

**С.А. Даденков, Е.Л. Кон**

Пермский национальный исследовательский  
политехнический университет

**ИССЛЕДОВАНИЕ ПРОИЗВОДИТЕЛЬНОСТИ  
АЛГОРИТМА ДОСТУПА К СРЕДЕ  
PREDICTIVE P-PERSISTENT CSMA-ПРОТОКОЛА**

*Представлено подробное описание алгоритма доступа к среде передачи прогнозирующего псевдопостоянного CSMA-протокола. Выполнен анализ алгоритма и получены аналитические выражения для расчёта основных характеристик производительности LonWorks-сети.*

Эволюция развития сетей связи показывает, что на всём протяжении существования сетей актуальной задачей являются разработка и использование методов множественного доступа, основа которых – разделяемая среда передачи данных. Основными способами организации совместного доступа устройств к разделяемым линиям связи являются: централизованный подход на основе управляющего устройства – арбитра и децентрализованный подход. Так, на протяжении долгого времени разделяемая среда с децентрализованным подходом была основным механизмом использования каналов связи для локальных сетей на основе технологий *Ethernet*, *ArcNet*, *Token Ring*, *FDDI* [1]. Сегодня методы множественного доступа широко используются в беспроводных локальных сетях, а также промышленных fieldbus-системах. Области применения методов обусловлены следующими факторами [1]:

- простота и экономичность аппаратных решений;
- низкая и слабопредсказуемая производительность при больших нагрузках;
- плохая масштабируемость.

Очевидно, что основным фактором, ограничивающим применение методов совместного использования канала устройствами сети,

является производительность. Поэтому большое внимание инженеров и исследователей уделено разработке алгоритмов множественного доступа, обеспечивающих оптимальный уровень производительности в зависимости от загрузки сети. Одним из таких алгоритмов с децентрализованным подходом является прогнозирующий псевдопостоянный алгоритм множественного доступа с контролем несущей и предотвращением коллизий predictive p-persistent CSMA. Данный алгоритм нашёл применение в fieldbus-системах *LonWorks*, которые распространены и используются в области автоматизации зданий и технологических процессов промышленных предприятий. *LonWorks* являются системами реального времени. Поэтому актуальной задачей является исследование производительности алгоритма множественного доступа pp-CSMA.

Анализ публикаций, посвящённых данной тематике, позволил выявить ряд работ авторов Marek Miśkiewicz [4], Moshe Kam [5]. Указанные работы посвящены решению задачи оценки производительности сегментов Lon-сети, в основу которой положен уровень доступа к среде передачи. Это обусловлено тем, что уровень доступа определяет результирующую производительность сети в случае низких скоростей передачи и длинных пакетов, но и требует учёта в случае высоких скоростей передачи и коротких пакетов, когда производительность сети в значительной мере определяется быстродействием *Neuron Chip* [2, 3]. В отличие от указанных публикаций данная работа направлена на выполнение подробного анализа принципов функционирования алгоритма pp-CSMA протокола, а также на разработку аналитических выражений для оценки параметров производительности, учитывающих специфику механизма приоритизации в сети.

### **1. Протокол predictive p-persistent CSMA**

Алгоритм управления доступом к каналу Lon-сети определён псевдопостоянным (p-persistent) CSMA-протоколом с предотвращением коллизий. Узел соперничает за совместно используемый канал согласно p-CSMA протоколу, успешно начинает передачу сообщения с вероятностью  $p$ , если канал неактивен, и задерживает передачу с вероятностью  $(1-p)$  [3]. Оптимальность использования канала в p-CSMA сильно зависит от значения  $p$ , которое представляет уровень настойчивости протокола. В частности, большие значения  $p$  вызывают чрезмерные коллизии в сети, в то время как маленькие значения вероятности

передачи ухудшают использование пропускной способности, приводя к неактивности канала. Таким образом, необходим компромисс между большими и маленькими значениями  $p$ , чтобы обеспечить использование пропускной способности на удовлетворительном уровне. Однако постоянный уровень настойчивости максимизирует пропускную способность только для предварительно выбранного числа узлов, конкурирующих за канал, что значительно ограничивает полезность чистого р-CSMA на практике. В случае, когда количество узлов, готовых к передаче, неизвестно или изменяется во времени, значение  $p$  не может быть установлено оптимально, и, следовательно, производительность р-CSMA может быть значительно уменьшена. Поэтому для решения задач выбора оптимального уровня настойчивости в Lon-сети используется прогнозирующий (predictive) р-CSMA-протокол с предотвращением коллизий.

Прогнозирующий р-persistent CSMA-протокол является адаптивной версией чистого р-CSMA и разработан специально для локальных операционных сетей (Lon), объединяющих интеллектуальные датчики и исполнительные механизмы [2]. Вероятность успешной передачи  $p$  становится переменной величиной и динамически корректируется согласно ожидаемой нагрузке, уменьшается в случае коллизий и увеличивается после каждой успешной передачи [3].

### 1.1. Алгоритм доступа к среде

Передача данных в Lon-сети выполняется пакетными циклами (packet cycle). Циклы принято разделять на свободные, когда в канале нет активности после передачи, и занятые, когда выполняется передача. Каждый занятый пакетный цикл представляет собой время передачи пакета с сообщением и время задержки, необходимое для того, чтобы узлы в сети смогли начать передачу. На рис. 1 изображен занятый пакетный цикл [3].

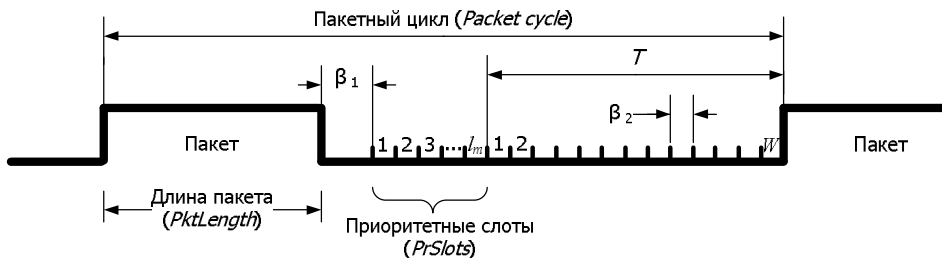


Рис. 1. Пакетный цикл

Алгоритм доступа к среде CSMA включает следующие этапы (см. рис. 1) [3]:

1. Проверка активности канала. Узел (интеллектуальный датчик или исполнительный механизм) для передачи сообщения в сеть следит за состоянием канала. Если канал занят, узел продолжает прослушивание канала. Когда узел не обнаруживает активности в канале, выполняется следующий этап алгоритма.

2. Межпакетный интервал. Узел выжидает минимальный временной межпакетный интервал  $\beta_1$  для определения того, что канал свободен.

3. Приоритетный интервал. Узел выжидает фиксированное количество приоритетных временных (тайм) слотов  $l = [0..127]$  длительностью  $\beta_2$  каждый. Приоритеты в сети работают только после занятого пакетного цикла, если передаче предшествовал пустой пакетный цикл, то доступ к каналу происходит без приоритетного интервала [3].

4. Интервал избегания коллизий. Узел формирует случайную задержку  $T$ , которая определяется случайным количеством тайм-слотов продолжительностью  $\beta_2$ . Если канал после случайной задержки  $T$  свободен, то узел начинает передачу пакета. В противном случае узел ждёт освобождение канала и снова пытается конкурировать за канал. В случае, если несколько узлов выбрали одинаковое количество временных слотов  $\beta_2$  и начали передачу одновременно, в сети возникает коллизия. Все пакеты, участвующие в столкновении, становятся повреждёнными.

Случайное время задержки  $T$  определяется как псевдослучайное число состязательных слотов, взятых согласно равномерному распределению из интервала  $(0..(BL \cdot W_{\text{base}} - 1))$ , где  $W_{\text{base}}(W)$  – ширина основного состязательного окна, равная 16 слотам для EIA-709.  $BL$  – величина отставания канала (backlog), которая характеризует загруженность канала, изменяется от 1 до 63. Для свободного канала  $BL$  равняется единице и состязательное окно  $W$  содержит  $W_{\text{base}}$  временных слотов. Когда нагрузка на канал увеличивается, количество слотов растёт, так как растёт отставание канала  $BL$ , и ширина состязательного окна может достигать максимального значения  $W = BL \cdot W_{\text{base}} = 1008$  слотов. Таким образом, уровень настойчивости р-СМА равняется  $1/(W_{\text{base}} \cdot BL)$  и является переменным, имеет нижнюю границу вероятности успешной передачи одним узлом ( $1/16 = 0,0625$ ) и верхнюю границу ( $1/1008 = 0,0009$ ).

## 1.2. Алгоритм подсчёта отставаний (backlog) BL

Значение счётчика отставаний BL изменяется от одного пакетного цикла к другому. Изменение счётчика BL производится на величину, значение которой передаётся в закодированном 6-битном поле  $\Delta_{BL}$  пакета и представляет собой количество сообщений (подтверждений), которые должны быть сформированы получателем и отправлены в ответ на данное сообщение. В литературе значение BL называют незавершённой работой в сети. Для одноадресных (unicast) сообщений  $\Delta_{BL} = 1$ , а для многоадресных (multicast)  $1 < \Delta_{BL} < 63$ . Таким образом, величина счётчика отставания характеризует загруженность канала и определяет число пакетов, ожидающих соревнования во время следующего пакетного цикла [3].

Алгоритм изменения отставания канала зависит от вида службы, используемой для передачи сообщения по сети. Протокол LonTalk определяет четыре службы доставки сообщений и алгоритмы изменения отставаний [2, 3]:

- служба с подтверждением (Acknowledged). Приращение счётчика отставаний осуществляется на величину количества подтверждений, которые должны быть сформированы получателями;
- служба без подтверждения (Unacknowledged). Подразделяется на два вида – с повторениями и без повторений (repeated). Приращение счётчика отставаний производится на величину числа повторений для первичного сообщения и на нулевое значение для повторяемых сообщений в рамках одной транзакции;
- служба запрос/ответ (Message-Reminder). Приращение счётчика отставаний осуществляется на величину числа сообщений, которые ожидаются к получению.

Каждый узел в сети вычисляет отставание автономно, специальным счётчиком, реализованным в прошивке *LonWorks*-узла. Изменение отставания канала происходит согласно следующим принципам [3]:

- увеличение отставания BL происходит в следующих случаях:
  - передача или приём пакета с ненулевым полем  $\Delta_{BL}$  ( $+\Delta_{BL}$ );
  - обнаружение коллизии (+1);
- уменьшение отставания BL происходит в следующих случаях:
  - в конце каждого успешного пакетного цикла (-1);
  - при успешно переданном пакете подтверждения ( $\Delta_{BL}=0$ ) (-1);
  - пакетный цикл завершается бездействием канала (-1).

## 2. Средняя задержка доступа узла к каналу

Средняя задержка доступа узла к каналу (mean access delay) определяется как среднее время с момента, в который узел пытается начать отправку пакета, и до начала его успешной передачи [3].

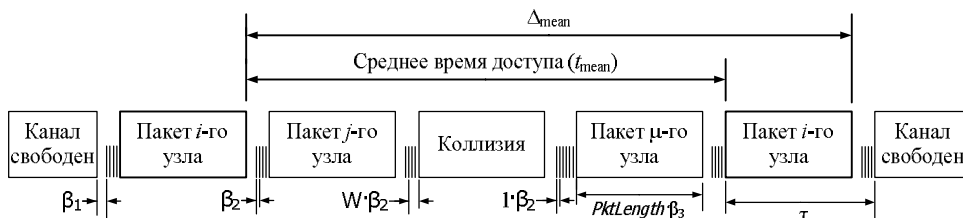


Рис. 2. Время доступа узла к каналу

Задержка доступа узла к каналу  $t_{\text{mean}}$  состоит из следующих компонентов (рис. 2):

- задержка передачи пакетов другими узлами сети;
- задержка передачи, в случае коллизии;
- задержка нахождения пакета в очереди узла до отправки.

В работе для оценки времени доступа узлов к каналу учитываются только две первые компоненты. Согласно LonTalk/EIA-709.1 коллизия обнаруживается в конце пакетной передачи, что позволяет вычислять продолжительность пакетного цикла при неудачной передаче аналогично случаю с успешной передачей.

Передача пакетов организована в пакетных циклах продолжительностью  $\tau$ , которая включает:

- фиксированный межпакетный интервал  $\beta_1$ ;
- фиксированный приоритетный интервал  $l \cdot \beta_2$ ;
- разыгрываемый случайный интервал времени  $W \cdot \beta_2$ ;
- интервал времени передачи сообщения  $PktLength \cdot \beta_3$ .

### 2.1. Оценка средней задержки доступа

Среднее время задержки по передаче пакета сообщения может быть оценено как средняя продолжительность временного интервала  $\Delta_{\text{mean}}$  между двумя последовательными успешными попытками доступа к каналу, предпринятыми одним узлом (см. рис. 2). Временной интервал  $\Delta_{\text{mean}}$  находится по соотношению

$$\Delta_{\text{mean}} = \xi \cdot \tau. \quad (1)$$

где  $\xi$  обозначает число попыток доступа к каналу, предпринятых определённым узлом, прежде чем выполнить успешную передачу пакета,  $\tau$  – средняя длина пакетного цикла.

Время задержки узла на доступ к каналу  $t_{\text{mean}}$  означает задержку до начала успешной передачи и при этом не учитывает времени, требуемого на передачу пакета, после того, как узел выигрывает состязание:

$$t_{\text{mean}} = \Delta_{\text{mean}} - PktLength \cdot \beta_3. \quad (2)$$

где  $PktLength$  – длина пакета в битах,  $\beta_3$  – время передачи одного бита сообщения.

Для простоты в работе предполагается, что длина пакетов (сообщение/подтверждение), передаваемых через канал, является постоянной. Это предположение верно в случае, когда поле данных в сообщении короткое по сравнению со служебной информацией протокола. Это так, когда узлы обмениваются короткими явными сообщениями или обновлением сетевых переменных [3].

Из (1) и (2) следует, что для оценки средней задержки доступа  $t_{\text{mean}}$  должны быть определены число попыток до успешной передачи  $\xi$  и средняя длина пакетного цикла  $\tau$ .

## 2.2. Среднее количество попыток до успешной передачи

Рассмотрим сеть в определённый момент времени, в которой присутствует  $n$  узлов-соперников за доступ к каналу. Вычислим среднее число попыток передачи, прежде чем определённый узел выигрывает соревнование за канал.

Различие между узлами сети по количеству попыток за доступ к каналу заключается в разнице вероятностей доступа, которые зависят от уровня приоритетности сообщений, передаваемых узлами. В работе сделано предположение, что все сообщения, передаваемые определённым узлом, имеют один уровень приоритета, поэтому далее в работе используется понятие «уровень приоритетности узла». Приоритетные узлы имеют меньшее число приоритетных временных слотов  $\beta_2$  для ожидания и поэтому характеризуются большей вероятностью выигрыша в соревновании за канал. Рассмотрим два различных сегмента сети – без приоритетов и с приоритетами. Слововое представление конкурирующих узлов в рассматриваемых сегментах представлено на рис. 3.

В сегменте сети с приоритетами находится  $n_l$  узлов каждого  $l$ -приоритета, где  $l = 0 \dots l_{\text{max}}$ , причём большее значение  $l$  соответствует

меньшему уровню приоритета. Значение приоритета  $l$  характеризует дополнительное фиксированное количество временных слотов, которые узел должен выждать, прежде чем получить доступ к каналу (см. рис. 3). Общее количество узлов в сегменте сети  $n = \sum n_l$ , в диапазоне  $l = 0 \dots l_{\max}$ .

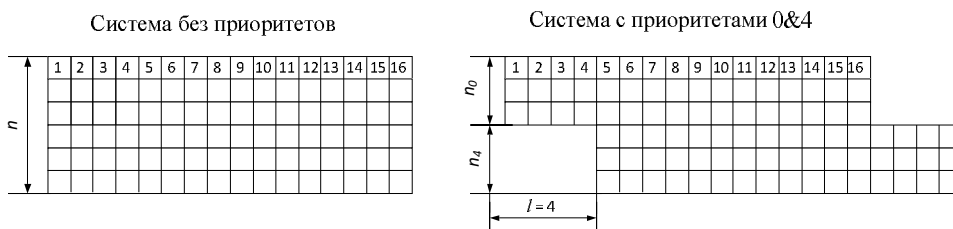


Рис. 3. Конкурирующие узлы в системе с приоритетами и без них

Определим среднее количество попыток на доступ к каналу до успешной передачи для узлов в рассматриваемых сегментах. Сеть без приоритетов является частным случаем с  $l = 0$  сети с приоритетами. Обозначим через  $x$  число испытаний для пакета, прежде чем будет выполнена его успешная передача. Обозначим  $p_{\text{succ}(l)}^l$  вероятность того, что узел с уровнем приоритета  $l$  успешно выполняет передачу при каждом испытании. Поскольку каждый узел в сети может выиграть состязание, вероятность успешной передачи любого пакета в канале со многими  $n$  соперниками

$$p_{\text{succ}} = \sum_{l=0}^{l_{\max}} n_l \cdot p_{\text{succ}(l)}^l \quad (3)$$

В сети без приоритетов все узлы характеризуются минимальным числом дополнительных слотов  $l = 0$ , и поэтому выражение (3) принимает частный вид  $p_{\text{succ}} = n \cdot p_{\text{succ}(0)}^0$ . Далее в работе все полученные выражения представляются в общем виде для системы с приоритетами.

Вероятность коллизии в сети вычисляется как

$$p_{\text{coll}} = 1 - p_{\text{succ}} \quad (4)$$

Вероятность  $p^l(x)$  выигрыша соревнования узлом с приоритетом  $l$  на попытке с номером  $x$  определяется как произведение [4]:

– вероятности того, что узел с приоритетом  $l$  выигрывает соревнование за канал с определённой попытки;

– вероятности того, что до попытки с номером  $x$  узел не выигрывал.



Поэтому

$$p^l(x) = (1 - p_{\text{succ}(1)}^l)^{x-1} \cdot p_{\text{succ}(1)}^l. \quad (5)$$

Среднее число попыток передачи узлом с  $l$ -приоритетом  $\xi^l$  определяется следующим выражением:

$$\xi^l = \sum_{x=1}^{\infty} x \cdot p^l(x) = \sum_{x=1}^{\infty} x \cdot (1 - p_{\text{succ}(1)}^l)^{x-1} \cdot p_{\text{succ}(1)}^l. \quad (6)$$

Умножая обе стороны выражения (6) на величину  $(1 - p_{\text{succ}(1)}^l)$ , получаем

$$(1 - p_{\text{succ}(1)}^l) \cdot \xi^l = \sum_{x=1}^{\infty} x \cdot (1 - p_{\text{succ}(1)}^l)^x \cdot p_{\text{succ}(1)}^l. \quad (7)$$

Разность выражений (6) и (7) позволяет записать соотношение для определения  $\xi^l(x)$ :

$$\xi^l = [1 + (1 - p_{\text{succ}(1)}^l) + (1 - p_{\text{succ}(1)}^l)^2 + \dots]. \quad (8)$$

Правая сторона выражения (8) включает сумму бесконечного геометрического ряда, поэтому приближённо выражение может быть записано как

$$\xi^l = \frac{1}{p_{\text{succ}(1)}^l}. \quad (9)$$

Среднее количество попыток до успешной передачи узлами в целом по сети может быть определено следующим образом:

$$\xi = \frac{1}{n} \cdot \sum_{l=0}^{l_{\max}} \frac{n_l}{p_{\text{succ}(1)}^l}. \quad (10)$$

Полученную формулу (10) можно использовать и для оценки среднего числа попыток до неудачной из-за коллизий передачи. Формула (10) определяет среднюю задержку доступа как среднее число пройденных испытаний, прежде чем узел выиграет соревнование за доступ к каналу в сети с  $n$  соперниками. Неизвестными величинами в выражении (10) являются вероятности выигрыша узлом с  $l$ -приоритетом соревнования за канал с одной попытки. Поэтому выполним более детальный анализ систем с приоритетами и определим выражения для определения вероятности успешной/неудачной передачи.

Расширим определение вероятности  $p_{\text{succ}(1)}^l$  и обозначим её через  $p_{\text{succ}(1)}^{W,k,l}(n)$  – вероятность, с которой узел с уровнем  $l$  приоритета выигрывает соревнование за канал при ширине соревновательного окна, равной  $W \cdot k$ , в сети с количеством узлов-соперников  $n$ .

В каждом пакетном цикле конкурирующие узлы выбирают случайное количество временных слотов, которое складывается из фиксированного числа приоритетных слотов  $[0..127]$  и случайного числа слотов для избегания коллизий из диапазона  $[0..W \cdot k]$ . Узел, который выбирает меньшее количество слотов, выигрывает соревнование за канал и начинает передачу.

С учётом вышеизложенного вероятность  $p_{\text{succ}(1)}^{W,k,l}(n)$  определяется как сумма вероятностей выбора узлом каждого слота  $s$  из соревновательного окна:

$$p_{\text{succ}(1)}^{W,k,l}(n) = \sum_{s=1}^{W \cdot k} p_{\text{succ}(1)}^{W,k,l,s}(n). \quad (11)$$

Вероятность выбора узлом слота с номером  $s$  при условии, что узел выигрывает на данном слоте соревнование за канал  $p_{\text{succ}(1)}^{W,k,l,s}(n)$ , вычисляется как произведение:

- вероятности того, что узел победитель с приоритетом  $l$  выбирает определённый слот  $s$  из диапазона  $W \cdot k$ ;
- вероятности того, что остальные узлы с приоритетом  $l$  выбирают один из  $(W \cdot k - s)$  более поздних слотов;
- вероятности того, что все остальные узлы с приоритетами, отличными от  $l$ , выбирают один из более поздних слотов.

Таким образом, выражение по определению вероятности можно записать как

$$p_{\text{succ}(1)}^{W,k,l,s}(n) = I(s > l) \cdot \frac{1}{W \cdot k} \cdot \left( \frac{W \cdot k - s}{W \cdot k} \right)^{n_l - 1} \cdot \prod_{r=0 \& \neq l}^{l_{\max}} D(s > r). \quad (12)$$

В выражении (12)  $I$  – индикаторная функция вероятности выбора узлом с приоритетом  $l$  слота с номером  $s$ :

$$I(\cdot) = \begin{cases} 1, & \text{если выполнено условие } (\cdot), \\ 0, & \text{если условие } (\cdot) \text{ не выполнено.} \end{cases} \quad (13)$$

Функция  $I(s > l)$  позволяет учитывать невозможность выбора узлом с приоритетом  $l$  слота с номером  $s \leq l$ .

Функция  $D$  в выражении (12) определяет вероятность выбора узлами с приоритетами  $r$ , отличными от приоритета  $l$  выигравшего узла, более позднего слота с номером  $s$ :

$$D(\cdot) = \begin{cases} \left( \frac{W \cdot k - s + r}{W \cdot k} \right)^{n_r}, & \text{если выполнено условие } (\cdot), \\ 1, & \text{если условие } (\cdot) \text{ не выполнено.} \end{cases} \quad (14)$$

Стоит заметить, что нумерация слотов производится относительно узлов с максимальным приоритетом, то есть для узлов с приоритетом, отличным от максимального, номер слота складывается из фиксированного числа приоритетных слотов и числа собственных случайных слотов.

Таким образом, определены все необходимые величины для расчёта вероятностей успешной/неудачной передачи в сети по выражениям (3–4). Выражение для определения вероятности успешной передачи принимает вид:

$$p_{\text{succ}}(n) = \sum_{l=0}^{l_{\text{max}}} \left[ n_l \sum_{s=1}^{W \cdot k} \left[ I(s > l) \frac{1}{W \cdot k} \left( \frac{W \cdot k - s}{W \cdot k} \right)^{n_l - 1} \prod_{r=0 \& \neq l}^{l_{\text{max}}} D(s > r) \right] \right]. \quad (15)$$

С учётом представленных выше выражений формула (9) для определения среднего числа пакетных циклов до успешной передачи узлом с приоритетом  $l$  принимает вид:

$$\xi^l = \left[ \sum_{s=1}^{W \cdot k} \left[ I(s > l) \frac{1}{W \cdot k} \left( \frac{W \cdot k - s}{W \cdot k} \right)^{n_l - 1} \prod_{r=0 \& \neq l}^{l_{\text{max}}} D(s > r) \right] \right]^{-1}. \quad (16)$$

### 2.3. Средняя длина пакетного цикла

Доступ к общему каналу организован в пакетных циклах. Каждый пакетный цикл – попытка предпринятой передачи узлами, у которых есть сообщение для отправки. Пакетный цикл начинается с межпакетного разрыва, фиксированного числа приоритетных слотов и случайного числа состязательных слотов. Результатом каждой попытки является либо успешная передача, либо коллизия.

Средняя длина пакетного цикла  $\tau$  в сети с  $n$  узлами-соперниками определяется как сумма длин успешных и неудачных пакетных циклов [4]:

$$\tau = p_{\text{coll}}(n) \tau_{\text{coll}}(n) + p_{\text{succ}}(n) \tau_{\text{succ}}(n), \quad (17)$$

где  $\tau_{\text{succ}}(n)$ ,  $\tau_{\text{coll}}(n)$  – средние длины успешных/неудачных пакетных циклов.

Средние длины пакетных циклов  $\tau_{\text{succ}}(n)$ ,  $\tau_{\text{coll}}(n)$  могут быть определены по формулам:

$$\tau_{\text{succ}}(n) = \beta_1 + [d_{\text{succ}}(n) - 1] \cdot \beta_2 + PktLength \beta_3, \quad (18)$$

$$\tau_{\text{coll}}(n) = \beta_1 + [d_{\text{coll}}(n) - 1] \cdot \beta_2 + PktLength \beta_3, \quad (19)$$

где  $d_{\text{succ}}(n)$  – среднее число слотов, при котором узел выигрывает конкуренцию и начинает передачу,  $d_{\text{coll}}(n)$  – среднее количество слотов, при котором происходит коллизия.

Определим среднее количество слотов в случае успешной/неудачной передачи. Отметим, что при расчётах среднего числа слотов следует учитывать также приоритетные слоты (см. рис. 3). Определим среднее количество слотов  $d_{\text{succ}}^{W,k,l}(n)$ , выбранное выигравшим узлом с приоритетом  $l$  в успешном пакетном цикле:

$$d_{\text{succ}}^{W,k,l}(n) = \sum_{s=1}^{W \cdot k} s p_{\text{succ}(1)}^{W,k,l,s'}(n). \quad (20)$$

В выражении (20)  $p_{\text{succ}(1)}^{W,k,l,s'}(n)$  – условная вероятность того, что определённый узел с приоритетом  $l$  выигрывает состязание за канал с номером слота  $s$ , где  $s$  находится в промежутке от 1 до  $W \cdot k$ , при условии, что рассматриваемый пакетный цикл успешен:

$$p_{\text{succ}(1)}^{W,k,l,s'}(n) = \frac{p_{\text{succ}(1)}^{W,k,l,s}(n)}{p_{\text{succ}(1)}^{W,k,l}(n)}. \quad (21)$$

Среднее количество слотов, выбираемых узлами сети, безотносительно к приоритету, определяется как:

$$d_{\text{succ}}(n) = \sum_{s=1}^{W \cdot k} \sum_{l=0}^{l_{\text{max}}} n_l \cdot p_{\text{succ}(1)}^{W,k,l,s'}(n). \quad (22)$$

Среднее количество слотов, когда происходит конфликт,

$$d_{\text{coll}}(n) = \sum_{s=1}^{W \cdot k} \left( \frac{s}{W \cdot k} \right)^{\sum_{l=0}^{s-1} n_l - 1}. \quad (23)$$

С учётом формулы (4) выражение (17) принимает вид:

$$\tau = \beta_1 + [d_{\text{coll}}(n) \cdot p_{\text{coll}}(n) + d_{\text{succ}}(n) \cdot p_{\text{succ}}(n) - 1] \cdot \beta_2 + PktLength \cdot \beta_3. \quad (24)$$

С учётом полученных формул выражение для вычисления среднего времени задержки доступа к каналу по формуле (2) принимает вид:

$$t_{\text{mean}} = \frac{\tau}{n} \cdot \sum_{l=0}^{l_{\text{max}}} \frac{n_l}{p_{\text{succ}(1)}^l} - PktLength \cdot \beta_3. \quad (25)$$

### 3. Исходные данные для анализа производительности сети

Получено, что для оценки средней задержки доступа  $t_{\text{mean}}$  по формуле (25) должны быть вычислены следующие параметры:

- вероятность успешной передачи  $p_{\text{succ}}(n)$ ,  $p'_{\text{succ}(1)}$  и вероятность коллизий  $p_{\text{coll}}(n)$ ;
- среднее количество слотов при успешной передаче  $d_{\text{succ}}(n)$  и среднее число слотов, когда происходит коллизия  $d_{\text{coll}}(n)$ .

Исходными данными для вычисления указанных параметров и средней задержки доступа являются:

- данные протокола:
  - $\beta_1, \beta_2, \beta_3$  – временные интервалы,
  - $W$  – ширина базового состязательного окна,
  - $k = BL$  – уровень отставания (backlog) в сети;
- данные сегмента сети:
  - $n, n_l$  – количество активных узлов в сети и количество активных узлов с приоритетом  $l$ ,
  - $l_{\text{max}}$  – максимальное количество приоритетных слотов в системе,
  - $PktLength$  – длина пакета в битах.

Все указанные исходные данные на протяжении функционирования сегмента сети носят неизменный характер, за исключением числа активных узлов, соперничающих за канал, и параметра отставания канала. Как утверждалось ранее, отставание изменяется в зависимости от загрузки сети согласно представленному алгоритму, воздействует на размер состязательного окна во время сетевой работы для достижения допустимого уровня коллизий в сети. Поэтому для произведения полного анализа производительности необходимо использовать устойчивое состояние сети со средней величиной отставания. Для нахождения устойчивого состояния системы должны использоваться аналитический и имитационные методы моделирования. Разработке и анализу таких моделей посвящены работы [4, 5].

#### Заключение

В статье представлено подробное описание алгоритма доступа к среде передачи рр-CSMA-протокола. Представлен подход к оценке производительности LonWorks-сети, в основе которой лежит анализ

уровня доступа к среде протокола *LonTalk*. Получены аналитические соотношения для оценки производительности Lon-сети, учитывающие уровни приоритетности узлов.

### **Библиографический список**

1. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы: учебник для вузов. – 4-е изд. – СПб.: Питер, 2010.
2. Тирш Ф. Введение в технологию LonWorks. – М.: Энергоатомиздат, 2001.
3. LonTalk protocol specification, ANSI/CEA-709.1-B. – United States: ISO/IEC JTC 1 SC 25, 2006.
4. Marek Miśkiewicz. Analysis of Mean Access Delay in Variable-Window CSMA // Sensors. – Kraków, Poland: MDPI, 2007.
5. Moshe Kam. Collision Resolution Simulation for Distributed Control Architectures using LonWorks // IEEE International Conference on Automation Science and Engineering. – Edmonton, Canada: IEEE, 2005.

Получено 05.09.2012