

# Измерительные фрагменты в моделях Петри телекоммуникационных сетей

Д.А.Зайцев

*Одесская национальная академия связи им. А.С.Попова  
Ул. Кузнечная 1, Одесса 65038, Украина  
<http://www.geocities.com/zsoftua>*

## Аннотация

Представлена технология измерения характеристик моделей Петри коммутируемых сетей с помощью специальных измерительных фрагментов раскрашенных сетей Петри. Особенности применения измерительных фрагментов изучены на примере определения времени отклика в модели коммутируемой локальной сети. Для автоматизированного построения модели в форме иерархической раскрашенной сети Петри использована инструментальная система CPN Tools. Модель отражает такие особенности функционирования локальной сети Ethernet как процедуры доступа CDMA, полнодуплексный режим работы, а также обеспечивает лёгкую масштабируемость сетей Петри при отображении структуры конкретных ЛВС.

**Ключевые слова:** ЛВС, коммутатор, измерительный фрагмент, время отклика, раскрашенная сеть Петри

## 1. Введение

Раскрашенные иерархические временные сети Петри [1] является мощным и удобным инструментом для построения и исследования моделей телекоммуникационных систем. Эти сети успешно применены для моделирования протоколов телефонной связи, новых поколений мобильных телефонов, компьютерных сетей Token Ring и Ethernet [1,2,3].

В настоящее время университетом Орхуса (Дания) в сотрудничестве с исследовательской лабораторией фирмы Microsoft представлен новый программный продукт – моделирующая система CPN Tools [4], которая призвана заменить известную моделирующую систему Design/CPN. CPN Tools свободно распространяется для использования в некоммерческих, в особенности, учебных целях и представляет собой интегрированную среду для графического ввода моделей, имитации их динамики и определения основных свойств. Кроме средств рисования сетей система содержит встроенный язык программирования ML предназначенный для описания атрибутов графических объектов. Следует отметить, что в проекте CPN Tools применена новая технология работы с графическими объектами, базирующаяся на стандартах Open GL, позволяющая создавать крупномасштабные модели в короткие сроки. Обеспечена совместимость двух систем: модели, ранее построенные в среде Design/CPN, могут быть автоматически конвертированы в CPN Tools.

При моделировании телекоммуникационной системы наряду с основной задачей обеспечения адекватности исследуемому объекту возникает задача измерения характеристик модели. Встроенные средства CPN Tools позволяют построить пространство состояний модели и определить такие свойства сети Петри как ограниченность, безопасность, отсутствие тупиков, живость [5]. Следует отметить, что модели реальных систем имеют количество состояний, измеряемое сотнями тысяч, что делает анализ пространства состояний практически неосуществимым. Кроме того, практическую значимость имеют нетривиальные характеристики модели, такие, например, как время отклика, количество коллизий, вероятности перегрузок. Имитация динамики модели и визуальное наблюдение позволяют выполнить лишь отдельные измерения в отладочных целях. Таким образом, имеется научная проблема, состоящая в организации измерения характеристик моделей Петри телекоммуникационных систем.

Следует отметить, что расширенные сети Петри представляют собой универсальную алгоритмическую систему [5]. При проектировании систем автоматизированного управления А.И.Слепцовым было предложено представлять в форме сети Петри как управляемую систему, так и алгоритм управления [6]. Были показаны способы реализации с помощью

расширенных сетей Петри основных логических и арифметических операций [7]. Эта методология была, затем, обобщена для временных сетей Петри, множество базисных операций которых включает специальную операцию временной задержки [8].

Целью настоящей работы является представление методологии измерения характеристик моделей Петри телекоммуникационных систем с помощью специальных измерительных фрагментов сетей Петри. Такой подход позволяет организовать измерение нетривиальных характеристик модели. Детальное изучение принципов построения измерительных фрагментов выполнено на примере измерения времени отклика в модели коммутируемой локальной сети. Отметим, что ранее представленная модель коммутируемой сети [2] была существенно модифицирована и доведена до промышленного уровня качества. Она обеспечивает реализацию процедур CDMA, полнодуплексного режима работы сети Ethernet, а также имеет модульную структуру, позволяющую отображать конкретные ЛВС с относительно небольшими трудозатратами.

## 2. Коммутируемая ЛВС

В последнее время сеть Ethernet (IEEE 803.x) стала самой распространённой ЛВС. С появлением гигабитных технологий открылся новый этап её популярности. Основным элементом сети в настоящее время являются коммутатор [9]; пассивное оборудование, такое как концентратор, служит лишь для соединения терминального и сетевого оборудования, также как и кабельная система. Терминальное оборудование ЛВС состоит из серверов и рабочих станций. Рабочие станции являются источниками запросов к серверам. Серверы выполняют запросы рабочих станций, направляя им результаты выполнения запросов. Причём взаимодействие сервера и рабочей станции может иметь более сложный алгоритм, определяемый спецификой задач, решаемых в среде ЛВС.

Задачей коммутатора является перенаправление входящих фреймов в порт, к которому подключено устройство назначения, задаваемое с помощью MAC-адреса. Заголовок фрейма содержит MAC-адреса источника и приёмника. Как правило, в настоящее время Ethernet работает в полнодуплексном режиме, обеспечивая одновременную передачу и приём фреймов. Для определения номера порта используется таблица коммутации, которая может быть как статической, так и динамической.

Исходной информацией для построения модели является структурная схема сети, представляющая сетевое и терминальное оборудование ЛВС, а также способ их подключения с помощью кабельной системы. Кроме того, необходима информация о быстродействии сетевых карт, портов коммутатора, временных характеристиках процессов взаимодействия рабочей станции и сервера. На основе этой информации определяются единицы измерения модельного времени и временные характеристики элементов модели.

Выполним построение модели ЛВС для структурной схемы, представленной на Рис. 1. ЛВС имеет архитектуру звезды и состоит из одного коммутатора (**SW1**), трёх концентраторов (**HUB1-HUB3**), а также двух серверов (**S1, S2**) и пяти рабочих станций (**WS1-WS5**). На схеме указаны также модельные значения MAC-адресов, которые могут быть получены с помощью простой нумерации реальных MAC-адресов сетевых карт.

### Scheme of sample switched LAN

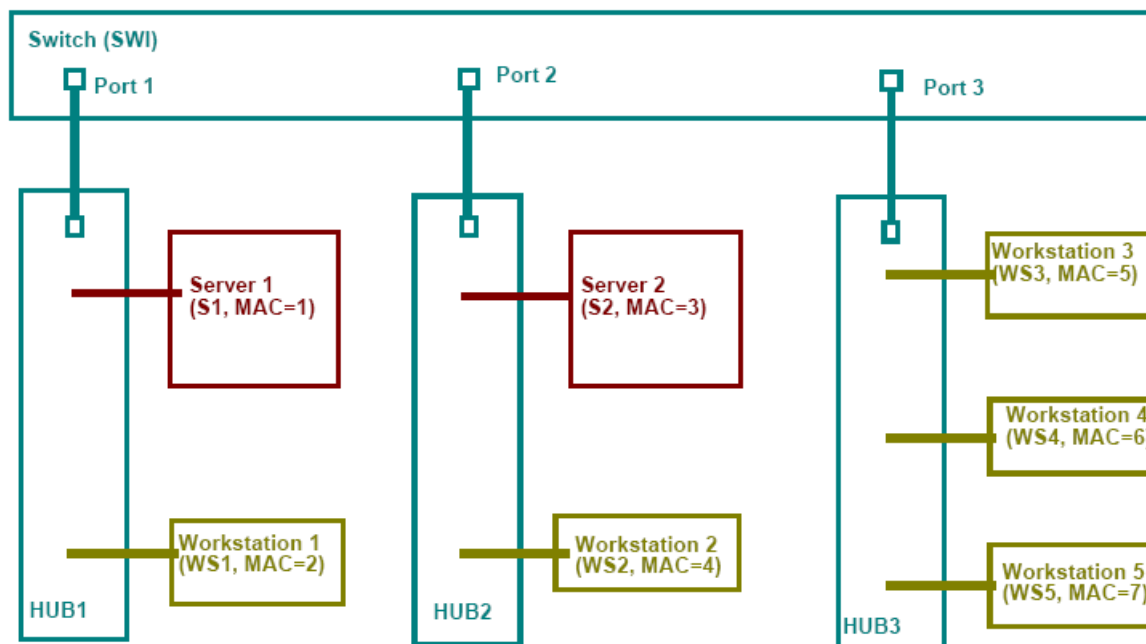


Рис. 1. Структурная схема ЛВС

### 3. Модель ЛВС

Модель ЛВС, структурная схема которой изображена на Рис.1, представляет собой иерархическую сеть, состоящую из пяти отдельных листов. На Рис. 2 изображён верхний уровень иерархии модели (**LAN**), на остальных листах представлены модели коммутатора (**SWI**), сервера (**S**), рабочей станции (**WS**) и измерительной рабочей станции (**MWS**).

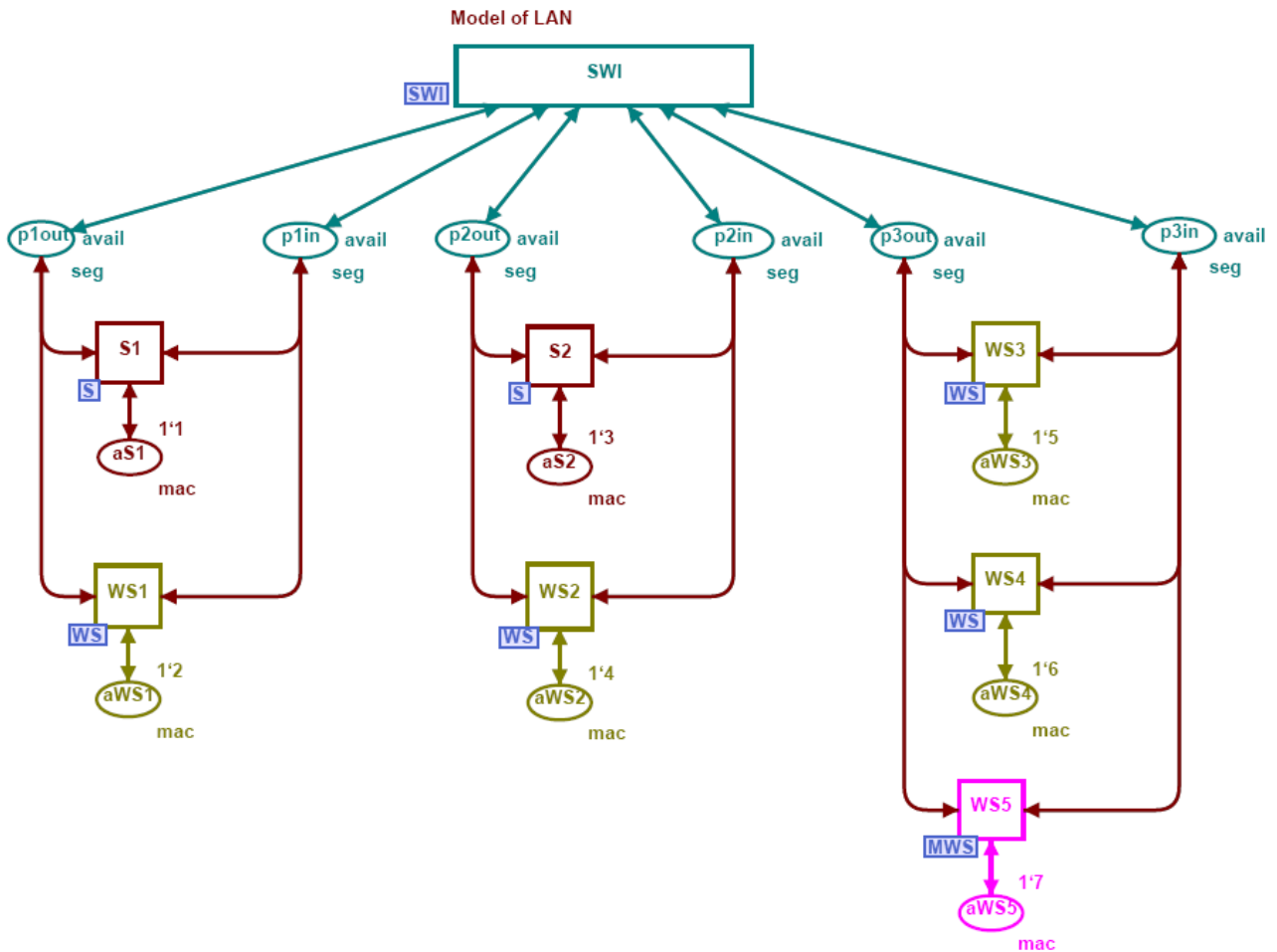


Рис. 2. Модель ЛВС

Рабочие станции **WS1-WS4** имеют одинаковый тип **WS**, в то время как рабочая станция **WS5** имеет специальный тип **MWS** и предназначена для измерения времени отклика. Серверы **S1** и **S2** имеют одинаковый тип **S**. Коммутатор **SWI** в единственном экземпляре представлен сетью типа **SWI**.

Каждый сервер и рабочая станция имеет собственный MAC-адрес, указываемый в позициях **aS\***, **aWS\*** соответственно. Коммутатор содержит отдельные позиции для входящих (**p\*in**) и исходящих (**p\*out**) фреймов для каждого порта. Таким образом, моделируется полнодуплексный режим работы. Двухнаправленные дуги используются для моделирования процедур проверки несущей. Одна из дуг проверяет состояние канала, в то время как другая осуществляет передачу фрейма.

```

color mac = INT timed;
color portnum = INT;
color nfrm = INT;
color sfrm = product nfrm * INT timed;
color frm = product mac * mac * nfrm timed;
color seg = union f:frm + avail timed;
color swi = product mac * portnum;
color swf = product mac * mac * nfrm * portnum timed;
color remsv = product mac * nfrm timed;
var src, dst, target: mac;
var port: portnum;
var nf, rnf: nfrm;
var t1, t2, s, q, r: INT;
color Delta = int with 1000..2000;
fun Delay() = Delta.ran();
color dex = int with 100..200;
fun Dexec() = dex.ran();
color dse = int with 10..20;
fun Dsend() = dse.ran();
color nse = int with 10..20;
fun Nsend() = nse.ran();
fun cT()=IntInf.toInt(!CPN*Time.model_time)

```

Рис. 3. Описания типов, переменных и функций

Все описания типов (**color**), переменных (**var**) и функций (**fun**), использованные в модели, представлены на Рис. 3. MAC-адрес Ethernet моделируется целым числом (**color mac**). Фрейм представлен тройкой **frm**, которая содержит адреса источника (**src**) и назначения (**dst**), а также специальное поле **nfrm** для нумерации фреймов в целях организации вычислений времени отклика. Мы абстрагируемся от остальных полей фрейма, предусмотренных стандартами Ethernet. Тип **seg** представляет однонаправленный канал и может быть либо доступным для передачи (**avail**), либо занятым передачей фрейма (**f.frm**). Выбор представлен с помощью стандартного типа объединения **union**. Заметим, что описатель **timed** использован для фишек, которые задействованы во временных операциях, таких как задержки либо временные штампы.

Маркировка позиций представлена в CPN Tools с помощью мультимножеств. Каждый из элементов принадлежит мультимножеству с определённой кратностью, другими словами – в нескольких экземплярах. Например, начальная маркировка позиции **aWS2** равна **1^4**. Это означает, что позиция **aWS2** содержит 1 фишку со значением 4. Объединение фишек представлено с помощью двойного знака плюс (**++**). Фишки временных типов имеют форму **x@t**, которая означает, что фишка **x** может быть задействована лишь после момента модельного времени **t**. Поэтому запись **@+d** используется для представления временной задержки на интервал **d**.

#### 4. Модель коммутатора

Модель коммутатора, изученная в [2], существенно модифицирована для отображения полнодуплексного режима работы и процедур CDMA. Отметим также модульный принцип построения модели. Каждый из портов представлен стандартным фрагментом, из которых можно легко собрать модель для произвольного заданного числа портов с помощью операции совмещения позиций. Такая конструкция модели позволяет, кроме того, избежать многократных пересечений дуг [2]. Модель коммутатора (**SWI**) изображена на Рис. 4. Таблица коммутации соответствует структурной схеме ЛВС (Рис. 1) и является статической.

### Model of Switch (SWI)

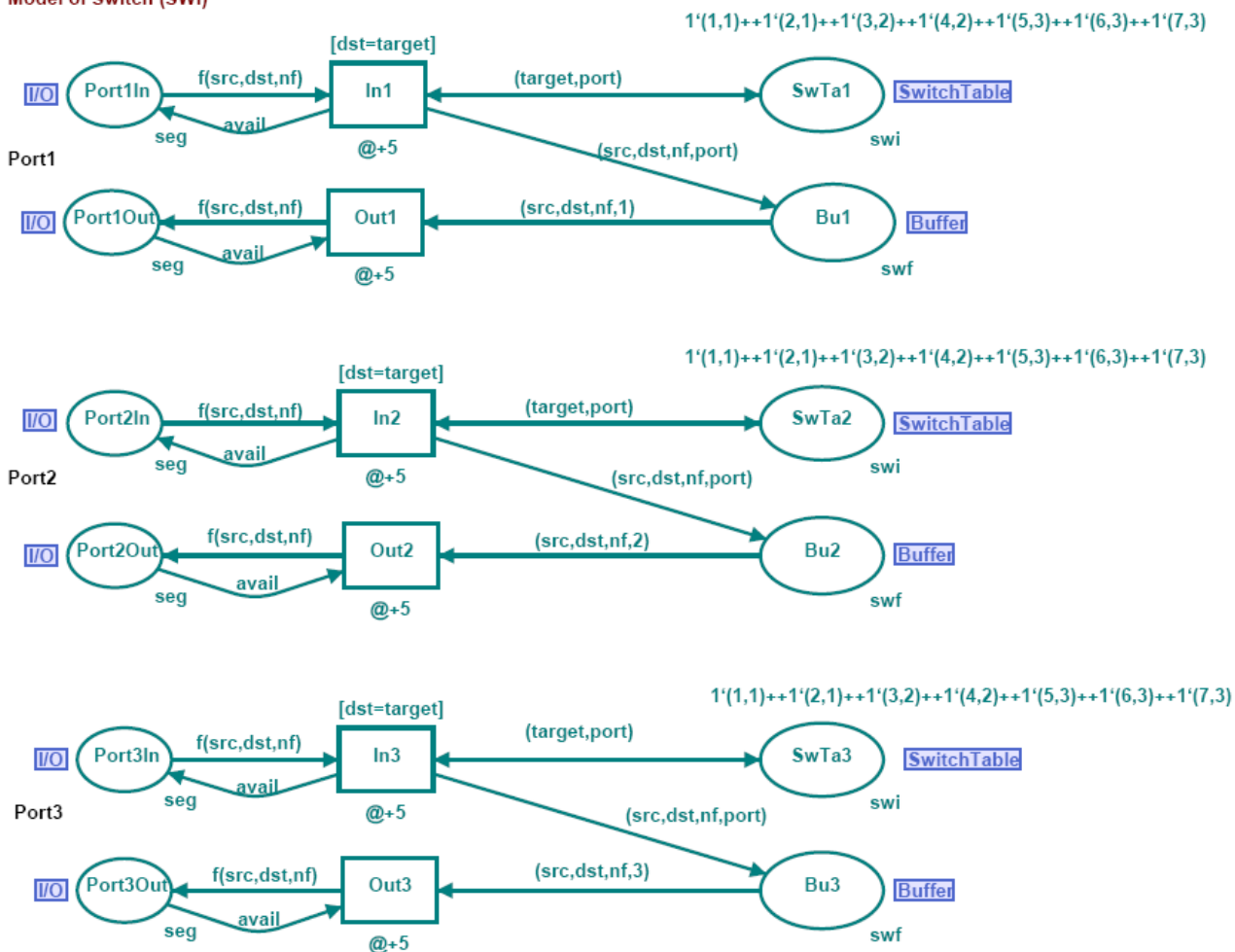


Рис. 4. Модель коммутатора (SWI)

Тип **swi** представляет записи таблицы коммутации, которая отображает каждый известный MAC-адрес (**mac**) на номер порта (**nport**). Тип **swf** описывает скомутированные фреймы, ждущие выделения выходного буфера порта. Поле **portnum** хранит номер порта назначения. Позиции **Port\*In** и **Port\*Out** представляют входной и выходной буфер порта соответственно. Позиция **SwitchTable** моделирует таблицу коммутации; каждая фишка в этой позиции представляет запись таблицы. Например, фишка  $1'(4,2)$  начальной маркировки означает, что компьютер с MAC-адресом 4 присоединён к порту 2. Позиция **Buffer** соответствует буферу скомутированных фреймов. Отметим ещё раз, что каждая их позиций **SwitchTable** и **Buffer** являются совмещённой. Так, совмещённая позиция **SwitchTable** представлена на схеме позициями **SwTa1**, **SwTa2**, **SwTa3**, а совмещённая позиция **Buffer** – позициями **Bu1**, **Bu2**, **Bu3**. Такой способ построения модели позволяет наглядно изображать модели с произвольным количеством портов.

Переходы **In\*** моделируют обработку входящих фреймов. Фрейм извлекается из входного буфера только в случае, если таблица коммутации содержит запись с адресом, который равен адресу назначения фрейма (**dst=target**); в процессе перемещения фрейма порт назначения (**port**) сохраняется в буфере. Переходы **Out\*** моделируют перемещение скомутированных фреймов в выходные буферы портов. Фиксированные временные задержки (**@+5**) присвоены операциям коммутации и записи фреймов в выходные буферы портов.

Остановимся более подробно на описаниях процедур CDMA доступа к ЛВС. Когда фрейм извлекается из входного буфера переходом **In\***, он замещается пометкой **avail**. Пометка **avail** показывает, что канал свободен и доступен для передачи. Перед тем, как переход **Out\*** посылает фрейм в порт, он ждёт освобождения канала, проверяя наличие фишки **avail**.

Заметим, что представленная модель абстрагируется от временных процессов трансляции фрейма в сегменте и, таким образом, не допускает коллизий. Указанные процессы детально изучены в [10].

Остановимся на принципах компоновки модели ЛВС. Позиции **Port\*In** и **Port\*Out** модели коммутатора являются контактными. Они помечены прямоугольниками I/O. Контактные позиции используют для построения иерархических сетей с помощью операции подстановки перехода. Например, переход **SWI** в модели ЛВС верхнего уровня (Рис. 2) замещается целой сетью **SWI**, представленной на Рис. 3. При этом позиции **Port\*In** и **Port\*Out** отображаются в позиции **p\*in** и **p\*out** соответственно.

## 5. Модели рабочей станции и сервера

Модели рабочей станции и сервера, изученные в [2], дополнены отдельными буферами входящих и исходящих фреймов, а также проверкой занятости канала. Рассмотрены периодически повторяющиеся запросы рабочих станций. В ответ на принятый запрос сервер посылает несколько пакетов в адрес запрашивающей рабочей станции. Количество посылаемых пакетов, периодичность и время выполнения запросов являются случайными равномерно распределёнными величинами. При необходимости могут быть изучены более сложные схемы взаимодействия в системах клиент-сервер, обусловленные спецификой решаемых задач.

Model of Workstation (WS)

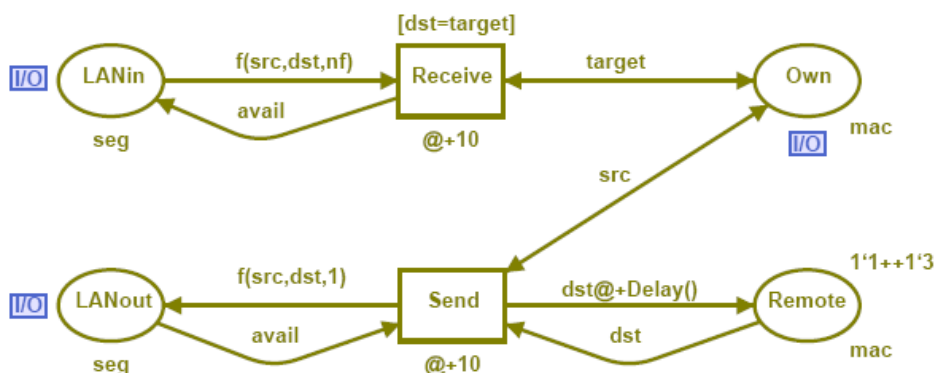


Рис. 5. Модель рабочей станции

Модель рабочей станции (WS) представлена на Рис. 5. Позиции **LANin** и **LANout** моделируют входные и выходные каналы сегмента, работающего в полнодуплексном режиме. Рабочая станция слушает сегмент с помощью перехода **Receive**, который получает фреймы с адресами назначения, равными собственному адресу рабочей станции ( $dst=target$ ), сохранённому в позиции **Own**. Обработка фреймов представлена их поглощением. Рабочая станция посылает периодические запросы к серверу с помощью перехода **Send**. Адреса серверов хранятся в позиции **Remote**. После посылки запроса использование адреса сервера блокируется на случайный интервал времени, заданный функцией **Delay()**. Посылка фрейма выполняется лишь в том случае, если сегмент ЛВС свободен, что реализовано проверкой позиции **LANout** на наличие фишки *avail*. Рабочая станция может взаимодействовать с несколькими серверами, сохраняя их адреса в позиции **Remote**.

Заметим, что третье поле фрейма с именем **nfrm** не используется обычной рабочей станцией. Рабочая станция лишь присваивает ему значение равное единице. Это поле использовано в модели специальной измерительной рабочей станции **MWS**. Копии описанной модели **WS** представляют рабочие станции **WS1-WS4**. Для уникальной

идентификации каждой рабочей станции использована контактная позиция **Own**. Эта позиция представлена в общей модели ЛВС (Рис. 2) позициями **aWS\*** и содержит MAC-адрес компьютера.

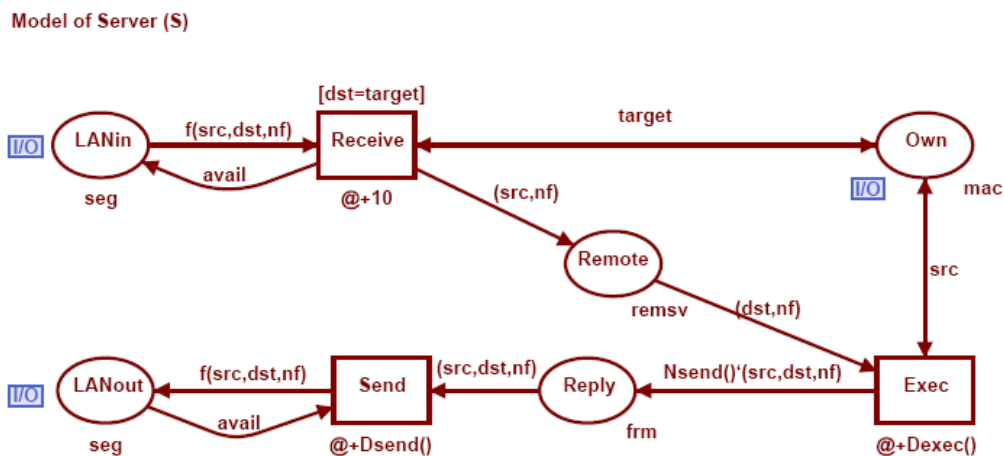


Рис. 6. Модель сервера

Модель сервера (S) изображена на Рис. 6. Прослушивание сегмента сети аналогично модели рабочей станции, отличие состоит в том, что адрес отправителя фрейма **src** и порядковый номер запроса **nf** сохраняются в позиции **Remote**. Переход **Exec** моделирует выполнение запроса рабочей станции сервером. В результате выполнения запроса сервер генерирует случайное число **Nsend()** фреймов ответа, которые сохраняются в позиции **Reply**. Затем эти фреймы по одному передаются в сеть переходом **Send**. Заметим, что номер запроса **nf**, сохранённый в позиции **Remote**, используется для идентификации ответа тем же самым номером, что и запрос.

## 6. Модель измерительной рабочей станции

Модель измерительной рабочей станции (MWS) изображена на Рис. 7. По существу, она представляет собой рассмотренную ранее модель рабочей станции **WS**, дополненную измерительными элементами, образующими фрагмент сети, выделенный малиновым цветом.



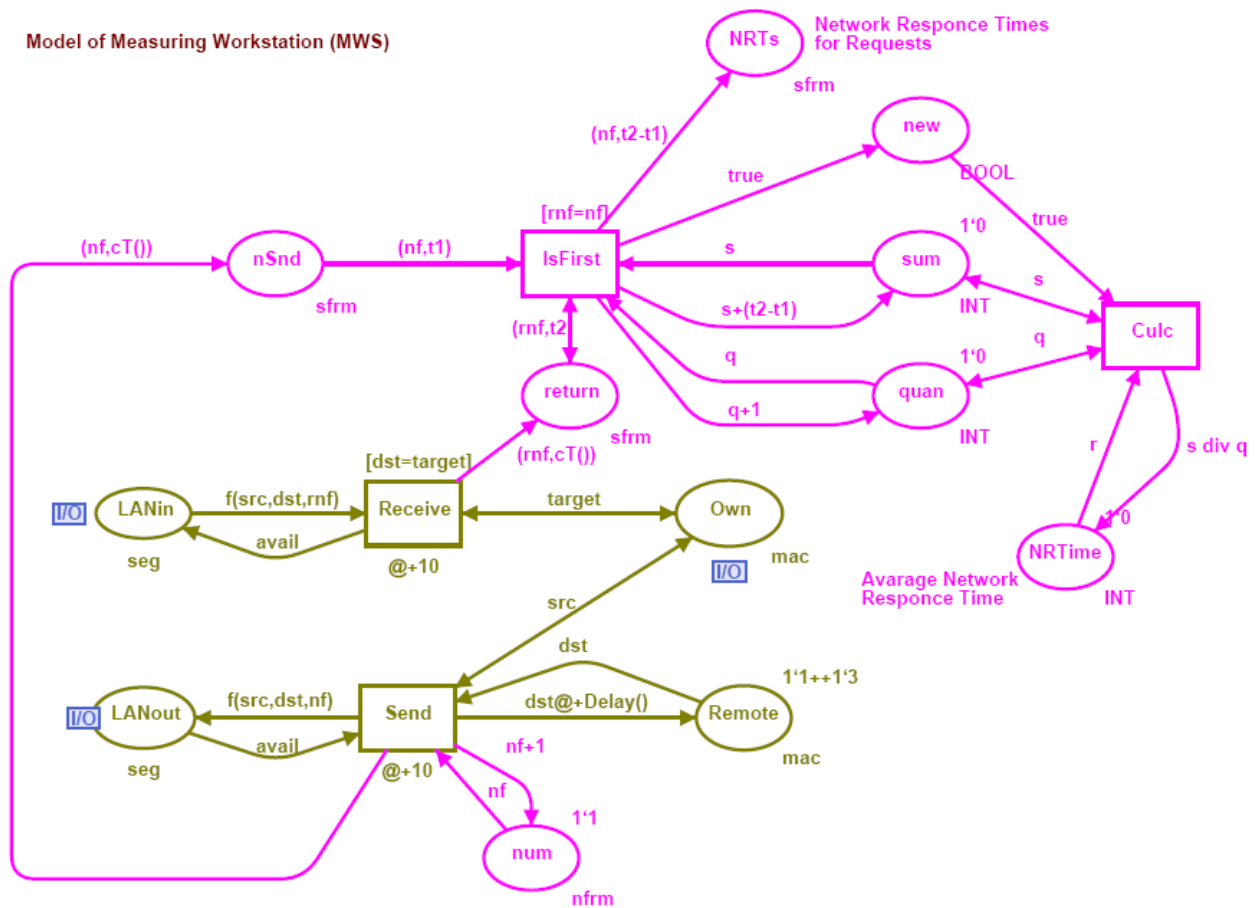


Рис. 7. Модель измерительной рабочей станции

Напомним, что расширенные сети Петри являются универсальной алгоритмической системой [5] и позволяют описывать не только исследуемый объект, но также дополнительные алгоритмы, предназначенные как для управления объектом [6], так и для организации измерений параметров функционирования. При этом алгоритм может быть сформирован из фрагментов сетей, представляющих логические, арифметические операции [7], альтернативные и параллельные процессы [8], характерные для современных языков программирования. Детальные описания таких алгоритмов, полностью собранные из фрагментов сетей могут быть довольно громоздкими. CPN Tools позволяет получить разумный компромисс между единством инструментальных средств и компактностью представления модели. Встроенный язык ML даёт возможность дополнить сеть Петри операторами языка программирования, ассоциированными как с дугами, так и с переходами сети. При этом необходимые величины могут сохраняться как в позициях сети Петри, так и в переменных языка ML.

Рассмотрим более детально измерительные элементы, представленные на Рис. 7. Каждый запрос рабочей станции нумеруется уникальным числом, содержащимся в позиции **num**. Время отправления запроса сохраняется в позиции **nSnd**. Функция **cT()** определяет текущее модельное время. Позиция **nSnd** запоминает пару: номер запроса (фрейма) **nf** и время отправления запроса в сеть.

Позиция **return** запоминает временные штампы всех вернувшихся фреймов. В качестве времени отклика сети рассмотрен интервал времени между посылкой запроса и получением первого из фреймов ответа. Это значение сохраняется в позиции **NRTs** для каждого обработанного запроса. Переход **IsFirst** распознаёт первый фрейм ответа. Описатель дуги, соединяющей переход **IsFirst** с позицией **NRTs**, вычисляет время отклика ( $t_2 - t_1$ ).

Оставшаяся часть измерительного фрагмента вычисляет среднее время отклика. Позиции **sum** и **quant** накапливают сумму времён отклика и количество принятых ответов

соответственно. Прибытие нового ответа распознаётся позицией **new** и инициирует перевычисление среднего времени отклика с помощью перехода **Culc**. Результат сохраняется в позиции **NRTTime**.

## 8. Технология измерений

Построенная модель отлажена и протестирована в пошаговом режиме имитации динамики сети Петри. Для этих целей фреймы, генерируемые рабочей станцией, трассировались через модель ЛВС по пути к серверу и обратно. Также наблюдалось поведение модели в процессе автоматической имитации с визуальным отображением динамики сети – в режиме так называемой игры фишек. Это позволяет оценить модель с помощью визуального наблюдения процессов на основной странице и страницах компонентов модели в процессе имитации.

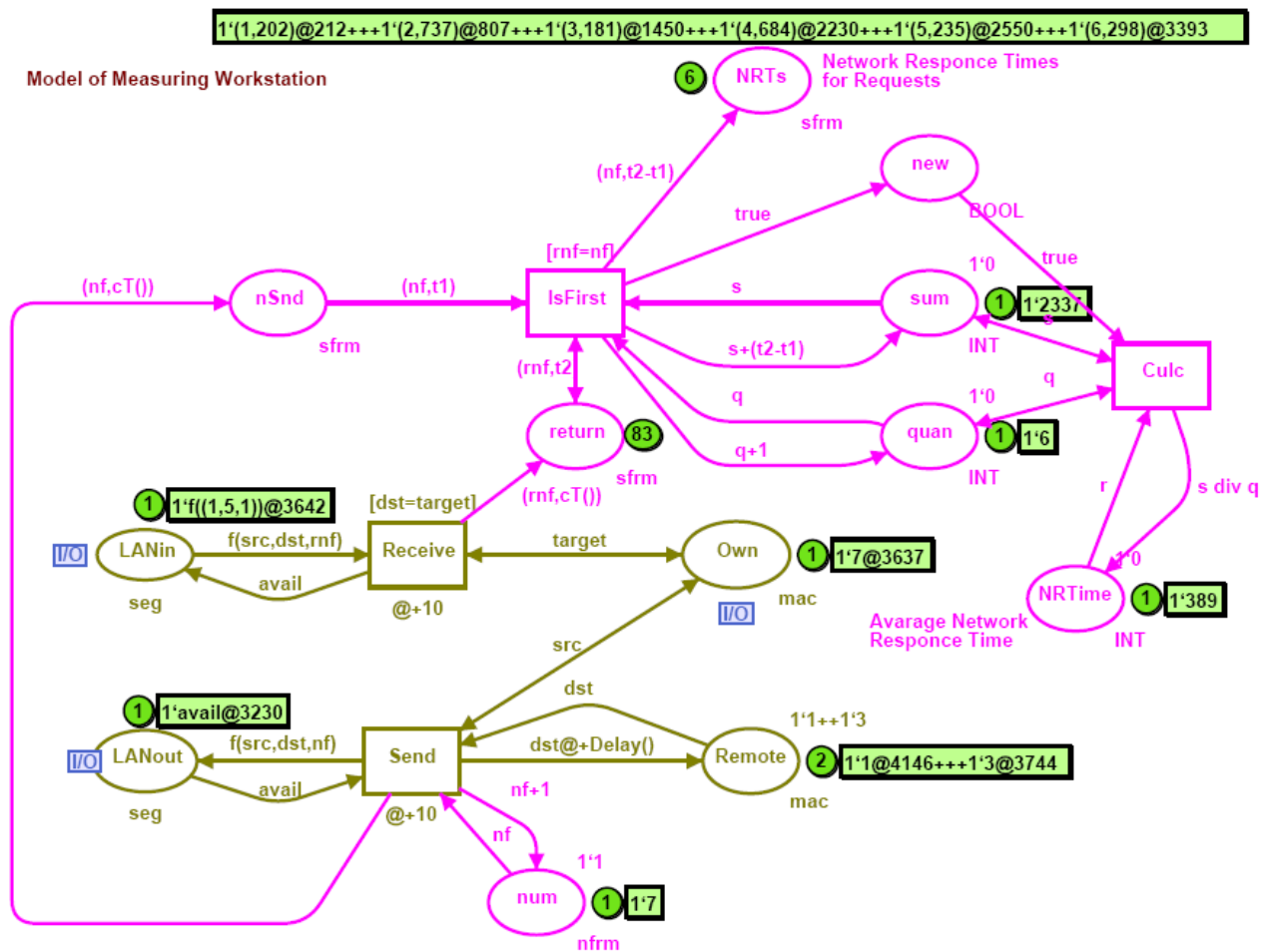


Рис. 8. Оценка времени отклика

Для более точной оценки времени отклика выбирались значительные интервалы модельного времени. При этом удобно использовать автоматический режим имитации без отображения промежуточных маркировок, целью которого является накопление статистической информации.

Образ экрана измерительной рабочей станции представлен на Рис. 8. Ярко зелёные пометки представляют текущую маркировку моделирующей системы. Позиция **LANin** содержит фрейм (1,5,1). Позиция **LANout** представляет доступное состояние канала **avail**. Номер следующего запроса в соответствии с маркировкой позиции **num** равен 7. Позиция **return** содержит 83 фрейма полученных ответов серверов. Позиция **NRTs** содержит времена отклика для каждого из 6 обработанных запросов. Например, время отклика сети для запроса

5 составляет 235 единиц модельного времени. Можно проверить, что среднее время отклика 389 в позиции **NRTIME** равно 2337/6 в соответствии с маркировками позиций **sum** и **quant**.

## 8. Параметры модели

Правильный выбор единицы измерения модельного времени также как и вычисление временных параметров элементов модели является ключевым вопросом для построения адекватной модели. Для этого требуется тщательный анализ характеристик реального оборудования и программного обеспечения конкретной ЛВС.

Схема, изображённая на Рис. 1, представляет фрагмент ЛВС диспетчерского центра железной дороги, оснащённого программным обеспечением ГИД Урал-ВНИИЖТ [11]. Ядро системы состоит из пары зеркальных серверов **S1** и **S2**. Рабочие станции **WS1-WS5** расположены непосредственно на рабочих местах железнодорожных диспетчеров.

Необходимо рассмотреть производительность конкретных коммутаторов ЛВС и сетевых адаптеров для вычисления временных задержек, моделируемых переходами **In\***, **Out\***, **Send**, **Receive**. Кроме того, необходимо изучить особенности взаимодействия в системе клиент-сервер программного обеспечения ГИД Урал для оценки таких параметров как задержка между запросами **Delta** и время выполнения запроса **dex**. Так как единицей информации, передаваемой в сети, является Ethernet-фрейм, следует выразить длины сообщений в количестве передаваемых фреймов. Для этих целей выберем максимальную длину фрейма Ethernet, равную 1,5 Кб.

Типы используемого оборудования ЛВС указаны в Таблице 1.

Таблица 1. Типы оборудования

Устройство	Тип
Сетевой адаптер	Intel EtherExpress 10/100
Коммутатор ЛВС	Intel SS101TX8EU
Сервер	HP Brio BA600
Рабочая станция	HP Brio BA200

В Таблице 2 представлены параметры описанной ранее модели. Наименьшее значение времени соответствует продолжительности операции чтения/записи фрейма для коммутатора ЛВС. Но в целях дальнейшего использования более производительного оборудования выбрана единица модельного времени (ЕМВ) равная 100 нс.

Таблица 2. Параметры модели

Параметр	Переменная/ Элемент	Реальное значение	Модельное значение
Время чтения фрейма коммутатором ЛВС	In*	500 нс	5
Время записи фрейма коммутатором ЛВС	Out*	500 нс	5
Время чтения фрейма сетевым адаптером	Receive	1 мс	10
Время записи фрейма сетевым адаптером	Send	1 мс	10
Время обработки запроса сервером	Dex	10-20 мс	100-200
Интервал между запросами клиента	Delta	100-200 мс	1000-2000
Длина запроса		1,2 Кб	1
Длина ответа	Nse	15-30 Кб	10-20

Таким образом, полученное среднее время отклика сети равно 389 ЕМВ или около 39 мс. Эта величина удовлетворяет требованиям оперативного управления движением поездов.

## 9. Заключение

Таким образом, в настоящей работе представлена методология измерения характеристик моделей Петри телекоммуникационных систем с помощью специальных измерительных фрагментов сетей Петри. Особенности применения измерительных фрагментов изучены на примере определения времени отклика в модели коммутируемой локальной сети.

Представленная методология построения моделей в форме иерархических раскрашенных временных сетей Петри, а также измерения параметров модели может быть успешно применена при исследовании и проектировании различного телекоммуникационного оборудования.

## Литература

1. Jensen K. Colored Petri Nets – Basic Concepts, Analysis Methods and Practical Use. Springer-Verlag, Vol. 1-3, 1997.
2. Зайцев Д.А., Шмельёва Т.Р. Моделирование коммутируемой локальной сети раскрашенными сетями Петри // Зв'язок, № 2(46), 2004, с. 56-60.
3. Zaitsev D.A. Switched LAN simulation by colored Petri nets // Mathematics and Computers in Simulation, vol. 65, no. 3, 2004, p. 245-249.
4. Beaudouin-Lafon M., Mackay W.E., Jensen M. et al. CPN Tools: A Tool for Editing and Simulating Coloured Petri Nets. LNCS 2031: Tools and Algorithms for the Construction and Analysis of Systems, 2001, 574-580.
5. Котов В.Е. Сети Петри.- М. Наука, 1984.- 160 с.
6. Слепцов А.И., Юрасов А.А. Автоматизация проектирования управляющих систем гибких автоматизированных производств / Под ред. Б.Н.Малиновского.- К. Техніка, 1986.- 160 с.
7. Слепцов А.И. Уравнения состояний и эквивалентные преобразования нагруженных сетей Петри (алгебраический подход) // Формальные модели параллельных вычислений: Докл. и сообщ. Всесоюзн. конф. – Новосибирск, 1988, с. 151-158.
8. Зайцев Д.А., Слепцов А.И. Уравнение состояний и эквивалентные преобразования временных сетей Петри // Кибернетика и системный анализ.- 1997, № 5, с. 59-76.
9. Rahul V. LAN Switching. ОНЮ, 2002.
10. Зайцев Д.А. Верификация протоколов Ethernet // Научные труды Одесской национальной академии связи им. А.С. Попова.- №1.- 2004.
11. Зяби́ров Х.Ш., Кузнецов Г.А., Шевелев Ф.А., Слободенюк Н.Ф., Крашенинников С.В., Крайсвитний В.П., Ведищев А.Н. Автоматизированная система оперативного управления эксплуатационной работой ГИД "Урал-ВНИИЖТ" // Железнодорожный транспорт, №2, 2003, с. 36-45.

*Опубликовано в Зв'язок №2(54), 2005, с. 65-71.*