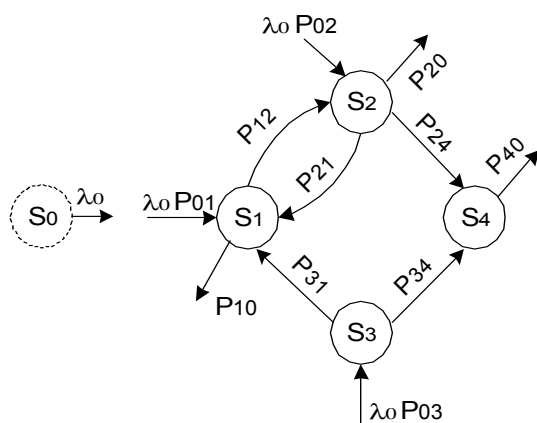


Рис. 3.1. Пример разомкнутой сети

Стохастическая сеть может быть задана следующими параметрами:

- числом n СМО в сети (S_0, S_1, \dots, S_n) , где S_0 – фиктивная СМО, моделирующая источник заявок, которая рассматривается как СМО с бесконечным числом заявок и с интенсивностью их обслуживания λ_0 ;
- числом каналов k_i в каждой СМО сети (k_1, k_2, \dots, k_n) ;
- интенсивностью потока заявок λ_0 источника S_0 ;
- интенсивностью потока на входе i -й СМО λ_i ;
- средним временем обслуживания заявок в каждой СМО сети (v_1, v_2, \dots, v_n) ;
- матрицей вероятности передач $P=[p_{ij}]$, где p_{ij} – вероятность того, что заявка покидающая СМО S_i , поступит на обслуживание в СМО S_j .

Для сети, состоящей из n СМО, матрица будет иметь следующий вид:

$$P = [p_{ij}] = \begin{matrix} & \begin{matrix} S_0 & S_1 & \dots & S_n \end{matrix} \\ \begin{matrix} S_0 \\ S_1 \\ \dots \\ S_n \end{matrix} & \begin{bmatrix} p_{00} & p_{01} & \dots & p_{0n} \\ p_{10} & p_{11} & \dots & p_{1n} \\ \dots & \dots & \dots & \dots \\ p_{n0} & p_{n1} & \dots & p_{nn} \end{bmatrix} \end{matrix}$$

Представленная матрица является стохастической. Для такой матрицы должно выполняться условие $\sum_{j=0}^n p_{ij} = 1$, которое означает, что сумма элементов каждой строки матрицы равна единице.

Условие стохастической матрицы вытекает из того, что заявка, выходящая из S_i , обязательно попадает в некоторую СМО S_j .

Поскольку заявку из СМО S_i поступают в СМО S_j с вероятностью p_{ij} , то интенсивность потока, поступающего из S_i в S_j равна $\lambda_i p_{ij}$, где λ_i – интенсивность потока, выходящего из СМО S_i . Следовательно, с учётом всех источников на входе СМО S_j интенсивность будет определяться выражением

$$\lambda_j = \sum_{i=0}^n \lambda_i p_{ij} . \quad (3.1)$$

Коэффициент передачи в i -ю СМО $\alpha_i = \lambda_i / \lambda_0$ определяет число этапов обслуживания в i -й СМО, которое получит каждая заявка, поступающая на вход сети из источника моделируемого СМО.

3.2.2. Характеристики СМО с неограниченной длиной очереди.

Если заданы параметры сети, то можно определить следующие характеристики каждой СМО и сети в целом:

- среднюю длину очереди заявок в i -й СМО - l_i и в сети - L ;
- среднее число заявок, пребывающих в i -й СМО - m_i и в сети - M ;
- среднее время ожидания обслуживания заявки i -й СМО ω_i и в сети - W ;
- среднее время пребывания заявки в i -й СМО - u_i и в сети - U .

Состояние рассматриваемой сети описывается вектором

(a_1, a_2, \dots, a_n) , где a_i - число заявок в i -й СМО. Если через $p_i(a_i)$ обозначить распределение вероятностей того, что в установившемся режиме в i -й СМО будет находится a_i заявок, то распределение числа заявок в сети определяется произведением распределений по всем СМО:

$$p(a_1, a_2, \dots, a_n) = p_1(a_1) \cdot p_2(a_2) \dots p_n(a_n), \quad (3.2)$$

где $p(a_i)$ - вероятность нахождения a_i заявок в i -й СМО.

Справедливость выражения (3.2) была доказана теоремой Джексона. Произведение распределений вероятностей в выражении (3.2) означает, что между отдельными СМО сети в значительной степени существует независимость. Это даёт основание декомпозировать сеть на n независимых СМО.

Если характеристики сети не меняются во времени (т.е. когда средние значения имеют постоянные величины), то в сети существует стационарный режим. Существование его связано с существованием стационарного режима в каждой СМО сети и определяется её загрузкой ρ_i . Для одноканальной СМО условием существования стационарного режима является:

$$\rho_i = \lambda_i v_i < 1 \quad (3.3)$$

для многоканальной -

$$\rho_i = \frac{\lambda_i v_i}{k_i} = \frac{\beta_i}{k_i} < 1, \quad (3.4)$$

где k_i - число каналов в i -й СМО; $\beta_i = \lambda_i v_i$ - среднее число занятых каналов. Стационарный режим означает, что в отдельных СМО и в сети в целом конечны средние задержки и длины очередей.

Рассмотрим характеристики отдельных СМО сети.

Средняя длина очереди заявок для одноканальной СМО определяется выражением:

$$l_i = \frac{\rho_i^2}{1 - \rho_i}, \quad (3.5)$$

Для многоканальной СМО

$$l_i = \frac{\beta_i^{k_i+1}}{k_i! k_i \left(1 - \beta_i / k_i\right)^2} p_{0i}, \quad (3.6)$$

где p_{0i} - вероятность отсутствия заявок в i -й СМО,

$$p_{0i} = \left[\sum_{a_i=0}^{k_i-1} \frac{\beta_i^{a_i}}{a_i!} + \frac{\beta_i^{k_i}}{k_i! \left(1 - \beta_i / k_i\right)} \right]^{-1}$$

Среднее число заявок, пребывающих в системе, для одноканальной СМО определяется средней длиной очереди l_i и средним числом заявок, находящихся на обслуживании, ρ_i ,

$$m_i = l_i + \rho_i = \frac{\rho_i}{1 - \rho_i}. \quad (3.7)$$

Для многоканальной СМО среднее число заявок, находящихся на обслуживании, определяется числом занятых каналов, поэтому

$$m_i = l_i + \beta_i = \frac{\beta_i^{k_i+1}}{k_i! k_i \left(1 - \beta_i / k_i\right)^2} p_{0i} + \beta_i. \quad (3.8)$$

Среднее время ожидания заявки в очереди для одноканальной СМО равно частному от деления средней длины очереди l_i на интенсивность λ_i входящего в i -й СМО потока:

$$\omega_i = \frac{l_i}{\lambda_i} = \frac{v_i \cdot \rho_i}{1 - \rho_i}, \quad (3.9)$$

Для многоканальной СМО

$$\omega_i = \frac{l_i}{\lambda_i} = \frac{v_i \cdot \beta_i^{k_i}}{k_i! k_i \left(1 - \beta_i / k_i\right)^2} p_{0i}. \quad (3.10)$$

Среднее время пребывания заявки в системе определяется средней задержкой её в очереди и временем обслуживания в i -й СМО.

Для одноканальной СМО

$$u_i = \omega_i + v_i = \frac{v_i}{1 - \rho_i}; \quad (3.11)$$

для многоканальной

$$u_i = \omega_i + v_i = \frac{v_i \cdot \beta_i^{k_i}}{k_i! k_i \left(1 - \beta_i / k_i\right)^2} p_{0i} + v_i \quad (3.12)$$

На основании полученных характеристик отдельных СМО определяют характеристики сети в целом.

Среднее число заявок, ожидающих обслуживания в сети (т.е. среднее число заявок в очередях сети),

$$L = \sum_{i=1}^n l_i . \quad (3.13)$$

Среднее число заявок, пребывающих в сети,

$$M = \sum_{i=1}^n m_i \quad (3.14)$$

Поскольку каждая заявка может получить обслуживание в i -й СМО в среднем раз, то время ожидания обслуживания и время пребывания её в системе увеличится в раз. Среднее время ожидания заявки в очередях сети

$$W = \sum_{i=1}^n \alpha_i \omega_i , \quad (3.15)$$

а время пребывания

$$U = \sum_{i=1}^n \alpha_i u_i . \quad (3.16)$$

3.2.3 Характеристики СМО с ограниченной длиной очереди.

Рассмотренные характеристики применимы для СМО с неограниченной длиной очереди. Однако в реальных системах число мест в очереди может быть ограничено и заявка, застигнувшая очередь заполненной, получает отказ в обслуживании.

В случае ограниченной длины очереди перед одноканальной СМО с дисциплиной обслуживания FIFO, распределение вероятностей наличия в системе τ заявок определится выражением

$$P_{\tau i} = \frac{\rho_i^{\tau} (1 - \rho_i)}{1 - \rho_i^{R+1}} \quad \text{при} \quad 0 \leq \tau \leq R , \quad (3.17)$$

где R – число мест в очереди обслуживания в i -й СМО.

Длина очереди к i -й СМО

$$l_i = \frac{1 - \rho_i}{1 - \rho_i^{R+1}} \left[\sum_{\tau=1}^R \tau \rho_i^{\tau} + 1 \right] - 1 , \quad (3.18)$$

Число заявок, пребывающих в i -й СМО,

$$m_i = \frac{1 - \rho_i}{1 - \rho_i^{R+1}} \sum_{\tau=1}^R \tau \rho_i^{\tau} . \quad (3.19)$$

Время ожидания в очереди

$$\omega_i = \frac{1 - \rho_i}{\lambda_i (1 - \rho_i^{R+1})} \left[\sum_{\tau=1}^R \tau \rho_i^{\tau} + 1 \right] - \frac{1}{\lambda_i} . \quad (3.20)$$

Время пребывания в i -й СМО

$$u_i = \frac{1 - \rho_i}{\lambda_i (1 - \rho_i^{R+1})} \sum_{\tau=1}^R \tau \rho_i^{\tau}. \quad (3.21)$$

3.2.4. Характеристики СМО с приоритетами.

Реальные системы не всегда используют дисциплину обслуживания *FIFO* (первая поступившая заявка обслуживается первой). Особенно часто встречающейся дисциплиной является приоритетная, в которой заявки, имеющие приоритет, обслуживаются вне очереди рис.(1.2).

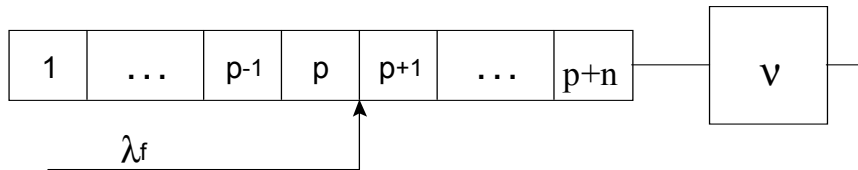


Рис. 1.2. СМО с приоритетным обслуживанием

Если заявка, имеющая приоритет, немедленно прерывает обслуживание другой заявки, то система работает с абсолютным приоритетом; если же заявка дожидается окончания обслуживания ранее поступившей, то система работает с относительным приоритетом.

Рассмотрим способ определения среднего времени ожидания для систем с относительным приоритетом.

Предположим, что поступающие заявки принадлежат одному из F различных приоритетных классов, обозначаемых через $f (f=1, 2, \dots, F)$. Будем считать, что чем больше значение индекса класса, тем выше приоритет этого класса. Таким образом, заявкам из приоритетного класса f представляется преимущественное обслуживание по сравнению с заявками из приоритетного класса $f-1$. Если в СМО находится несколько заявок с одинаковым приоритетом, то их обслуживание производится в соответствии с дисциплиной *FIFO*.

Рассмотрим модель, в которой заявки из приоритетного класса f образуют пуассоновский поток с интенсивностью λ_f , а время обслуживания v_f распределено по экспоненциальному закону.

Определим параметры модели:

$$\begin{aligned} \lambda &= \sum_{f=1}^F \lambda_f; \\ v &= \sum_{f=1}^F \frac{\lambda_f}{\lambda} v_f; \\ p_f &= \lambda_f v_f; \end{aligned}$$

$$\rho = \lambda v = \sum_{f=1}^F \rho_f .$$

Здесь под ρ_f подразумевается нагрузка обслуживающего прибора заявки из приоритетного класса f ; ρ - общая нагрузка обслуживающего прибора, причём для обеспечения стационарного режима необходимо выполнение условия $\rho < 1$.

Введём следующие обозначения: ω_p - среднее время ожидания заявки из приоритетного класса f ; u_f - среднее время пребывания в СМО заявки, поступившей из приобретённого класса f .

Время ожидания разлагается на три составляющие: 1) время, связанное с тем, что в момент поступления данной заявки прибор был занят обслуживанием другой заявки; 2) время, связанное с тем, что в момент поступления данной заявки в очереди уже находятся заявки с приоритетом, равным или более высоким, чем у поступившей; 3) время, обусловленное тем, что заявки, поступившие позже, имеют более высокий приоритет.

Исследование приоритетных систем обычно начинают с поступившей вновь заявки из приоритетного класса f . Будем называть эту заявку меченой. Обозначим через ω_0 среднюю задержку меченой заявки, связанную с наличием другой заявки на обслуживание; ω_0 будем определять через нагрузку прибора ρ_i , которая представляет собой вероятность занятия обслуживающего прибора заявками из i -го класса. ρ_i также может быть интерпретирована как доля времени, в течение которого прибор занят заявками из i -го класса, тогда.

$$\omega_0 = \sum_{i=1}^F \rho_i . \quad (3.22)$$

Рассмотрим теперь вторую составляющую времени ожидания, связанную с тем, что перед меченой заявкой обслуживаются заявки с равным или более высоким приоритетом, которые меченая заявка застала в очереди. Введём обозначение l_{if} - число заявок из класса i , которые застала в очереди меченая заявка из класса f . В соответствии с формулой Литтла $l_f = \lambda_i a_i$, где $i=f+1, \dots, F$. Отсюда средняя задержка меченой заявки составит $\sum_{i=f}^F v_i \lambda_i \omega_i$.

Аналогично можно определить третью составляющую среднего времени ожидания (задержка меченой заявки за счёт того, что заявки, поступающие после неё, имеют более высокий приоритет). Пусть m_{if} - число заявок из класса i , поступающих в СМО, когда меченая заявка (из класса f) находится в очереди, и получающих обслуживание раньше

меченой заявки. Тогда $m_f = \lambda_{i_f} a_f$, где $i=f+1, f+2, \dots, F$, а средняя задержка составит $\sum_{i=f+1}^P \lambda_i \omega_f$.

Таким образом, для СМО с относительным приоритетом

$$\omega_f = \omega_o + \sum_{i=f}^F v_i \lambda_i \omega_i + \sum_{i=f+1}^F v_i \lambda_i \omega_f. \quad (3.23)$$

Так как $v_i \lambda_i = \rho_i$, то, преобразовав выражение относительно ω_f , получим

$$\omega_f = \frac{\omega_o + \sum_{i=f+1}^F v_i \lambda_i \omega_f}{1 - \sum_{i=f}^F \rho_i}, \quad (3.24)$$

где $f=1, 2, \dots, F$

Система уравнений (3.24) может быть легко решена рекуррентно, т.е. сначала находится ω_f , затем ω_{f-1} и т.д. Например, имеется три класса приоритетов, причём наибольший приоритет имеет класс 3. Тогда среднее время задержки каждого класса определится так:

$$\begin{aligned} \omega_3 &= \frac{a_o}{1 - \rho_3}; \\ \omega_2 &= \frac{a_o + \rho_3 a_3}{1 - \rho_2 - \rho_3}; \\ \omega_1 &= \frac{a_o + \rho_2 a_2 + \rho_3 a_3}{1 - \rho_1 - \rho_2 - \rho_3}. \end{aligned}$$

3.2.Макроанализ производительности многопроцессорных систем

3.2.1.Модель с разделяемой памятью

Многопроцессорная система с разделяемой (общей) памятью приведена на рис.2.1. Предварительно для упрощения будем считать, что в системе отсутствуют как внутрипроцессорные, так и внешние кэш. Виртуальная память системы содержит два уровня, причём на первом

уровне находится основная память, являющаяся общей для всех процессорных модулей, а на втором уровне – ВЗУ. Обмен между уровнями производится посредством канала прямого доступа.

Предлагается, что в основной памяти системы размещается только часть программ и данных, используемых оперативно, другая часть программ и данных хранится в ВЗУ.

Будем считать, что система однородна, т.е. все процессоры одинаковы. Однородность предоставляет возможность системе исполнять любую программу на любом процессоре. Поскольку любой процессор может принять на обслуживание любую заявку, то система работает в режиме разделения нагрузки.

Будем считать, что задания на решение задач формируются с помощью терминалов (на рис.2.1 терминалы представлены как устройства ввода/вывода - УВВ), т.е. поступают в систему извне. В процессе решения задача с некоторой вероятностью p может обращаться в ВЗУ за недостающими программами или данными (см. рис.3.1). Обслуженные в ВЗУ задачи возвращаются на дообслуживание в процессор. Решенные задачи с вероятностью $(1-p)$ выходят из системы и возвращаются на терминалы пользователей. Таким образом, рассматриваемая многопроцессорная система функционирует в режиме оперативной обработки.

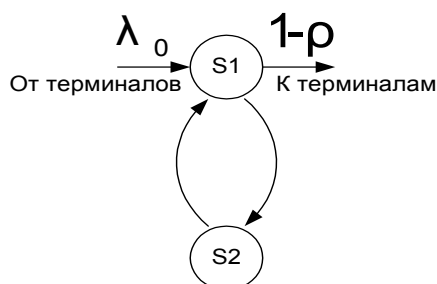


Рис. 3.1. Обобщенная модель системы оперативной обработкой

Функционирование многопроцессорной системы с двухуровневой памятью в режиме оперативной обработки можно представить в виде разомкнутой сетевой модели. Такая модель позволяет проанализировать задержки, связанные с обращением задач в ВЗУ и с ограничением числа процессоров в системе. Чтобы не усложнять модель детализацией, будем игнорировать потери, связанные с ограничением других ресурсов.

Будем считать, что на вход сети поступает пуассоновский поток заявок с интенсивностью λ_0 . Для реальной системы это соответствует потоку заданий на входе многопроцессорной системы. Кроме того принято, что времена обслуживания во всех устройствах распределены по

экспоненциальному закону. Заявка обслуживается в начале в СМО S_1 , которая моделирует обработку, происходящую в системе процессор - память (ЦП/ОП). Получив обслуживание в S_1 , она с некоторой вероятностью p обращается в СМО, которая моделирует ВЗУ (S_2). С вероятностью $(1-p)$ заявка покидает систему.

Модель системы с разделяемой (общей) памятью, одним каналом ПДП и k индивидуальными ВЗУ представлена на рис.3.2.

Поскольку каждый процессор может выполнять любую заявку, размещённую в общей памяти, то очередь для всех процессоров единая. Модель процесса выполнения заявок в процессорах может быть представлена многоканальной СМО S_1 с временем обслуживания v_1 .

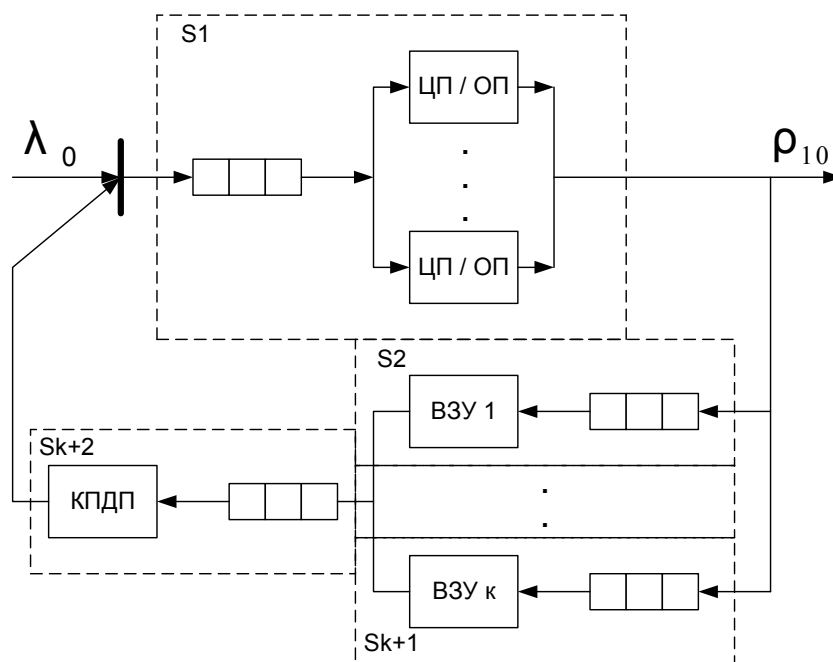


Рис. 3.2. Модель системы с разделяемой памятью и индивидуальными ВЗУ

Вначале заявка получает обслуживание в СМО S_1 . Если процесс её обработки закончился, то она с вероятностью p_{10} покидает систему. Если задаче в процессе её обработки потребуется обращение к ВЗУ, то она с вероятностью p_{1i} (где $i=2, \dots, k+1$) поступает на обслуживание в СМО S_i , имеющее среднее время обслуживания v_i . Если заявка допускает обслуживание только в одном ВЗУ, то система внешней памяти представляется совокупностью одноканальных СМО

Поскольку весь процесс обмена с внешней памятью ведётся через канал прямого доступа, то модель КППДП может быть представлена как одноканальная СМО с временем обслуживания v_{k+2} .

Таким образом модель, изображенная на рис.3.2. представляется как совокупность одноканальных и многоканальных СМО, объединённых в стохастическую сеть.

Граф передач такой сети будет иметь вид, изображённый на рис.3.3. Над дугами указаны вероятности передач из одной СМО в другую. Над теми дугами, где вероятности не указаны, их значения равны единице.

Матрица вероятностей передач для такой сети будет иметь вид:

$$P = [p_{ij}] = \begin{matrix} & \begin{matrix} S_0 & S_1 & S_2 & \cdot & S_{k+2} \end{matrix} \\ \begin{matrix} S_0 \\ S_1 \\ S_2 \\ \cdot \\ S_{k+2} \end{matrix} & \begin{bmatrix} 0 & 1 & 0 & \cdot & 0 \\ p_0 & 0 & p_1 & \cdot & p_{1,k+2} \\ 0 & 0 & 0 & \cdot & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & 0 & 0 & \cdot & 0 \end{bmatrix} \end{matrix}$$

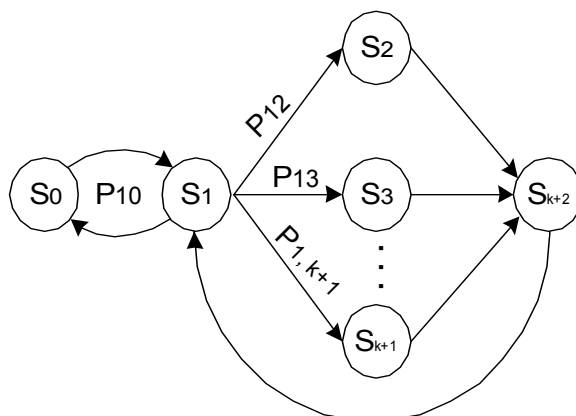


Рис. 3.3. Граф передач сети, изображенной на рис. 3.2

Для определения характеристик отдельных характеристик отдельных СМО и всей сети в целом необходимо знать интенсивности потоков заявок на входе каждой СМО. Для этого можно воспользоваться системой уравнений (2.1), которая будет состоять из $k+2$ уравнений:

$$\lambda_0 = \lambda_1 \cdot P_{10};$$

$$\lambda_1 = \lambda_0 + \lambda_{k+2};$$

$$\lambda_2 = \lambda_1 \cdot P_{12};$$

$$\dots$$

$$\lambda_{k+1} = \lambda_l \cdot P_{l,k=1};$$

$$\lambda_{k+2} = \lambda_2 + \lambda_3 + \dots + \lambda_{k+1}$$

Если в сети существует стационарный режим, определяемый условиями (2.3) и (2.4), то характеристики отдельных СМО и сети в целом определяются выражениями (2.5), (2.7), (2.9), (2.11), (2.13), (2.14), (2.15), (2.16). Если заявка, поступившая во внешнюю память, может обслуживаться любым ВЗУ, то модель внешней памяти будет представлена в виде многоканальной СМО (рис.3.4) с временем обслуживания ν_2 . На рис.3.5 показан граф передач такой сети. Параметры и характеристики такой сети определяются аналогично предыдущей модели.

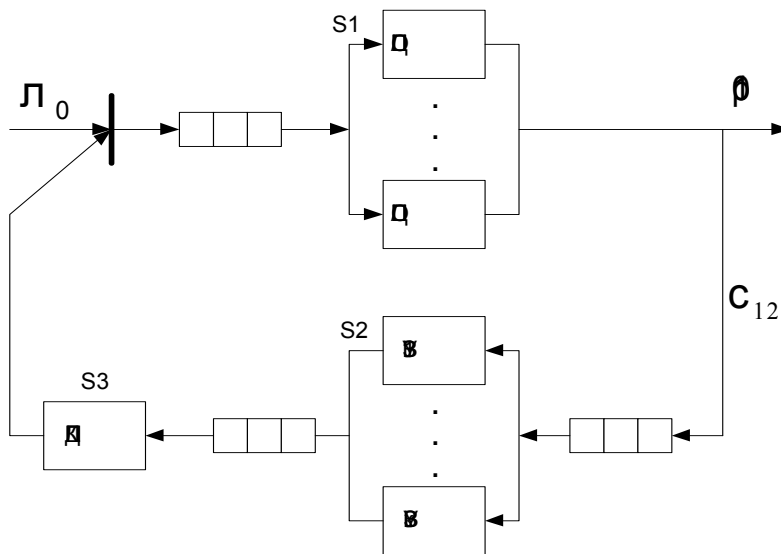


Рис. 3.4. Модель системы с разделяемой памятью и общими ВЗУ

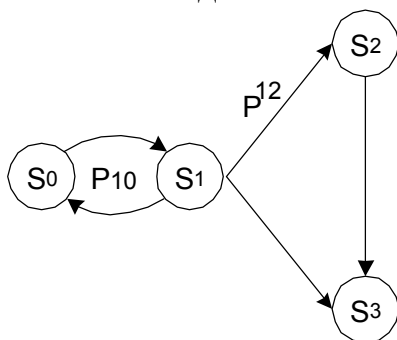


Рис. 3.5. Граф передач сети, изображенной на рис. 3.4

3.2.2. Модель системы с разделяемой распределённой памятью.

Структура многопроцессорной системы с разделяемой распределённой памятью изображена на рис.2.3. Виртуальная память

системы, как и в предыдущем случае, является двухуровневой, состоящей из локальной памяти, размещенной в процессорных модулях, и ВЗУ. Кэш-память с целью упрощения модели не представлена ни на одном уровне.

В такой системе каждый процессор в основном работает с локальной памятью, в которой хранятся копии необходимых для функционирования программ и данных, загружаемых из внешней памяти. В процессе решения задач процессоры могут обращаться во внешнюю память за недостающими программами и данными.

Поскольку процессоры работают в основном самостоятельно (такие системы называют с разделением функций), то их можно представить в модели как n отдельных СМО с временем обслуживания $\nu_1, \nu_2, \dots, \nu_n$.

В зависимости от способа обслуживания внешней памятью возможны два варианта моделей. Если заявка обслуживается только в одном ВЗУ, то модель системы примет вид, изображённый на рис.3.6.

Она представляется как совокупность отдельных СМО ($S_1, S_2, \dots, S_{n+1}, \dots, S_{n+k}, S_{n+k+1}$), объединённых в стохастическую сеть. Граф передач такой сети будет иметь вид, изображённый на рис.3.7.

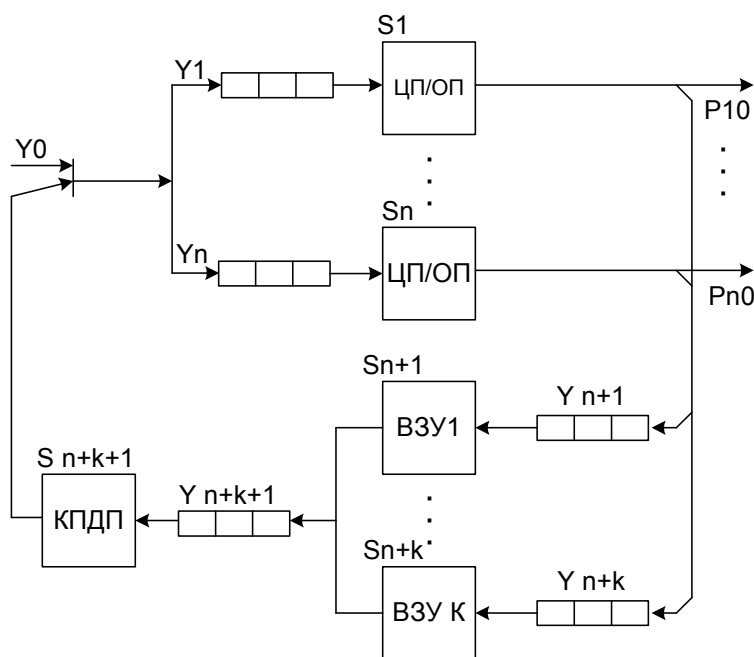


Рис.3.6. Модель системы с разделяемой распределённой памятью и индивидуальными ВЗУ.

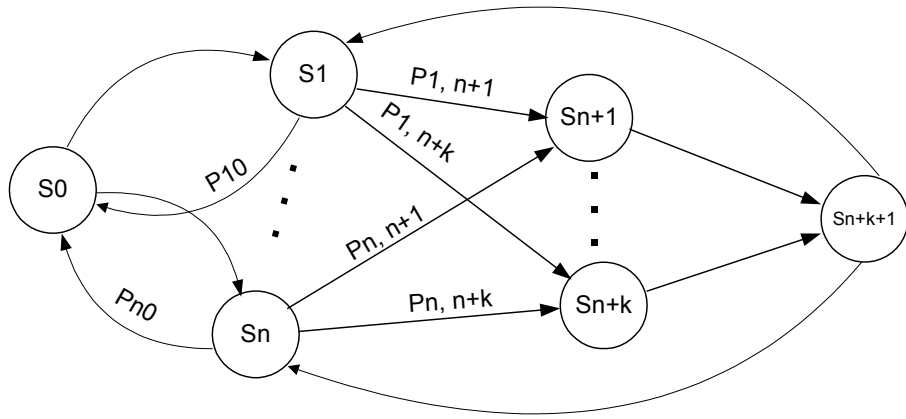


Рис.3.7. Граф передач сети, изображённой на рис.3.6.

Интенсивность потоков на входах СМО определяется из системы уравнений (3.1):

$$\lambda_o = \lambda_1 P_{10} + \lambda_2 P_{20} + \dots + \lambda_n P_{n0};$$

$$\lambda_1 = \lambda_o + \lambda_{n+k+1} P_{n+k+1,1};$$

$$\lambda_2 = \lambda_o + \lambda_{n+k+1} P_{n+k+1,2};$$

...

$$\lambda_n = \lambda_o + \lambda_{n+k+1} P_{n+k+1,n};$$

...

$$\lambda_{n+k} = \lambda_1 P_{1,n+k} + \lambda_2 P_{2,n+k} + \dots + \lambda_n P_{n,n+k};$$

$$\lambda_{n+k+1} = \lambda_1 + \lambda_2 + \dots + \lambda_n;$$

$$\lambda_{n+1} = \lambda_1 P_{1,n+1} + \lambda_2 P_{2,n+1} + \dots + \lambda_n P_{n,n+1};$$

Стационарный режим в сети будет существовать в том случае, когда выполняется условие (3.3). Характеристики отдельных СМО и сети в целом определяются выражениями (3.5), (3.7), (3.9), (3.11), (3.13)-(3.16).

Если заявка, поступившая во внешнюю память, может обслуживаться любым ВЗУ, то её модель будет многоканальной СМО. Тогда модель всей системы примет вид, изображённый на рис.3.7. Граф передач сети будет иметь вид, изображённый на рис.3.8.

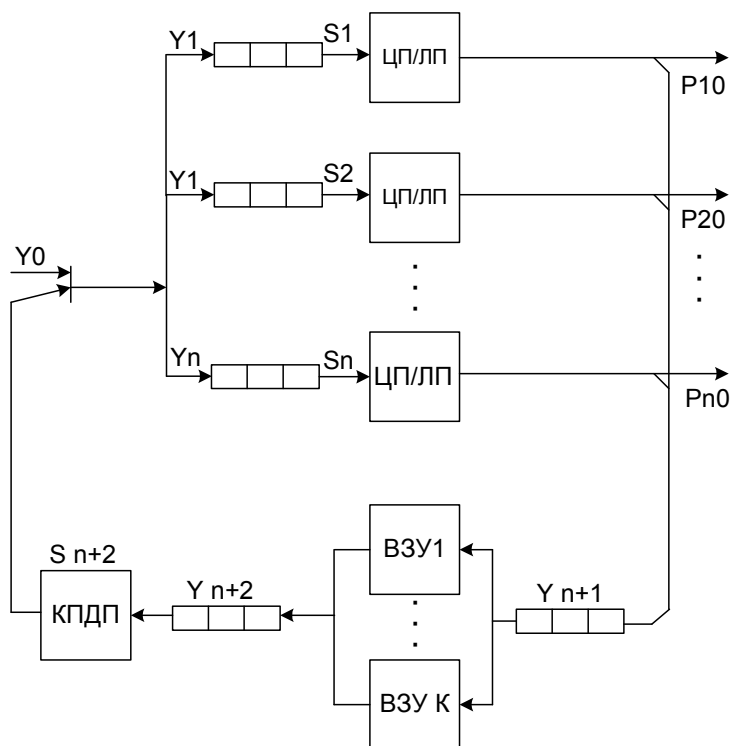


Рис.3.7. Модель системы с разделяемой распределённой памятью и общими ВЗУ.

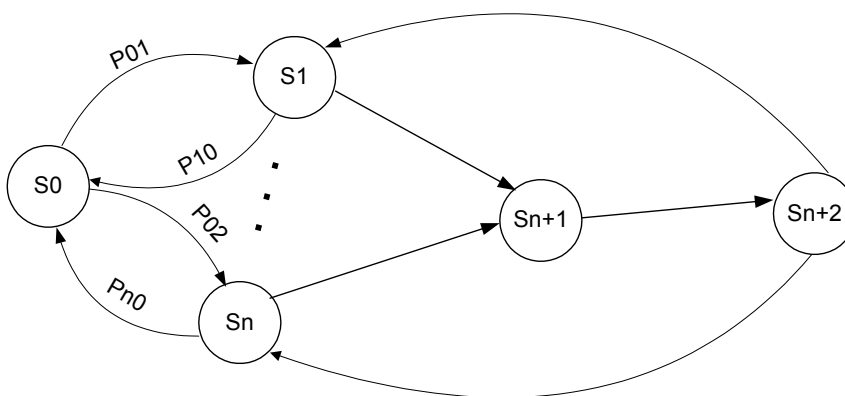


Рис.3.8. Граф передач сети, изображённой на рис.3.7.

Параметры и характеристики отдельных СМО и сети в целом определяются по выражениям (3.1), (3.3)-(3.16).

3.3. Микроанализ производительности многопроцессорных систем.

Микроанализ используют в качестве инструмента исследования ВС на более низких уровнях проектирования по сравнению с макроанализом. В его задачу может входить оценка пропускной способности коммуникационной среды, оценка влияния различных методов доступа к общей памяти (например, по прерываниям или прямым доступом) на её пропускную способность, оценку времени доступа к файлам, размещаемым в ВЗУ, а также оценка задержек в обслуживании, связанных с аппаратными конфликтами, возникающими в общих ресурсах. Но чаще всего микроанализ производят с целью оценки потерь производительности. Почему необходима такая оценка? Дело в том, что производительность многопроцессорной системы всегда ниже суммарной производительности входящих в неё процессоров. Это объясняется в основном конечностью числа ресурсов системы, находящихся в общем пользовании. Конечность числа ресурсов приводит к очередям, возникающим при запросе этих ресурсов, а следовательно, к задержкам в обслуживании.

В системах с ОШ в качестве общих ресурсов могут выступать буферные ЗУ, используемые в каналах передачи данных, блоки разделяемой памяти, внешние запоминающие устройства, шина обмена.

В дальнейшем будут рассмотрены оценки влияния аппаратных конфликтов, возникающих в общей шине и общей памяти на производительность многопроцессорных систем с ОШ. Для этого необходимо более подробно рассмотреть структурную организацию центральной части таких систем.

3.3.1. Модель системы с разделяемой памятью для анализа конфликтов, возникающих в общих ресурсах.

Модель представляется в виде разомкнутой стохастической сети, в которой источником заявок выступают процессоры, генерирующие поток команд (в отличие от моделей для макроанализа, где источником заявок являются задания, поступающие на обработку в ВС извне, например, с терминалов), а в качестве обслуживающих приборов – ОШ и модули общей памяти.

Модель источника заявок может быть представлена так, как изображено на рис.3.9. φ_i моделирует поток команд обращения i -го

процессора к общей памяти, γ_i - к локальной памяти. В структурах систем с общей памятью поток γ_i отсутствует, т.к. отсутствует локальная память; в структурах с локальной памятью действуют оба потока.

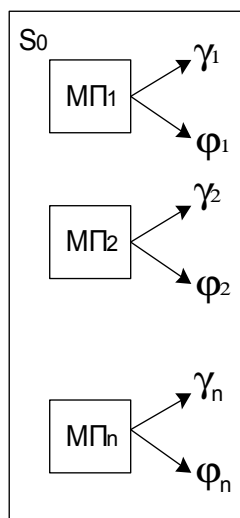


Рис.3.9. Модель источника заявок.

Процесс выполнения заявок из потока ϕ_i проходит две фазы обслуживания: вначале – общей шиной S_1 , затем одним из модулей памяти $S_j (j=2, \dots, m+1)$. Заявки из потока γ_i проходят одну фазу – только модуля памяти S_i . Будем считать все потоки простейшими, а времена обслуживания распределёнными по экспоненциальному закону.

Структурная схема модели системы с общей памятью показана на рис.3.17. Интенсивность потока заявок на входе сети определяется суммарным потоком заявок ϕ_i роцессоров. Заявка процессора, поступившая в СМО S_1 и заставшая его занятым, становится в очередь, которую будем считать неограниченной. При обслуживании с приоритетами очередь перед СМО S_1 устанавливается в соответствии с классом приоритета заявки.

Заявка, обслуженная в СМО S_1 , с вероятностью p_{ij} поступает на обслуживание в одну из СМО $S_j (j=2, \dots, m+1)$. Процесс обслуживания в СМО S_j может быть либо в соответствии с дисциплиной *FIFO*, либо с приоритетной дисциплиной. Если очередь перед СМО S_2, \dots, S_{m+1} ограничена, то в случае её занятости заявка получит отказ в обслуживании и с вероятностью p_{11} возвратится в конец очереди к СМО S_1 (при беспriorитетной дисциплине обслуживания). В случае приоритетного обслуживания СМО S_1 заявка, получившая отказ в обслуживании в одной

из СМО S_j , приобретает тот же приоритет, который имела ранее. Получив полное обслуживание, заявка покидает сеть.

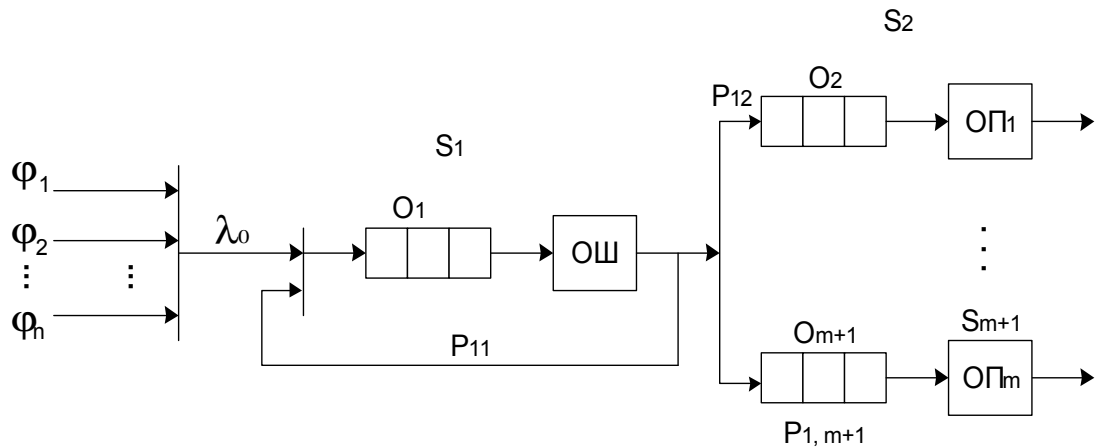


Рис.3.10. Разомкнутая сетевая модель системы с разделяемой памятью.

Рассмотренная модель предназначена для исследования многопроцессорных систем с интерфейсом, использующим принцип обмена с буферизацией сообщений, т.е. структуры, изображённой на рис.3.9. СМО S_1 моделирует ОШ, а S_2, \dots, S_{m+1} - модули общей памяти. В процессе обращения микропроцессоров к модулям общей памяти каждая заявка проходит два этапа. Вначале она требует доступа к ОШ, затем к одному из регистров буферного накопителя размещённого в коммутаторах КОП модуля памяти. Буферный накопитель моделируется очередью $O_j (j=2, \dots, m+1)$.

Если в момент выдачи заявки ОШ оказывается занятой, то процессор, её выдавший, переходит в режим ожидания, который длится до тех пор, пока не освободится ОШ. Режим ожидания моделируется очередью O_1 , которая считается неограниченной. Граф передач сети представляется в виде, изображённом на рис.3.11.

Рассмотрим методику определения влияния конфликтных ситуаций, возникающих в ОШ и общей памяти, на время выполнения команды процессора. Очевидно, что время выполнения команды зависит от времени обращения к памяти. Поскольку процесс обращения к памяти состоит из двух составляющих – обслуживания в общей шине и обслуживания в общей памяти, то время доступа к памяти будет определяться временем пребывания заявки в СМО S_1 и в СМО S_j , т.е. временем пребывания заявки в сети.

Интенсивность потока заявок на входе сети $\lambda_o = \sum_{i=1}^n \varphi_i$, где φ_i - интенсивность потока команд i -го процессора. Она может быть определена как $\varphi_i = N_i / T_i$, где N_i - среднее число команд обращения i -го микропроцессора к памяти за время решения задачи T_i .

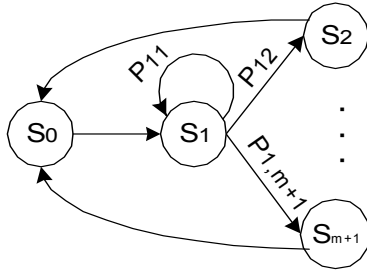


Рис.3.11. Граф передач сети, изображённой на рис.3.10.

Интенсивность потоков на входах СМО сети составит $\lambda_j = P_{1j} \cdot \lambda_1$,

а поскольку $\lambda_1 = \lambda_o + P_{11}\lambda_1$, то отсюда следует: $\lambda_1 = \frac{\lambda_o}{1 - P_{11}}$.

Вероятности обращения заявок в СМО $S_j (j=2, \dots, m+1)$ могут быть определены как $P_j = N_{1j} / N$, где N_{1j} - среднее число команд обращения в j -й модуль памяти за время решения задачи; N - суммарное число команд обращения к памяти микропроцессоров, т.е. $N = \sum_{j=2}^{m+1} N_j$.

В случае неограниченных длин очередей в СМО сети отказы в обслуживании будут отсутствовать, т.е. $P_{11}=0$, $\lambda_1=0$.

Время пребывания заявки в СМО S_1 составит $u_1^o = \frac{\rho_1}{\lambda_1(1 - \rho_1)}$,

где $\rho_1 = \lambda_1 t_B$ - загрузка общей шины, t_B - время обмена данными по общей шине. Время пребывания заявки в СМО $S_j (j=2, \dots, m+1)$ составит $u_j = \frac{\rho_j}{\lambda_j(1 - \rho_j)}$, где $\rho_j = \lambda_j t_C$ - загрузка модуля памяти; t_C - цикл памяти.

В случае ограничения длины очереди перед СМО S_j время пребывания определится как $u_j = \frac{1 - \rho_j}{\lambda_j(1 - \rho_j^{R+1})} \sum_{\tau=1}^R \tau \rho_j^\tau$, где R - суммарное число мест в очереди и в обслуживающем приборе.

В этом случае возможны отказы в обслуживании СМО S_j , причём вероятность отказа p_{11} будет складываться из вероятностей отказа в

каждом из СМО S_2, \dots, S_{m+1} . Учитывая, что отказы начинаются с момента, когда $r=R$, получим:

$$P_1 = \sum_{j=2}^{m+1} \frac{(1-\rho_j)\rho_j^R}{1-\rho_j^{R+1}}.$$

При обслуживании СМО S_l с относительными фиксированными приоритетами время пребывания заявки, имеющий приоритет, в этой СМО определится через время задержки ω_f (в соответствии с выражением (2.24)) и время обслуживания обмена t_B :

$$u_1^f = \frac{\omega_o + \sum_{i=f+1}^n \rho_i \cdot \omega_i}{1 + \sum_{i=f}^n \rho_i} + t_B,$$

где $f=1, 2, \dots, n$ - классы приоритетов заявок, присвоенных процессорам. Время пребывания заявки в сети $U = \alpha_1 u_1 + \alpha_j u_j$, где $\alpha_l = \lambda_l / \lambda_0$ и $\alpha_j = \lambda_j / \lambda_0$ - коэффициенты передачи сети, определяющие, сколько раз заявка, поступившая в сеть, получает обслуживание в соответствующем узле.

Полученное время пребывания является временем обращения микропроцессора к памяти. Если в каждой из СМО сети длина очереди неограниченна, то это время составит

$$t_M = \frac{\alpha_1 \rho_1}{\lambda_1 (1 - \rho_1)} + \frac{\alpha_j \rho_j}{1 - \rho_j}.$$

Аналогично можно получить выражения t_{MG} для тех случаев, когда в узлах сети используются другие дисциплины обслуживания.

Если используется способ связи без буферизации сообщений, то общей шиной и, следовательно, общей памятью будет монопольно владеть только один вычислительный модуль, причём до тех пор, пока полностью не завершится обмен между ВМ и модулем памяти. Модель системы в этом случае преобразуется в одноузловую (рис.3.12). Время обращения к общей памяти

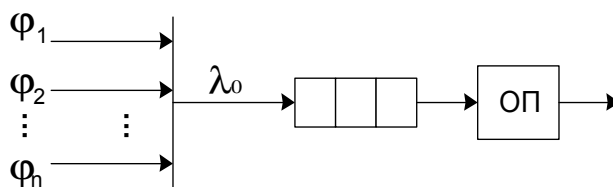


Рис.3.12. Модель системы с общей памятью и шинным интерфейсом без буферизации сообщений

$$t_{\text{ш}} = \frac{\rho}{\lambda_o(1-\rho)}, \text{ где } \lambda_o = \sum_{i=1}^n \varphi_i; \quad \rho = \lambda_o \cdot t_B.$$

3.3.2. Микроанализ конфликтов в системах с общей шиной и распределённой памятью.

Построим модель такой системы в предположении, что она состоит из n процессоров, каждый из которых имеет локальную память. Часть ЛП, доступная всем микропроцессорам системы, определена как общая.

В процессе вычислительный модуль вначале запрашивает ОШ. Получив доступ к шине, ВМ-источник выставляет адреса, управляющие сигналы и данные для их записи (считывания) в область общей памяти ВМ-приёмника. Будем считать, что обращение к общей памяти производится прямым доступом, т.е. ВМ-источник обладает приоритетом по отношению ВМ-приёмнику.

Для передачи одного слова данных от одного вычислительного модуля другому требуется один цикл шины при межмодульной связи без буферизации сообщений. При связи с буферизацией для записи требуется один цикл шины, для чтения – два цикла. Средняя длительность цикла шины в этом случае определяется выражением (3.27). Микропроцессоры, потребовавшие доступа к занятой ОШ, переходят в режим ожидания.

Вначале рассмотрим модель системы, в которой используется принцип связи по ОШ без буферизации сообщений. На первой фазе обслуживания (рис.3.13) она содержит модель ОШ (СМО S_1), а на второй – модель модуля памяти (СМО S_2, \dots, S_{n+1}). Число СМО, моделирующих память, равно числу ВМ, т.к. вся память системы как локальная, так и общая распределена по вычислительным модулям.

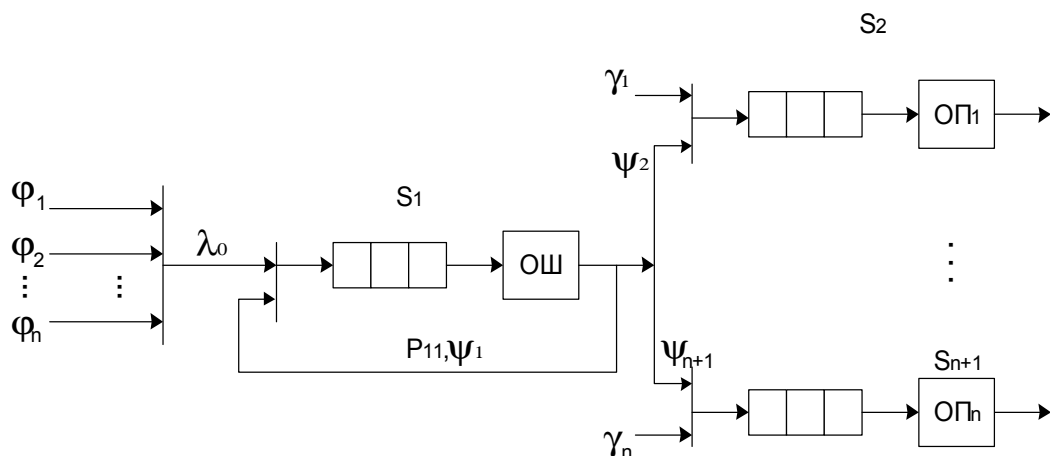


Рис.3.13. Модель системы с распределённой памятью.

Будем считать, что заявка, поступившая из потока ψ_1, \dots, ψ_n , принимается на обслуживание немедленно (с относительным приоритетом). Если СМО $S_j (j=2, \dots, n+1)$ занята обслуживанием ранее поступившей заявки из этого же потока, то она получает отказ в обслуживании и возвращается в конец очереди к СМО S_1 . Таким образом, моделируется процесс прямого доступа к памяти со стороны внешних, по отношению к рассматриваемому, вычислительных модулей. Суммарный поток отказов узлов S_2, \dots, S_{n+1} составит вероятность отказа p_{11} , причём $P_1 = \sum_{j=2}^{n+1} P_{omkj}$, где P_{omkj} - вероятность отказа СМО S_j . Поскольку отказы в СМО S_j возникают в том случае, когда обслуживающий прибор (в данном случае модуль памяти) занят обслуживанием ранее поступившей заявки из потока ψ_j , то P_{omkj} определяется вероятностью занятия прибора заявками из этого потока. Вероятность занятия одноканального прибора равна ρ_j , поэтому $P_{omkj} = \rho_j$, а $P_1 = \sum_{j=2}^{n+1} P_j = \sum_{j=2}^n \psi_j t_c$. Интенсивность заявок на входе СМО S_1 $\lambda_1 = \frac{\lambda_o}{1 - \rho_1}$, где $\lambda_o = \sum_{i=1}^n \phi_i$.

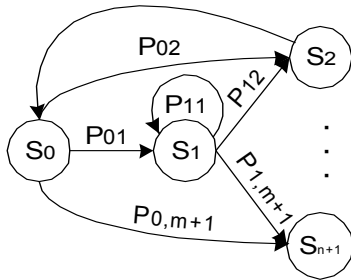


Рис.3.14. Граф передач сети, изображённой на рис.3.13.

Заявка, обслуженная СМО S_1 , с вероятностью p_{1j} поступает на обслуживание в один из узлов S_2, \dots, S_{n+1} , причём, как и в предыдущих моделях, $p_{1j} = N_j/N$. Интенсивность потока заявок во второй фазе обслуживания $\psi_j = p_{1j} \cdot \lambda_1$.

Заявки из потока λ_1 проходят две фазы обслуживания: вначале в СМО S_1 , моделирующем ОШ, затем в одном из узлов S_2, \dots, S_{n+1} . Время пребывания в СМО S_1 при беспriorитетной дисциплине обслуживания

$$u_1 = \frac{\rho_1}{\lambda_1(1 - \rho_1)}, \text{ где } \rho_1 = \lambda_1 t_B - \text{загрузка ОШ.}$$

Во второй фазе обслуживания (СМО S_2, \dots, S_{n+1}) заявки из потока φ_i , а следовательно, и заявки из потока φ_j получают приоритетное обслуживание. Поскольку приоритет у всех заявок из потока φ_j одинаков ($f=1$), то в соответствии с выражением (3.24) получим время ожидания их обслуживания $\omega_j = \frac{a_o}{1-\rho_j}$. Тогда время пребывания заявки в СМО

$S_j (j=2, \dots, n+1)$ определится как $u_j = a_j + \rho_j$, где $\rho_j = \psi_j \cdot t_C$ - загрузка памяти заявками из потока φ_j . Следовательно, $u_j = \frac{a_o}{1-\rho_j} + \rho_j$, где ω_0 в

выражениях является остаточным временем обслуживания заявок из потока φ_j . Оно определяется как доля времени, в течение которого прибор будет занят заявками из этого потока, т.е. $\omega_0 = \rho_j$, следовательно,

$$u_j = \frac{\rho_j}{1-\rho_j} + \rho_j.$$

Поток команд обращения i -го микропроцессора к локальной памяти моделируется потоком заявок φ_i . Заявка, заставшая СМО $S_i (i=2, \dots, n+1)$ занятым, становится в одностую очередь и ожидает до тех пор, пока не будут обслужены заявки из потока ψ_i , поскольку они обладают более высоким приоритетом. Местом в очереди является непосредственно процессор (один из его регистров), выдавших заявку на обращение в локальную память, т.к. он переходит в режим ожидания, который будет длиться до тех пор, пока память не освободится. Таким образом, моделью процесса обмена с локальной памятью является СМО с ограниченной длиной очереди ($R=2$). В соответствии с (3.21) время пребывания заявки и, следовательно, время обращения к локальной памяти t_{ML} составит:

$$t_M = \frac{(1-\rho_i)\rho_i}{\gamma_i(1-\rho_i^3)} + \frac{(1-\rho_j)\rho_i^2}{\gamma_i(1-\rho_i^3)}.$$

Преобразив, получим $t_M = \frac{\rho_i(1-\rho_i)\rho_i^2}{\gamma_i(1-\rho_i^3)}$, где $\rho_i = \gamma_i t_C$; $\gamma_i = N_{MLi}/N_i$. Здесь

N_{MLi} - число команд обращения микропроцессора в локальную память; N_i - общее число команд в решаемой задаче i -м вычислительным модулем.

Общее время пребывания заявки из потока φ_i в сети определится как $U = \alpha_i u_i + \alpha_j u_j$, следовательно, время обращения в общую память составит

$$t_M = \frac{\alpha_i \rho_i}{\lambda_i(1-\rho_i)} + \frac{\alpha_j \rho_j}{1-\rho_j} + \alpha_j \rho_j.$$

Системы с распределённой памятью, в межмодульном интерфейсе которых использован принцип связи с буферизацией сообщений, отличается наличием буферного накопителя в каждом модуле памяти ёмкостью R . Модель будет отличаться от предыдущей наличием ограниченной очереди для заявок из потока ψ_j перед СМО $S_j (j=2, \dots, n+1)$.

При исследовании системы с помощью предложенной модели следует учесть, что дисциплина обслуживания перед СМО S_j может быть либо *FIFO*, либо приоритетная. Во втором случае более высокий приоритет следует давать заявкам из потока $\psi_j (j=2, \dots, n+1)$, чтобы не перегружать ОШ повторными передачами сообщений, получивших отказ в обслуживании СМО S_2, \dots, S_{n+1} из-за ограниченной длины очереди. Чаще всего число мест в очередях как для потока χ_j , так и для потока ψ_j , ограничивают до одного.

В заключение следует отметить, что построение модели вычислительной системы является достаточно трудоёмким процессом, т.к. требует тщательного изучения вычислительных процессов. Однако на этом создание модели не заканчивается. Возникают вопросы – достоверна ли модель; насколько хорошо она представляет моделируемую систему. Наилучший ответ на эти вопросы даст реальная вычислительная система. Однако такой ответ можно получить только через годы, которые уйдут на её изготовление. Самый простой метод проверки достоверности заключается в построении некоторой другой модели, например, марковский, результаты которой не должны противоречить полученным результатам.

Другой вопрос связан с точностью модели и её калибровкой. Точность оценивается разницей между реальными характеристиками системы (например, реальной производительностью) и характеристиками, полученными моделью. Чем меньше ошибка, тем точнее модель. Однако, как и в первом случае, на проверку точности могут уйти годы. Наиболее приемлемым методом проверки точности является построение другой модели; более детально описывающей вычислительную систему. В качестве такой модели может выступать, например, имитационная.

Если точность модели неудовлетворительна, то модель должна быть изменена, а процесс проверки повторён. Эта операция называется калибровкой. Процедура калибровки оканчивается, как только исследователь найдёт точность модели удовлетворительной. Только после этого она может быть рекомендована для проведения исследований класса вычислительных систем, на которые модель направлена.

