

ГЛАВА 5. ОСОБЕННОСТИ ПРЕДМЕТНОЙ ОБЛАСТИ ПОЛИГРАФИИ

Исследование полиграфической предметной области и представление ее особенностей рассматриваются в этой главе, в основном, с точки зрения допечатного производства (ДП). Существует три основных способа реализации допечатных процессов, которые связаны с использованием различных технологий:

Computer-To-Film — допечатный процесс, в результате которого изготавливаются фотоформы (фотопленки). Их, соответственно, используют для создания печатных форм.

Computer-To-Plate — допечатный процесс, при котором изготавливаются печатные формы без промежуточного этапа изготовления фотоформ.

Computer-To-Press — допечатный процесс, в результате которого печать производится непосредственно с ЭВМ.

Самым быстрым и автоматизированным способом является технология *Computer-To-Press*. На рис. 5.1 показаны схемы допечатных процессов для всех трех технологий [Самарин, 2000].

В работе рассматривается только технология *Computer-To-Film*. Данный способ ДП наименее автоматизированный и требует наибольшего контроля с точки зрения ЛПП. Как показано на рис. 5.1. технология *Computer-To-Film* состоит из восьми этапов. Будем относить к ДП только первые пять из них:

- ввод и обработка данных полного оттиска;
- растривание с помощью RIP;
- экспонирование и проявление полных пленок;
- цветопроба пленки;
- проверка и корректировка пленок.

Первый этап ДП является самым продолжительным и трудоемким в полном цикле работ при выполнении большинства заказов. Разобьем этот этап на пять подэтапов:

- принятие заказа;
- формирование задания (технологической карты);
- распределения заказа между сотрудниками;
- выполнение полного цикла работ;
- контроль.

На этапе *принятия заказа* определяются параметры будущего издания, объем тиража, тип печати, сроки выполнения, исходные материалы и стоимость работ. Обычно эту работу выполняет руководитель фирмы (подразделения) или другое ЛПП.

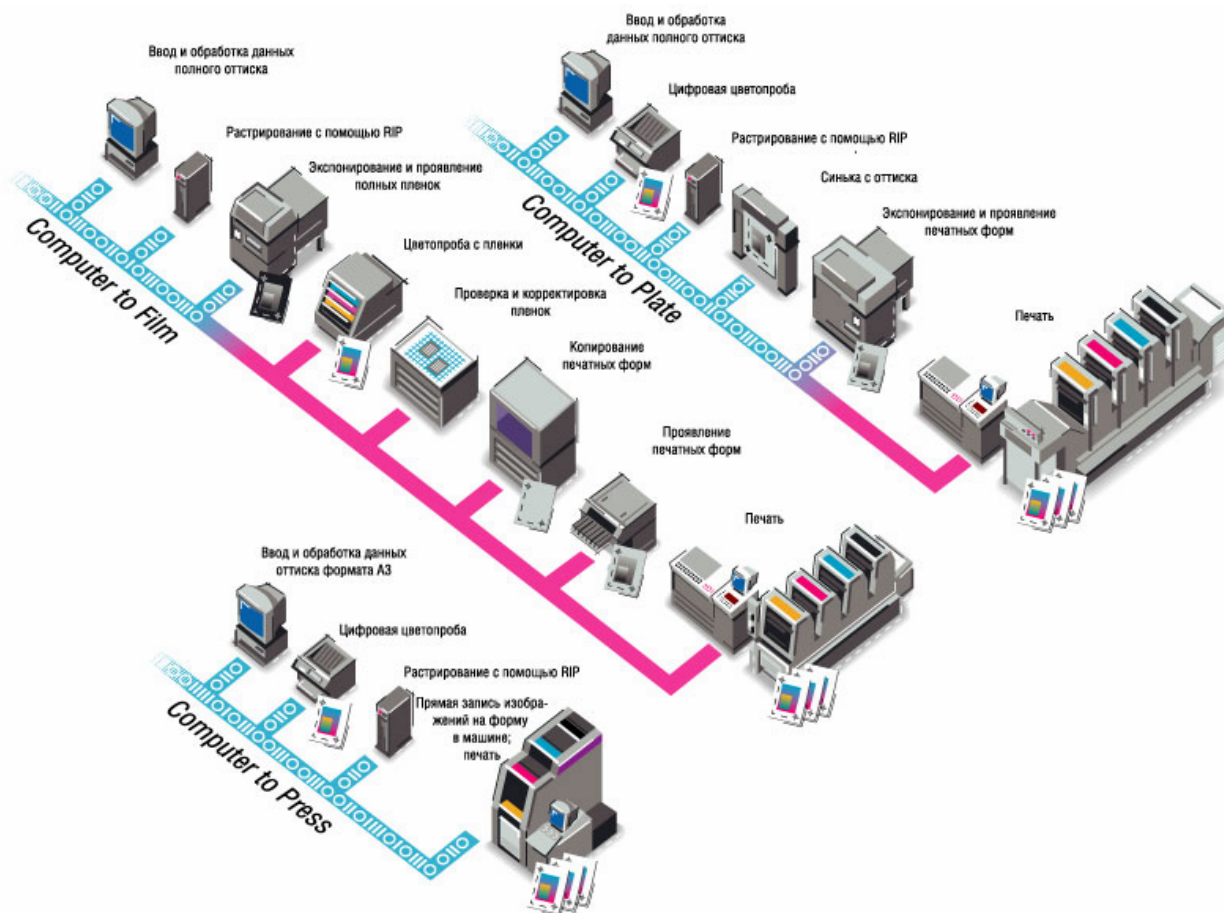


Рис. 5.1. Технологии допечатной подготовки.

На этапе *принятия заказа* определяются параметры будущего издания, объем тиража, тип печати, сроки выполнения, исходные материалы и стоимость работ. Обычно эту работу выполняет руководитель фирмы (подразделения) или другое ЛПР.

На этапе *формирования задания* передаются исходные данные, определяется точное задание на выполнения заказа, которое согласовывается с заказчиком.

На этапе *распределения заказа между сотрудниками* производится оценка временных и качественных возможностей сотрудников на выполнения отдельных работ, необходимых для реализации заказа. Осуществляется планирование и распределение ресурсов. Эту деятельность выполняет ЛПР.

На этапе *выполнения полного цикла работ* сотрудники выполняют поставленные задачи в соответствии с выданными указаниями.

Этап *контроля* осуществляется в течение выполнения полного цикла работ и после его завершения.

Систему допечатного производства условно можно разделить на три основные подсистемы (рис. 5.2).

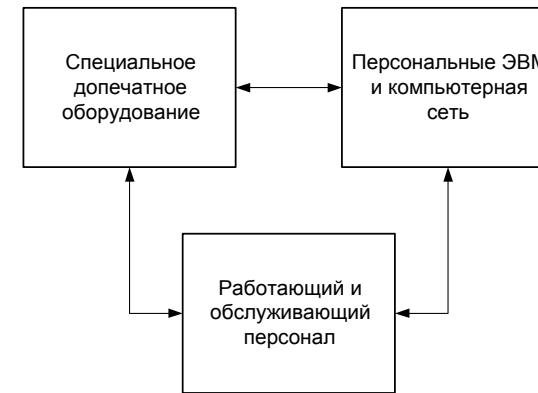


Рис. 5.2. Основные подсистемы допечатной подготовки.

В состав специального допечатного оборудования входят: сканеры, принтеры, фотонаборные аппараты (ФНА), проявочные машины, денситометры, цветопроба и др. Каждое из этих устройств можно рассматривать как обслуживающий аппарат. Для построения формальной модели устройств можно воспользоваться ТМО (аналитическим подходом) или теорий имитационного моделирования.

Для компьютерной сети можно использовать приближенные аналитические модели. Это связано с тем, что подсистема в очень малой степени зависит от так называемого “человеческого фактора”. В ней допустимы ошибки, так как время на их исправление очень мало. Подсистема мало изменяема, так как смена оборудования или программного обеспечения происходит достаточно редко [Кульгин, 1999].

Тем не менее, компьютерная подсистема может оказаться более сложной, чем подсистема допечатного оборудования. Для правильной работы сети необходимо сделать большое количество настроек. Ее параметры зависят от конкретных протоколов, операционных систем, установленных программ, аппаратных характеристик компьютеров и устройств связи.

5.1. Применение теории массового обслуживания в полиграфии

Возможность применения теории массового обслуживания для исследования предметной области определяется следующими факторами:

1. Количество заявок в системе (которая рассматривается как СМО) должно быть достаточно велико (массово) [Гнеденко, 1966].

2. Все заявки, поступающие на вход СМО, должны быть однотипными [Ивченко, 1982].

3. Для расчетов по формулам необходимо знать законы, определяющие поступление заявок и интенсивность их обработки. Более того, потоки заявок должны быть пуассоновскими [Павловский, 2000].

4. Структура СМО, т.е. набор обслуживающих аппаратов (ОА) и последовательность обработки заявки, должна быть жестко зафиксирована.

5. Необходимо исключить из системы субъектов или описывать их как ОА с постоянной интенсивностью обработки [Таха, 1985].

К перечисленным выше ограничениям можно добавить еще одно, оказывающее сильное влияние на размерность и сложность математической модели.

6. Количество используемых приоритетов должно быть минимальным. Приоритеты заявок должны быть постоянными, т.е. они не могут меняться в процессе обработки внутри СМО.

Массовость заявок. В работе [Ефимов, 1998] отмечается следующее: “Анализ полиграфического производства показывает, что теория массового обслуживания может быть использована для анализа различных производственных ситуаций. Первой из них является задача анализа функционирования издательств и типографий как систем массового обслуживания при поступлении в них случайных заявок на издание полиграфической продукции. За-

явки могут поступать в случайные моменты времени и могут обладать различной степенью сложности и различным объемом, что приводит к случайному времени исполнения заявок. Применение теории массового обслуживания позволяет оценить пропускную способность издательства или типографии, количество заказов, находящихся в производстве, время пребывания заявок и т.д.”.

Если рассматривать заказ как заявку в СМО, то количество заказов должно быть массовым. В средних допечатных полиграфических фирмах в среднем обрабатывается около 20 заказов в день. Ежедневный приток новых заказов в среднем составляет не более 25% [Сейфулин, 2000б]. Одна печатная машина редко за один день печатает более трех заказов. Аналогичная ситуация возникает и в фирмах, занимающихся послепечатной обработкой [Ефремов, 2002].

Из выше сказанного следует, что о массовости заказов можно говорить только при рассмотрении продолжительных интервалов времени (недели, месяцы). Для оценки производительности полиграфического предприятия чаще всего достаточно оценить небольшой промежуток времени. В этом случае использование аналитического подхода не оправдано. Для крупных полиграфических корпораций и для фирм, ориентированных на долгосрочные заказы использование ТМО может оказаться более эффективным.

Преодоление ограничения массовости возможно за счет рассмотрения в качестве заявок не заказов, а его составляющих частей. Например, можно представлять заказ как множество страниц.

Однотипность заявок. Проблема однотипности заявок является одним из основных препятствий для использования ТМО в полиграфической предметной области. Многие полиграфисты считают, что каждый заказ является уникальным, поэтому его нельзя типизировать. Если считать, что заказ имеет более 100 характеристик, каждая из которых может принимать несколько значений, то нельзя не согласиться с этим утверждением.

Преодоление этой проблемы заключается в разбиении заказа на составные части (страницы, технологические операции и др.) и пренебрежении некоторыми характеристиками. Например, задание плотности бумаги, типа и цвета кожи обложки не оказывает серьезного влияния на процесс печати. Параметры заказа, определяющие количество цветов (красок) и формат издания, наоборот, прямо определяют и возможность, и время печати.

В общем случае неучтенные характеристики заказа могут рассматриваться как необходимые ресурсы. Наличие или отсутствие ресурсов может оказывать серьезное влияние на последовательность и время ре-

шения задачи. Например, заказы, для которых имеется достаточное количество ресурсов, могут иметь более высокий приоритет. Сложностью такого подхода является то, что приоритеты могут меняться динамически в процессе работы.

Если рассматривать заказ как совокупность технологических операций, на каждую из которых существует норма, то разнородность заявок можно представить в виде нескольких простейших заявок. При этом количество простейших заявок зависит от технологической операции, объема материала и коэффициента сложности.

Еще одной сложностью, вызванной разнородностью, является наличие ОА, которые могут обслуживать только определенные типы заявок. Решение этой проблемы заключается в разделении соответствующих потоков заявок и рассмотрении системы в виде множества независимых последовательно или параллельно связанных СМО. Недостатками этого подхода являются: невозможность точного определения процесса окончания обработки заказа, так как заявки привязаны только к технологическим операциям; необходимость генерации и уничтожения заявок на стыках отдельных систем; использование ОА только для обработки одного типа заявок.

Законы поступления заявок. В ТМО, в основном, используются пуассоновские потоки заявок (заказов, технологических операций, компонент заказов). Для расширения области применения ТМО используют различные методы сведения произвольных потоков к пуассоновскому (дифференциальный и интегральный методы, метод Кендалла [Кофман, 1965], асимптотические методы [Ивченко, 1982]). Часто, не зная законы распределения, аналитики рассматривают самые “худшие” случаи, когда число поступающих заявок экспоненциально растет. Однако такой подход позволяет дать только грубую оценку системы.

Для того чтобы определить потоки заявок и интенсивность их обработки необходимо иметь статистические данные. В настоящее время количество исследований в полиграфии на данную тему крайне мало. Единственным источником для анализа полиграфической фирмы может служить собственная статистика, набранная за значительный период времени. Большинство малых и средних фирм оформляют и рассчитывают технологические карты только в бумажном виде. Кроме того, не документируется информация об интенсивности обработки, количестве и причине возникающих ошибок, сложности работы и т.д. Все это служит серьезным препятствием для использования ТМО.

Фиксированная структура. Для описания системы с помощью ТМО необходимо, чтобы она имела фиксированную структуру и последовательность обработки заявок. Необходимо отметить, что фиксированная последо-

вательность движения заявок не означает полную детерминированность всех процессов обработки. Например, если заявка находится в очереди, которую обслуживают несколько устройств (ОА), то процесс выбора ОА в общем случае является стохастическим. Однако количество ОА для одной очереди, последовательность и направление движение заявок не может меняться.

На практике последовательность выполнения заказа может сильно изменяться. Например, если невозможно осуществить сканирование материалов (зарезервировать ОА₁) или распечатку исходных данных для вычитки (использовать ОА₂), то можно осуществить ввод данных (ОА₃), верстку (ОА₄) и т.д. Часто один ПК (ОА) используется для выполнения различных технологических операций. При переполнении одной из очередей к ОА, ее необходимо сделать приоритетной, а заявки из других очередей перенаправить в другим ОА. Более того, в классическом представлении к ОА должна иметься только одна очередь. Тогда заявки, размещенные в очередь, должны определенным образом сортироваться либо с помощью динамического изменения приоритетов, либо с помощью других механизмов¹.

Решением этой проблемы является создание множества математических моделей, фиксация структуры в системе, рассмотрение приближенных случаев. Если количество возможных состояний системы велико, то размерность и сложность математической модели сильно возрастает и использование ТМО не оправдано.

Представление субъектов. Субъекты (персонал) можно представлять в виде ОА с определенной интенсивностью обработки. Их производительность следует определять из статистики, накопленной за продолжительный промежуток времени. Сложностью этого подхода является наличие в производстве новых субъектов (с неизвестной производительностью); постоянное, но неравномерное влияние опыта работника на его производительность; уникальность каждого субъекта.

Альтернативным вариантом задания интенсивности обработки заявки субъектом является использование нормативов. В этом случае математическая модель будет соответствовать “идеальному” производству. Такой подход можно применять для анализа крупных предприятий, где каждый работник выполняет (должен выполнять) один вид работы за фиксированное время. Результатами анализа может являться проверка адекватности нормативов и возможность их изменения.

Часто моделируемую систему исследуют на работоспособность в критических ситуациях: при наличии большого числа заказов, при возникновении большой срочности, при отсутствии или незначительном

¹ Более подробно этот вопрос рассматривается в параграфе, посвященном технологии виртуальных транзактов.

количестве работы и т.д. В этих случаях поведение субъектов крайне трудно спрогнозировать.

Представление субъектов в полиграфии, особенно в допечатном производстве, имеет специфику. Один сотрудник может выполнять различные виды работы с различным качеством. Отсюда следует, что субъект может обладать вектором специальностей, и для передачи ему работы необходимо оценивать его профессиональные качества и загрузку. Таким образом, переход заявки от одного ОА к другому не является случайным и зависит от многих параметров. Представление зависимостей такого рода в СМО крайне затруднительно.

Приоритеты. Использование приоритетов в полиграфической предметной области необходимо для задания срочности заказов (заявок), для формирования общей очереди к устройствам, выполняющим различные виды работ и т.д. Особенностью использования приоритетов является то, что они могут изменяться динамически в процессе работы. Это связано с тем, что заказы должны выполняться в течение определенного времени, и система должна работать без отказов заявок после их принятия к обработке. Ниже показано, что наличие приоритетов и невозможность отказа от заявки после ее принятия на обработку переводит СМО в разряд замкнутых.

5.2. Особенности построения аналитических моделей

Рассмотрим построение модели устройств с точки зрения теории массового обслуживания на примере сканера. В большинстве случаев сканер представляет собой одноканальное устройство, т.е. имеет только один канал обслуживания. Под каналом обслуживания будем понимать возможность выполнения одной задачи.

Будем считать, что обслуживающее устройство (сканер) имеет неограниченное количество источников заявок. На практике допечатная фирма работает на постоянно изменяющемся рынке заказчиков. Даже если есть небольшое количество постоянных заказчиков, вероятность появления заявки от нового заказчика очень велика. Кроме того, необходимо учитывать большое разнообразие заявок. Следовательно, наиболее подходящей моделью является одноканальная разомкнутая СМО.

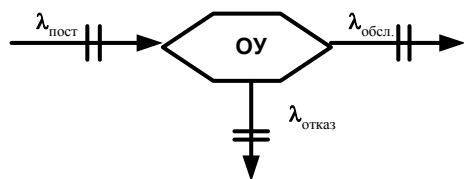


Рис. 5.3. Одноканальная разомкнутая СМО.

Для расчета можно принять, что очередь имеет неограниченный размер. Это означает, что необходимо обслужить все заявки, поступающие в очередь. Каждая заявка должна быть выполнена в определенные сроки, поэтому СМО должна содержать заявки с приоритетами. Приоритеты бывают абсолютные и относительные.

При относительном приоритете требование старшего приоритета, заставшее все каналы обслуживания занятыми, становится в очередь впереди требований младшего приоритета и дожидается освобождения канала.

В случае абсолютного приоритета требование, заставшее каналы занятыми, прерывает обслуживание требования младшего приоритета или становится в очередь, если все каналы обслуживают заявки со старшим приоритетом. Для этого варианта существует три разновидности:

- требование, обслуживание которого прервано, возвращается в очередь, и при возобновлении обслуживания учитывается время, ранее затраченное на обслуживание;
- прерванное требование возвращается в очередь, но затраченное время не учитывается при повторном обслуживании;
- требование младшего приоритета, обслуживание которого прервано, теряется.

На практике, заказы Z для сканирования можно разбить на r составных частей z_i . Каждую часть можно обрабатывать отдельно.

$$Z = \{z_1, z_2, \dots, z_r\}$$

Потеря заявок неприемлема для данной предметной области, так как может принести значительные расходы. Относительные приоритеты не позволяют выполнить срочный заказ, если в обслуживании находится большая работа, поэтому остаются два варианта с использованием абсолютных приоритетов. Выбор одного из них зависит от времени выполнения нового заказа T , требуемых для этого ресурсов Res , исполнителя Hu и др. Если процент выполненной работы меньше какого-нибудь критерия Kr , то заявка обслуживается заново полностью.

$$Z_i/Z < Kr, \quad Kr = \text{func}(T, Res, Hu \dots),$$

где Z_i — доля выполненной работы.

Основная сложность математической модели такой СМО заключается в том, что заявка с малым приоритетом может быть не обслужена. Для избежания таких ситуаций необходимо менять приоритеты во время обслуживания.

Главным критерием для выставления приоритетов, как отмечалось выше, является время выполнения заказа. Отсюда следует, что нужно использовать СМО с ограниченным временем ожидания (рис 5.4).

В рассматриваемой СМО не должно быть отказов, а это значит, что поток отказов по времени выполнения заявки $\lambda_{отк.ввз.}$, нужно перенаправить и добавить в поток срочных заказов $\lambda_{срочно}$ (рис. 5.5). В результате этих операций получится суммарный поток $\lambda_{\Sigma срочно}$. Объединение потоков переводит СМО в разряд замкнутых.

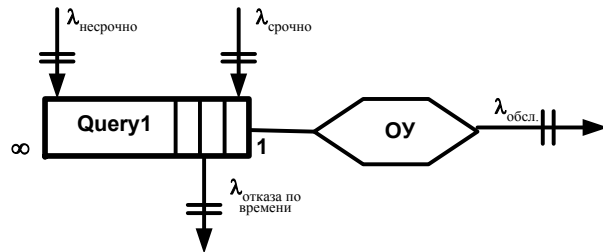


Рис. 5.4. Одноканальная разомкнутая СМО с бесконечной очередью и ограничением по времени.

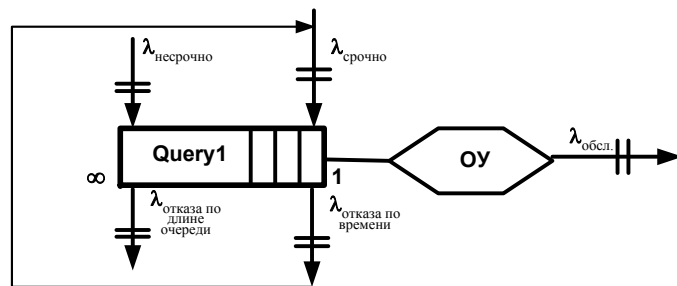


Рис. 5.5. Замкнутая одноканальная СМО с бесконечной очередью и ограничением по времени.

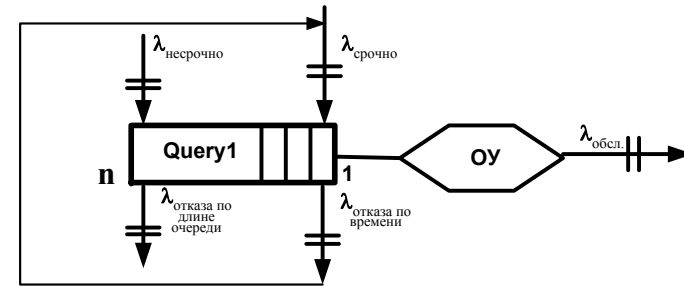


Рис. 5.6. Замкнутая одноканальная СМО с ограниченной очередью, и ограничением по времени.

Такая схема организации требует ограничения очереди таким образом, чтобы все заказы были выполнены вовремя. Размер очереди в этом случае уменьшится и будет равен n . Это приведет к появлению потока отказов по длине очереди $\lambda_{отк.до}$ (рис. 5.6). Если в системе имеется несколько типов приоритетов, то схема СМО значительно усложнится.

Итак, можно сделать вывод, что очередь должна быть конечной, из-за того, что в ней есть приоритеты, ограничение во времени и необходимость обслужить все заявки. Построение аналитической модели с приоритетами очень сложно, особенно если их более двух. Для упрощения математической модели можно воспользоваться одноканальной разомкнутой СМО с ограниченной очередью размером m .

Основными параметрами этой модели СМО являются абсолютная пропускная способность A (число заявок в единицу времени) и вероятность отказа заявки $P_{отк}$.

$$P_{отк} = [(\lambda/\rho)^{m+1} (1 - \lambda/\rho)] / [1 - (\lambda/\rho)^{m+2}]$$

$$A = \lambda (1 - P_{отк}) = \lambda (1 - [(\lambda/\rho)^{m+1} (1 - \lambda/\rho)] / [1 - (\lambda/\rho)^{m+2}]),$$

где λ — интенсивность поступления заявок;

ρ — интенсивность обслуживания заявок.

Средняя длина очереди (количество элементов, ожидающих обслуживания) ω , среднее время ожидания заявки в очереди T_ω и время пребывания заявки в СМО T_q являются важными характеристиками для СМО с очередями.

$$\omega = (\lambda/\rho)^2 \{1 - (\lambda/\rho)^m [m+1 - m(\lambda/\rho)]\} / [1 - (\lambda/\rho)^{m+2}] (1 - \lambda/\rho)$$

$$T_\omega = \omega / \lambda$$

$$T_q = T_\omega + A/\lambda\rho$$

Возникает закономерный вопрос о соотношении длин очередей n и m . Оно зависит от процентного содержания работ с разными приоритетами и от времени их выполнения.

Для простоты положим, что существует два типа приоритетов заказов. Введем некий коэффициент ξ , который характеризует соотношение между потоками срочных ($\lambda_{\Sigma\text{срочно}}$) и несрочных заказов ($\lambda_{\text{несрочно}}$).

$$\xi = (\lambda_{\Sigma\text{срочно}}) / (\lambda_{\text{несрочно}}) \quad (*)$$

$$(\lambda_{\Sigma\text{срочно}}) = \xi (\lambda_{\text{несрочно}})$$

Чтобы обеспечить эффективность работы, время обработки всех заказов должно быть больше, чем интенсивность заявок, т.е.

$$1/\rho < 1/(\lambda_{\Sigma\text{срочно}} + \lambda_{\text{несрочно}})$$

$$(\lambda_{\Sigma\text{срочно}} + \lambda_{\text{несрочно}}) < \rho$$

Подставляя в это соотношение формулу (*), можно получить:

$$\lambda_{\Sigma\text{срочно}} < \rho * \xi / (\xi + 1)$$

$$\lambda_{\text{несрочно}} < \rho / (\xi + 1)$$

Если известен оптимальный коэффициент ξ и интенсивность обработки заявок за единицу времени, то можно найти ограничения для потоков и соотношение между ними.

Попробуем оценить коэффициент ξ . Очевидно, что чем больше срочных заказов, тем меньше должна быть длина очереди, следовательно, коэффициент обратно пропорционален длине очереди.

$$\xi \sim 1/n \Rightarrow \xi = k/n \Rightarrow n_1 = k_1 * n_{50/50} / \xi, k = k_1 * n_{50/50}$$

где k — коэффициент пересчета длины очереди;

k_1 — коэффициент соответствия. Он определяется законами распределения поступления и обработки заявок;

$n_{50/50}$ — длина очереди при наличии 50% срочных и 50% несрочных заявок;

n_1 — новое значение длины очереди.

Если исходить из длины очереди $n_{\text{неср.}}$, в которой все заявки несрочные, то формула примет следующий вид:

$$n_1 = k_1 * n_{\text{неср.}} / (\xi + 1).$$

На рис. 5.7. приведен график, иллюстрирующий зависимость длины очереди от коэффициента ξ .

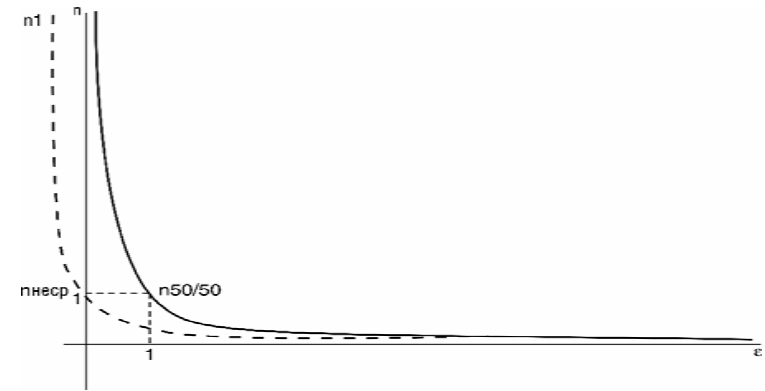


Рис. 5.7. График зависимости длины очереди от коэффициента ξ .

Рассмотрим предельные случаи:

При $\xi \rightarrow \infty, n \rightarrow 0$ очередь будет отсутствовать. Если используется два типа приоритетов, то граф переходов СМО будет иметь вид, представленный на рис. 5.8. На вход поступают два независимых потока заявок λ_1 и λ_2 . Требования первого потока обладают старшим абсолютным приоритетом.

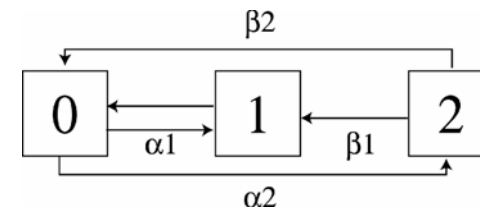


Рис. 5.8. Граф переходов СМО с отказами и двумя приоритетами.

Очередь отсутствует, поэтому заявка младшего приоритета, обслуживание которой прервано, теряется. Теряется также и заявка старшего приоритета, когда канал занят обслуживанием заявки этого же приоритета. Вероятность потери заявки $P_{\text{ном}}$ определяется следующей формулой:

$$P_{\text{ном}} = [\lambda_1(\lambda_1 + \lambda_2 + \rho_2) + \rho_1(\lambda_1 + \lambda_2)] / [(\lambda_1 + \lambda_2)(\lambda_1 + \lambda_2 + \rho_2)]$$

Если рассматривать заявки без приоритетов, то получится одноканальная СМО с отказами. Особенностью данной модели является работа без очереди, т.е. если заявка не может быть обслужена, она отбрасывается и в дальнейшей работе не участвует.

Такая ситуация вполне возможна для рынка полиграфических услуг. Например, имеется очень большое (для расчетов можно принять

бесконечное) количество заказов, тогда отсутствует необходимость создавать очередь. Такие ситуации могут возникать лишь на небольшие промежутки времени при сильном увеличении спроса. Работать по такой схеме не выгодно, так как можно потерять постоянных заказчиков.

$$A = \lambda\rho/(\lambda+\rho)$$

$$P_{отк} = \lambda/(\lambda+\rho)$$

При $\xi \rightarrow 0$, $n \rightarrow \infty$ очередь неограниченно возрастает, и модель системы приближается к одноканальной СМО с неограниченной очередью. Установившийся режим работы СМО может быть только в случае, когда интенсивность входного потока не больше интенсивности обслуживания.

$$\lambda/\rho \leq 1$$

Пропускная способность будет равняться интенсивности входного потока, а среднее число заявок в очереди m_k будет одной из важнейших характеристик данной СМО.

$$A = \lambda$$

$$m_k = (\lambda/\rho)^2 / (1 - \lambda/\rho)$$

При конкретных значениях, например, $\xi=20\%$, и притом, что суммарное время обработки срочных и несрочных заявок остается постоянным, длина очереди уменьшится почти на 20%. Данный расчет приблизительный и не учитывает законы распределения случайного появления заявок.

$$\xi=20\%, (\lambda_{\text{срочно}} + \lambda_{\text{несрочно}}) = \text{const}, k_1=1 \Rightarrow n_1=1/1.2=0.83$$

Подводя итог построению аналитических моделей подсистемы специального допечатного оборудования, можно сделать следующие выводы:

— Для расчета аналитической модели требуются сложные математические вычисления.

— Аналитические модели трудноизменяемы и не наглядны. Приближенные модели сильно типизированы и не позволяют учесть особенности устройств и работы с ними.

— Чтобы избежать этих недостатков, можно воспользоваться альтернативным вариантом и построить имитационную модель.

5.3. Технология виртуальных транзактов

На практике часто встречаются такие ситуации когда одна и та же работа (задача, транзакт) может быть успешно выполнена на различных устройствах. Например, ввод текста в ЭВМ можно осуществить на любом компьютере, ввод изображений — на тех ЭВМ, к которым подклю-

чены сканеры и т.д. Если количество обслуживающих устройств недостаточно для закрепления за каждым из них отдельного вида работы, то возникает необходимость их распределения.

Для выбора обслуживающего аппарата (ОА) транзакту (системе) необходимо сравнить альтернативные варианты. В системах имитационного моделирования для реализации такого взаимодействия можно использовать различные подходы. В работе рассматриваются три альтернативных подхода: прямое распределение транзактов, обратное распределение и технология виртуальных транзактов.

Прямое распределение. При таком подходе распределение транзактов из нескольких источников может быть случайным или поочередным, т.е. каждый раз выбирается новый источник.

В классическом представлении любой ОА (устройство) может иметь очередь. В СИМ Агента блок ОА неразрывно связан с «подблоком» очереди. В других широко известных системах (GPSS, SLAM II) каждый блок представляется отдельно. При большом (значительном) количестве транзактов, они должны размещаться в очередь к одному из возможных устройств. Поэтому, одним из критериев выбора ОА может служить длина очереди к нему.

$$\text{opt}(Ustr^1, Ustr^2, \dots, Ustr^n) \Rightarrow \min(Ustr^i.Q.Length)$$

где $Ustr^i$ – устройство (ОА);

$Ustr^i.Q$ – очередь к устройству;

$Q.Length$ – длина очереди.

Различные устройства могут иметь различную интенсивность обработки транзактов, которую также можно учитывать при размещении в очередь.

$$\text{opt}(Ustr^1, Ustr^2, \dots, Ustr^n) \Rightarrow \min \left(\frac{Ustr^i.Q.Length}{Ustr^i.\mu} \right),$$

где $Ustr^i.\mu$ – интенсивность обработки транзактов

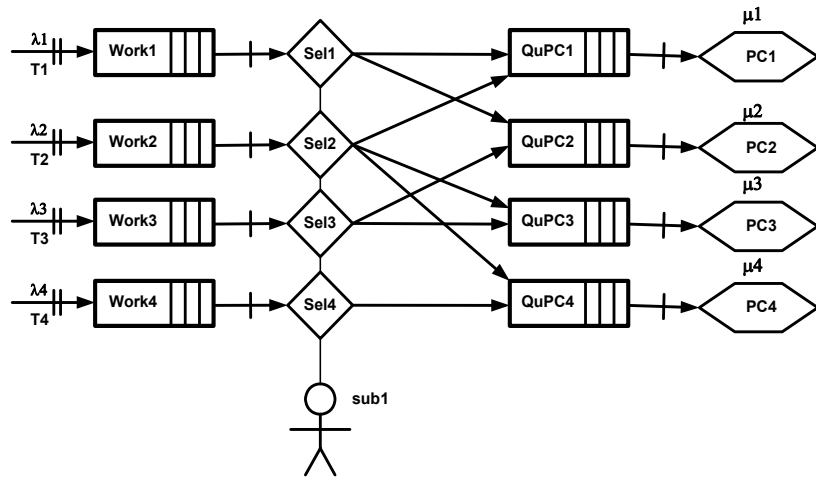


Рис. 5.9. Схема с прямым распределением.

На рис. 5.9 представлена схема обработки четырех типов транзактов — работ по сканированию (T1), вводу текста (T2), верстке (T3) и печати (T4). Транзакты размещаются в очереди на типы работ из потоков с интенсивностями $\lambda_1, \lambda_2, \lambda_3, \lambda_4$. Далее они распределяются селекторами Sel1, Sel2, Sel3, Sel4 в очереди к рабочим станциям PC1, PC2, PC3, PC4.

Селектор представляет собой особое устройство (субъекта), которое осуществляет распределение транзактов на дальнейшую обработку. В примере селекторы связаны с субъектом Sub1, который представляет собой лицо, принимающее решение по распределению работ.

Недостаток описанного подхода — выбор устройства эффективен только в случаях постоянной интенсивности обработки транзактов.

$$Ustr^i \cdot \mu = \mu_i = const$$

Если после размещения транзактов в очередь интенсивность обработки одного из устройств увеличится, то через некоторое время оно будет простаивать. В то же время потенциальные транзакты будут ожидать обслуживания в других очередях.

Возможны разные способы разрешения этой проблемы:

- осуществлять регулярное перераспределение транзактов по очередям;
- рассчитать среднюю интенсивность обработки транзактов;
- создать единую (неоднородную) очередь, и при каждом захвате осуществлять поиск подходящего транзакта в очереди.

Обратное распределение. Более приемлемую технологию распределения можно получить, немного изменив классический подход к многоканальным системам. Такие системы могут иметь одну общую очередь для нескольких эквивалентных устройств, т.е. устройства не имеют своей собственной очереди и обращаются к общей, когда заканчивают обработку очередного транзакта.

В предлагаемом варианте устройства обращаются к нескольким очередям и выбирают из них оптимальный транзакт (который имеет временной приоритет). Будем считать, что упорядочивание транзактов в очереди (сортировка) осуществляется во время обслуживания и не оказывает существенного влияния на временные задержки.

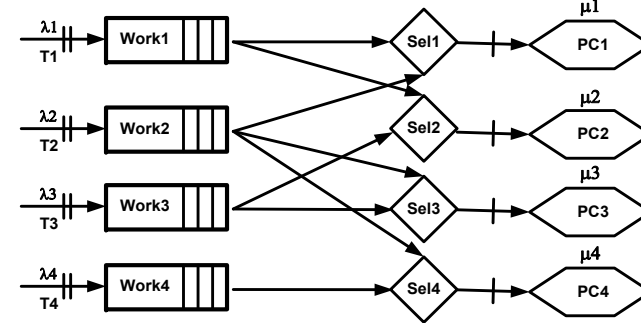


Рис. 5.10. Схема с обратным распределением.

На рис. 5.10 представлена схема такого взаимодействия. Выбор между очередями осуществляется селектором. Это приводит к одному из недостатков такого подхода, а именно, к нелинейности продвижения транзактов. Для того чтобы транзакт поступил на обработку устройство должно обратиться сначала к селектору.

Вторым, более существенным недостатком является необходимость выбора оптимального транзакта для каждого захвата. Это определяет временную задержку, зависящую от количества переборов.

$$T^{3д} = \tau_{if} * r$$

где $T^{3д}$ – время задержки;

τ_{if} – время одного перебора (сравнения);

r – количество переборов (очередей).

Фактически количество переборов определяется числом очередей, так как считается, что они отсортированы, и оптимальный транзакт в очереди уже стоит на первом месте. Это время можно сократить, если

осуществлять вычисление оптимального транзакта во время работы устройства. В общем случае неизвестно какое устройство освободится первым, поэтому нужно производить выбор транзакта для каждого устройства, что приведет к дополнительным затратам, а, возможно, и к недопустимым временным задержкам.

Третьим недостатком такого варианта является то, что невозможно осуществлять одновременную работу нескольких селекторов, которые связаны с общими очередями.

При такой технологии взаимодействия возникает вопрос об определении максимальной длины очереди для заданного типа работы. Для этого необходимо знать максимально допустимую длину очереди к устройству. Она легко определяется в первом варианте реализации распределения. Здесь будем считать, что все максимальные длины очередей к устройствам известны:

$$\forall (Ustr^i, i = \overline{1, n}) (Ustr^i.Q.Length = L^i),$$

где n – количество устройств (ОА).

Пусть каждый селектор имеет b входов, а каждая очередь — a выходов. Каждой линии связи сопоставляется вход селектора и выход очереди.

Если считать, что транзакты равномерно распределены по всем каналам и линиям связи, то можно получить следующее соотношение:

$$\forall (Sel^i, i = \overline{1, n}) \exists (b_i = |\{m^i\}|)$$

$$\forall (Qu^j, j = \overline{1, k}) \exists (a_j = |\{s^j\}|)$$

где n, k – количество селекторов и очередей;

b_i – количество входов селектора;

a_j – количество выходов очереди;

$\{m^i\}$ – множество входов i -ого селектора;

$\{s^j\}$ – множество выходов j -ой очереди.

$$\left. \begin{aligned} Can = \langle s^j, m^i \rangle \\ \forall (Can^r, r = \overline{1, p}) \exists (q_r = \frac{L_i}{b_i}) \end{aligned} \right\} \forall (Qu^j, j = \overline{1, k}) \exists (\{q_c^r\}, c = \overline{1, a_j}).$$

$$Qu^j.Length = \sum_{c=1}^{a_j} q_c^r,$$

где Can – линия связи, соединяющая селектор и очередь ;

p – количество каналов ;

q_r – количество транзактов, приходящихся на одну линию связи ;

$\{q_c^r\}$ – множество транзактов, определяемое линиями связи.

$Qu^j.Length$ – максимальный размер очереди.

Если рассчитать размеры очередей для схемы, приведенной на рис. 5.10, то можно получить следующие результаты:

$$Work1.Length = \frac{L^{PC1}}{2} + \frac{L^{PC2}}{2};$$

$$Work2.Length = \frac{L^{PC1}}{2} + \frac{L^{PC3}}{2} + \frac{L^{PC4}}{2};$$

$$Work3.Length = \frac{L^{PC3}}{2} + \frac{L^{PC2}}{2};$$

$$Work4.Length = \frac{L^{PC4}}{2};$$

В рассмотренном выше примере размеры транзактов считались одинаковыми и не влияли на размер очереди. Для учета различий между транзактами необходимо ввести понятия простейшего или элементарного транзакта, который выполняется за минимальное время и тогда все остальные можно выразить через соответствующие коэффициенты.

$$Qu^j.Length = \sum_{c=1}^{a_j} q_c^r * k^r, \quad k^r \geq 1;$$

где k^r – коэффициент, учитывающий тип (размер) транзакта.

Если подставить значения в описываемом примере, то получим следующие соотношения для длин очередей:

$$L^{PC1} = 2L; L^{PC2} = L; L^{PC3} = 2L; L^{PC4} = L; k^r = 1;$$

$$Work1.Length = 1.5L; Work2.Length = 2.5L;$$

$$Work3.Length = 1.5L; Work4.Length = 0.5L;$$

Из примера видно, что работы, которые могут быть выполнены на большем числе устройств, имеют большие длины очередей. Если количество транзактов различного типа приблизительно одинаково, то расчет максимально возможных размеров очередей не будет оптимальным. Это связано с тем, что при выборе учитывается время поступления транзактов. Чтобы выровнять длины очередей нужно определить вероятности поступления транзактов по каналам. Очевидно, что наибольшая вероятность поступления должна быть у транзактов из тех очередей, которые имеют меньшее число выходов.

Чтобы определить более точно размеры очередей нужно произвести дополнительные расчеты. Во-первых, необходимо вычислить коэффициент для каждого канала, зависящий от числа выходов очереди.

$$p_j = \frac{\min(\{a_j\}_k)}{a_j},$$

где p_j – коэффициент, учитывающий количество выходов;

a_j – количество выходов для очереди;

$\min(\{a_j\}_k)$ – значение для очереди с минимальным количеством выходов.

Во-вторых, в соответствии с полученными коэффициентами и количеством входов в селектор распределить множество транзактов, входящих на каждый канал.

$$\forall (Sel_i) \chi_i = \frac{1}{p_j^1 + p_j^2 + \dots + p_j^{b_i}},$$

где p_j – коэффициент входящего канала;

b_i – количество входящих каналов для i -ого селектора;

χ_i – коэффициент селектора.

$$Qu^j.Length = \sum_{c=1}^{a_j} p_j * \chi_i * L_i * k^r = p_j * \sum_{c=1}^{a_j} \chi_i * L_i * k^r$$

Произведем перерасчет длин очередей для примера, представленного на рис. 5.10:

$$L^{PC1} = 2L; L^{PC2} = L; L^{PC3} = 2L; L^{PC4} = L; k^r = 1;$$

$$p_1 = \frac{1}{2}; p_2 = \frac{1}{3}; p_3 = \frac{1}{2}; p_4 = \frac{1}{1};$$

$$\chi_1 = \frac{1}{p_1 + p_2} = \frac{6}{5}; \chi_2 = \frac{1}{p_1 + p_3} = 1; \chi_3 = \frac{1}{p_2 + p_3} = \frac{6}{5}; \chi_4 = \frac{1}{p_2 + p_4} = \frac{3}{4};$$

$$Work1.Length = \frac{1}{2} * \left(\frac{6}{5} * 2L + L \right) = 1.7L;$$

$$Work2.Length = \frac{1}{3} * \left(\frac{6}{5} * 2L + \frac{6}{5} * 2L + \frac{3}{4}L \right) = 1.85L;$$

$$Work3.Length = \frac{1}{2} * \left(\frac{6}{5} * 2L + L \right) = 1.7L;$$

$$Work4.Length = 1 * \frac{3}{4} * L = 0.75L..$$

Результаты, полученные за счет введения дополнительных коэффициентов, отражают влияние связей между очередями и селекторами. Неснятыми допущениями остаются одинаковая интенсивность поступления транзактов и постоянная длина очереди к устройству. Чтобы их снять необходимо регулировать значение коэффициента p_j и размер очереди L_i .

При изменениях интенсивности и длины очереди можно выбрать средние значения. Другим способом настройки коэффициентов может служить двухслойная нейронная сеть. Кроме того, многослойные сети могут быть использованы для многоуровневого распределения, но обсуждение этого вопроса выходит за рамки этой работы.

$$p'_j = p_j * k_\lambda = p_j * \frac{\min(\{\lambda_j\}_k)}{\lambda_j}; \lambda_j = const;$$

где p'_j – коэффициент, учитывающий интенсивность;

λ_j – интенсивность поступления транзактов;

$\min(\{\lambda_j\}_k)$ – минимальная интенсивность.

Технология виртуальных транзактов. Данный подход имеет очень яркое проявление в реальной жизни. Когда субъект имеет возможность выбрать одну из очередей, но не знает какая из них длинней и где быстрее обслуживают, он занимает место в нескольких очередях.

Виртуальные транзакты представляют собой копии реального транзакта, которые размещаются в несколько очередей одновременно. Когда осуществляется захват одного из транзактов, его копии удаляются. Что-

бы ускорить процесс поиска и удаления копий, в очередь размещаются не сами транзакты, а указатели на них. Устройство, обнаружив пустой или занятый транзакт, пропускает его копию. На рис. 5.11 представлена схема, на которой указаны очереди к устройствам с метками QuPC1, QuPC2, QuPC3, QuPC4 и очереди на четыре вида работ Work1, Work2, Work3, Work4.

После выхода из очереди Work транзакт автоматически дублируется и размещается во всех возможных очередях. Чтобы не возникло ситуации невозможности размещения всех копий виртуальных транзактов, очереди QuPC должны иметь неограниченную длину. Длины очередей Work могут быть вычислены аналогично с предыдущим примером.

Если в очередях QuPC осуществлять сортировку по времени (по срочности) во время работы устройства, то при этом можно автоматически удалять пустые виртуальные транзакты.

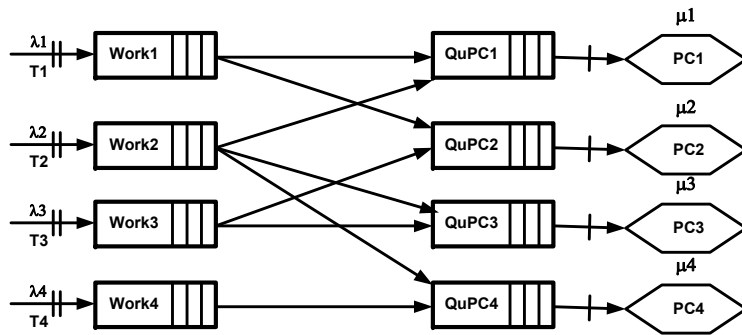


Рис. 5.11. Технология виртуальных транзактов.

Помимо расчета максимальной длины очередей типа Work, можно вычислять текущую длину очереди типа QuPC по следующей формуле:

$$QuPC^i.Length = \sum p_j * k^r, \quad k^r \geq 1,$$

где k^r – коэффициент, учитывающий тип (размер) транзакта;

p_j – коэффициент транзакта, пришедшего по каналу из очереди Work_j

В заключении следует отметить достоинства технологии виртуальных транзактов:

1. Возможность эффективного распределения транзактов в очереди к нескольким устройствам;

2. Возможность работы с неоднородными очередями и различными типами транзактов;
3. Оптимальное распределение по времени при условиях сильного изменения интенсивности обработки транзактов;
4. Минимальные временные затраты СИМ на распределение (ведение и удаление) транзактов.
5. Возможность прямого движения транзактов в системе.
6. Возможность расчета текущей и максимальной длины очереди.

К ограничениям технологии виртуальных транзактов следует отнести возможность ее применения. Для этого реальные прототипы транзактов должны иметь возможность быстрого перемещения от одного обслуживающего аппарата к другому.

Обычно количество ОА, на которых может быть выполнена задача, и множество типов работ незначительно. Поэтому вычислительные затраты классических подходов незначительны.

5.4. Стратегия отказа

Этот параграф посвящен вопросу автоматизации принятия решения о возможности выполнения заказа. При использовании СИМ для решения этой задачи необходимо ввести исходные данные и осуществить имитационное моделирование. Если оно закончится удачно, то заказ может быть выполнен. Наличие нескольких вариантов выполнения заказа (в разное время, на различном оборудовании) требует проведения серии имитаций.

Использование ЭС может обеспечить меньшие затраты на принятие решения. Например, если формат издания не соответствует печатной машине или ФНА, то нет необходимости моделировать весь ДП. Цель работы ЭС — получить ответ на вопрос о возможности выполнения заказа. В связи с этим правила и вопросы БЗ должны выбираться специальным образом, с помощью определенной стратегии, которая максимально быстро приводит к отказу (отрицательному ответу).

Рассмотрим формальное представление производственной системы. Вывод может производиться в несколько этапов. На каждом этапе имеются исходные данные (посылки), выходные данные (заклучения) и правило вывода, которое осуществляет переход между ними. Производственные системы впервые изобретены Постом в 1941 г. и имеют следующее представление [Герман, 1995]:

$$(t_1, t_2, \dots, t_n) / t$$

где t_1, t_2, \dots, t_n — посылки;

t — заключение.

Несколько изменим представление правил. Будем считать, что из посылок может следовать несколько заключений, т.е.

$$(x_1, x_2, \dots, x_n) / (y_1, y_2, \dots, y_m)$$

где x_1, x_2, \dots, x_n — посылки; y_1, y_2, \dots, y_m — заключения.

Вместо подстановок, предлагаемых Постом, введем на множестве посылок, заключений и правил операцию умножения.

$$\forall x_i, \forall x_{i+1} (x_i, x_{i+1}) \Leftrightarrow (x_i * x_{i+1}), \text{ где } i=1, n$$

$$\forall y_i, \forall y_{i+1} (y_i, y_{i+1}) \Leftrightarrow (y_i * y_{i+1}), \text{ где } i=1, m$$

Операция умножения обратна операции деления. Процесс сокращения членов знаменателя с членами числителя будем называть *операцией свертки вывода*. Для осуществления операции свертки необходимо, чтобы посылки и заключения могли быть приравнены.

$$\text{при } x_1 = y_1 \quad (x_1 * x_2 * \dots * x_n) / (y_1 * y_2 * \dots * y_m) = (x_2 * \dots * x_n) / (y_2 * \dots * y_m)$$

Будем считать, что имеется фиксированное множество продукционных правил $Prav$ размерностью p .

$$Prav = \{Prav_1, \dots, Prav_p\}$$

Вывод представляет собой суперпозицию правил, т.е. последовательное применение правил. Докажем, что при выбранном математическом представлении порядок не играет роли.

Теорема 1. Для любых двух или более правил их суперпозиция коммутативна.

$$\forall Prav_i, \forall Prav_j, \dots, \forall Prav_n$$

$$Prav_i * Prav_j = Prav_j * Prav_i$$

Доказательство: Предположим, что это не так, тогда для любых двух правил верно следующее:

$$\forall Prav_i, \forall Prav_j$$

$$Prav_i * Prav_j = V_1 \quad Prav_j * Prav_i = V_2 \quad V_2 \neq V_1 \Rightarrow$$

$$\frac{(x_{i1} * \dots * x_{in}) * (x'_{i1} * \dots * x'_{in})}{(y_{i1} * \dots * y_{im}) (y'_{i1} * \dots * y'_{im})} \neq \frac{(x'_{i1} * \dots * x'_{in}) * (x_{i1} * \dots * x_{in})}{(y'_{i1} * \dots * y'_{im}) (y_{i1} * \dots * y_{im})}$$

$$\frac{(x_{i1} * \dots * x_{in})(x'_{i1} * \dots * x'_{in})}{(y_{i1} * \dots * y_{im})(y'_{i1} * \dots * y'_{im})} \neq \frac{(x'_{i1} * \dots * x'_{in})(x_{i1} * \dots * x_{in})}{(y'_{i1} * \dots * y'_{im})(y_{i1} * \dots * y_{im})}$$

$$\forall x_i > 0, \forall x'_i > 0;$$

$$\frac{(x_{i1} * \dots * x_{in})(x'_{i1} * \dots * x'_{in})}{(x'_{i1} * \dots * x'_{in})(x_{i1} * \dots * x_{in})} \neq \frac{(y_{i1} * \dots * y_{im})(y'_{i1} * \dots * y'_{im})}{(y'_{i1} * \dots * y'_{im})(y_{i1} * \dots * y_{im})}$$

Используя операцию свертки, получаем, что $1 \neq 1$. Отсюда следует, что предположение неправильное, а теорема верна.

Теорема 2. Для любых правил их суперпозиция ассоциативна.

$$\forall Prav_i, \forall Prav_j, \forall Prav_n$$

$$Prav_i * (Prav_j * Prav_n) = (Prav_i * Prav_j) * Prav_n$$

Доказательство: Предположим, что это не так, тогда для любых трех правил будет верно следующее:

$$\forall Prav_i, \forall Prav_j, \forall Prav_n$$

$$Prav_i * (Prav_j * Prav_n) = V_1 \quad (Prav_i * Prav_j) * Prav_n = V_2$$

$$V_2 \neq V_1 \Rightarrow$$

$$\forall x_i > 0, \forall x'_i > 0; \forall x''_i > 0;$$

$$\frac{(x_{i1} * \dots * x_{in}) * \left[\frac{(x'_{i1} * \dots * x'_{in}) * (x''_{i1} * \dots * x''_{in})}{(y'_{i1} * \dots * y'_{im}) * (y''_{i1} * \dots * y''_{im})} \right]}{(y_{i1} * \dots * y_{im})} \neq$$

$$\neq \left[\frac{(x_{i1} * \dots * x_{in}) * (x'_{i1} * \dots * x'_{in})}{(y_{i1} * \dots * y_{im}) * (y'_{i1} * \dots * y'_{im})} \right] * \frac{(x''_{i1} * \dots * x''_{in})}{(y''_{i1} * \dots * y''_{im})}$$

$$\frac{(x_{i1} * \dots * x_{in})(x'_{i1} * \dots * x'_{in})(x''_{i1} * \dots * x''_{in})}{(x'_{i1} * \dots * x'_{in})(x_{i1} * \dots * x_{in})(x''_{i1} * \dots * x''_{in})} \neq$$

$$\neq \frac{(y_{i1} * \dots * y_{im})(y'_{i1} * \dots * y'_{im})(y''_{i1} * \dots * y''_{im})}{(y'_{i1} * \dots * y'_{im})(y_{i1} * \dots * y_{im})(y''_{i1} * \dots * y''_{im})}$$

Используя операцию свертки, получаем, что $1 \neq 1$. Отсюда следует, что предположение неправильное, а теорема верна.

Данные теоремы имеют большое значение, так как позволяют использовать правила в различном порядке, на разных этапах вывода. Будем считать, что возведение правила в степень равно самому правилу.

$$Prav_i * Prav_i = Prav_i^2 = Prav_i$$

Аналогичное утверждение справедливо для посылок и заключений. Введем понятие о срочном окончании вывода, при появлении которого в суперпозиции правил, результирующее заключение принимает фиксированное значение. Пусть это будет заключение y_0 . Можно принять, что значение y_0 равно 0, но при этом может возникнуть арифметическая ошибка деления на нуль.

Для расчетов примем, что данная операция возможна. Тогда результатом (заключением) вывода является отказ от выполнения дальнейших выводов O .

Теорема 3. Если при суперпозиции правил $Prav_1 \dots Prav_n$, хотя бы в одном правиле присутствует заключение y_0 , то суперпозиция приводит к отказу O .

$$\forall Prav_1 \dots \forall Prav_n,$$

$$\exists Prav_i = (x_1, x_2, \dots, x_n) / (y_1, y_2, \dots, y_0, \dots, y_m)$$

$$Prav_1 * Prav_2 * \dots * Prav_n = O.$$

Доказательство:

$$\forall Prav_i, \dots, \forall Prav_q$$

Из теоремы 1 и теоремы 2 следует, что

$$\begin{aligned} & \frac{(x_{i1} * \dots * x_{in}) * \dots * (x'_{i1} * \dots * x'_{in}) * \dots * (x_{q1} * \dots * x_{qn})}{(y_{i1} * \dots * y_{im}) * \dots * (y_0 * \dots * y'_{im}) * \dots * (y_{q1} * \dots * y_{qm})} = \\ & = \frac{(x_{i1} * \dots * x_{in}) \dots (x'_{i1} * \dots * x'_{in}) \dots (x_{q1} * \dots * x_{qn})}{(y_{i1} * \dots * y_{im}) \dots (y_0 * \dots * y'_{im}) \dots (y_{q1} * \dots * y_{qm})} = \\ & = \frac{(x_{i1} * \dots * x_{in}) \dots (x'_{i1} * \dots * x'_{in}) \dots (x_{q1} * \dots * x_{qn})}{y_0} = O \end{aligned}$$

Из этой теоремы вытекают важные практические следствия. Например, можно принять, что Y_0 соответствует отрицательному ответу на вопрос “Можно ли выполнить данный заказ”.

При продукционном выводе часто возникают такие ситуации, когда существует несколько вариантов применения последующих правил. Из теоремы 3 следует, что специфика (порядок) опроса на ввод исходных данных должна носить параллельный характер. Это означает, что сначала вводятся все независимые данные. После ввода каждого отдельного элемента они проверяются на наличие правил, которые сразу приведут к отказу.

Если Y_0 не найдено после проверки правил, ведущих к отказу за один такт, то проверяются правила, приводящие к отказу за два такта и т.д. Такую стратегию вывода будем называть *стратегией отказа*.

Для пояснения понятия такта разобьем все множество правил $Prav$ на r подмножеств $Prav_{III}$.

$$Prav = Prav_{III1} \cup Prav_{III2} \cup \dots \cup Prav_{IIIr}$$

Разбиение есть некая функция $F1$, которая зависит от исходных данных и от стратегии вывода. На каждом такте может быть применено одно из нескольких правил. Совокупность всех этих правил образует подмножество такта $Prav_{III}$. С каждым тактом уменьшается количество возможных правил. Если обозначить количество правил на такте L как результат функции разбиения, то можно записать следующее:

$$L1 > L2 > L3 > \dots > Lr$$

$$Li = Func(L_{i-1}, Strat, Id)$$

где Li — количество правил на i -ом такте;

Strat — выбранная стратегия;

Id — исходные данные.

Итак, если вывод осуществляется по стратегии отказа, то порядок выбора правил должен определяться близостью к заключению Y_0 .

Для ускорения процесса вывода можно заранее просчитать матрицы расстояний и переходов. Ниже приведена таблица переходов. Она показывает цепочки порядков применения правил, приводящих к отказу. Например цепочка $Prav_1 \rightarrow Prav_3 \rightarrow Prav_4 \rightarrow Prav_n \rightarrow 0$ будет иметь следующий вид (табл. 5.1).

	Prav ₁	Prav ₂	Prav ₃	Prav ₄	...	Prav _n
Prav ₁	0	0	1	0	0	0
Prav ₂	0	0	0	0	0	0
Prav ₃	0	0	0	1	0	0
Prav ₄	0	0	0	0	0	1
...	0	0	0	0	0	0
Prav _n	0	0	0	0	0	0

Табл.5.1. Таблица переходов.

В стратегии отказов нужно делать откаты, поэтому необходимо иметь таблицу обратных переходов (табл. 5.2). С помощью этой таблицы не придется пересчитывать заново разбиение множеств Li , нужно только исключить тупиковые варианты². При этом можно оценивать на каждом шаге отката близость к отказу².

	Prav ₁	Prav ₂	Prav ₃	Prav ₄	...	Prav _n
Prav ₁	0	0	0	0	0	0
Prav ₂	0	0	0	0	0	0
Prav ₃	1	0	0	0	0	0
Prav ₄	0	0	1	0	0	0
...	0	0	0	0	0	0
Prav _n	0	0	0	1	0	0

Табл. 5.2. Таблица обратных переходов.

Можно объединить обе таблицы, так как они являются симметричными относительно диагонали. Тогда таблица переходов примет следующий вид:

	Prav ₁	Prav ₂	Prav ₃	Prav ₄	...	Prav _n
--	-------------------	-------------------	-------------------	-------------------	-----	-------------------

² Будем считать, что из правила Prav_i нельзя перейти в правило Prav_{i-1}.

Prav ₁	0	0	1	0	0	0
Prav ₂	0	0	0	0	0	0
Prav ₃	-1	0	0	1	0	0
Prav ₄	0	0	-1	0	0	1
...	0	0	0	0	0	0
Prav _n	0	0	0	-1	0	0

Табл. 5.3. Общая таблица переходов.

Таким образом, определены пути возможных переходов. На каждом такте нужно выбирать следующее правило, но возникают такие ситуации, когда возможных путей несколько. Введем таблицу расстояний, в которой будут указываться минимальное количество тактов, приводящих к отказу.

	Количество тактов
Prav ₁	4
Prav ₂	0
Prav ₃	3
Prav ₄	2
...	0
Prav _n	1

$$A_i = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ \dots \\ a_n \end{pmatrix}$$

Табл. 5.4. Таблица расстояний.

Очевидно, что значения второго столбца этой таблицы будут меняться в процессе производственного вывода. Количество тактов зависит от посылок на предыдущих тактах. Для формализации представим табл. 5.4 в виде матричного оператора A_i . Чтобы изменять длину перехода в зависимости от исходных данных (посылок), представим их в виде матрицы коэффициентов M_i .

$$M_i = \begin{pmatrix} m_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & m_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & m_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & m_n \end{pmatrix}$$

Выбор оптимального пути перехода нужно высчитать с помощью произведения матриц коэффициентов и расстояний:

$$U_i = A_i * M_i$$

Вычислив новую матрицу ее необходимо отсортировать. Сортировка есть преобразование исходной матрицы U_i в новую U_i' . Под новой матрицей нужно понимать упорядоченную старую.

Очень вероятны случаи, когда из одного правила могут вести несколько эквивалентных путей. Для обозначения этой характеристики введем таблицу *тактов отказа*. Столбцы обозначают количество тактов, а в строках для каждого правила записывается количество эквивалентных путей для достижения отказа.

	1	2	3	4	...	N
Prav ₁	0	0	2	0	0	0
Prav ₂	0	0	5	0	0	1
Prav ₃	3	0	0	1	0	0
Prav ₄	0	0	7	0	0	1
...	0	4	0	0	0	0
Prav _n	0	0	0	3	0	0

Табл. 5.5. Таблица тактов отказа.

Выводы

В главе рассмотрены вопросы применения теорий массового обслуживания, имитационного моделирования и экспертных систем для представления процессов, протекающих в допечатном производстве.

В первом параграфе определяется возможность применения ТМО для полиграфии. Для этого выделяется ряд признаков, которыми должна обладать моделируемая система. В процессе анализа отмечается ограниченность применения ТМО из-за наличия в ней динамически изменяемых приоритетов, сильной неоднородности заявок, наличие большого числа субъектов, трудностей в определении потоков и т.д. Для преодоления этих ограничений предлагаются соответствующие решения.

Во втором параграфе рассматривается пример построения аналитической модели одного из устройств, используемых в ДП. Затрагиваются вопросы об определении длины очереди, ее зависимости от приоритетов, замкнутости СМО, выборе наиболее подходящей модели. Делаются выводы о сложности и нецелесообразности создания аналитической модели ДП.

В третьем параграфе рассматривается вопрос об оптимальном распределении заявок (транзактов) в СИМ, описывающей работу ДП. Сравниваются традиционные подходы с использованием одной очереди к устройству и предлагается технология виртуальных транзактов.

В четвертом параграфе решается задача автоматизации принятия решения о возможности выполнения заказа. Предлагается стратегия

отказа для ЭС, использующих производственную БЗ. Формулируется и доказывается несколько теорем. Описывается методика реализации стратегии отказов.

В целом можно отметить, что полиграфическое производство имеет ряд отличительных черт, реализация которых в системах имитационного или экспертного моделирования общего назначения крайне затруднительна. Для описания полиграфических процессов требуется разработка специализированного программного и технического обеспечения.