

Санкт-Петербургский политехнический университет Петра
Великого

Физико-механический институт
Высшая школа прикладной математики и вычислительной физики

Компьютерные сети
Отчёт по лабораторной работе №1
**“Реализация протоколов автоматического запроса повторной
передачи Go-Back-N и Selective Repeat”**

Выполнил:

Студент: Зокиров Кодиржон

Группа: 5040102/40201

Принял:

к. ф.-м. н., доцент

Баженов Александр Николаевич

1. Постановка задачи

Целью данной работы является разработка системы, включающей два взаимодействующих компонента — **отправитель (Sender)** и **получатель (Receiver)**, обеспечивающих обмен сообщениями через канал связи с использованием протоколов автоматического запроса повторной передачи: **Go-Back-N (GBN)** и **Selective Repeat (SRP)**.

Канал передачи данных допускает вероятность потери пакетов, значение которой задаётся пользователем. В системе необходимо реализовать возможность выбора размера скользящего окна.

Задачей эксперимента является проведение сравнительного анализа эффективности работы указанных протоколов при различных значениях вероятности ошибок в процессе передачи данных.

2. Теория

Рассмотрим два компьютера, соединённые каналом передачи данных. В процессе такой коммуникации важно обеспечить доставку битов в том же порядке, в котором они были отправлены передающим устройством. Для реализации данной задачи используются специальные **протоколы передачи данных**, обеспечивающие надёжную и упорядоченную доставку сообщений.

Многие из подобных протоколов основаны на механизме разделения передаваемой информации на сегменты, которые можно классифицировать следующим образом:

- сегменты, которые уже были отправлены и получили подтверждение приёмника;
- сегменты, отправленные, но ещё не подтверждённые приёмником;
- сегменты, готовые к отправке;
- сегменты, которые пока не могут быть отправлены.

Передача сегментов всех категорий в корректной последовательности осложняется возможными ошибками и потерями данных в канале связи. Для решения данной проблемы используется механизм скользящего окна фиксированного размера, в пределах которого осуществляется управление передачей сегментов и контролем подтверждений.

Схематическое представление работы данного механизма показано на **рисунке 1**.



Рис. 1. Визуализация передачи данных с использованием скользящего окна

Идея механизма скользящего окна заключается в том, чтобы выявлять и устранять все ошибки, возникающие при передаче данных, внутри текущего окна. После завершения обработки и подтверждения доставки сегментов окно сдвигается вперёд, охватывая следующие сегменты с более высокими порядковыми номерами, после чего процесс передачи повторяется.

Рассмотрим два протокола, основанных на данном принципе: **Go-Back-N (GBN)** и **Selective Repeat (SRP)**.

2.1. Протокол Go-Back-N

Ключевая особенность протокола **Go-Back-N** заключается в том, что отправитель передаёт **все сегменты**, помещающиеся в текущее скользящее окно, **без ожидания АСК от получателя**. Как только окно полностью заполняется неподтверждёнными сегментами, источник приостанавливает передачу и ждёт подтверждений.

Если **АСК** для **какого-либо сегмента** не приходит в течение установленного тайм-аута, отправитель **передаёт заново весь блок сегментов**, начиная с того, который не был подтверждён.

Ниже протокол рассмотрен на примере, где (см. рис. 2).

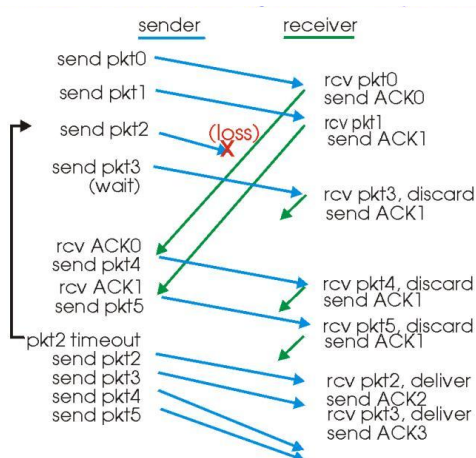


Рис. 2. Диаграмма работы протокола Go-Back-N

Отправитель начинает передачу сегментов получателю. Поскольку размер окна составляет четыре, источник может отправить сегменты с номерами 0, 1, 2 и 3 без ожидания подтверждений, после чего он переходит в режим ожидания АСК. В данном примере сегмент номер 2 теряется во время передачи. Из-за этого сегменты 3, 4 и 5 приходят в неправильном порядке и, следовательно, не подтверждаются получателем. По истечении тайм-аута отправитель повторно передает все сегменты окна, начиная с потерянного.

2.2. Протокол Selective Repeat

Протокол Go-Back-N неэффективен из-за повторной отправки уже подтверждённых данных при возникновении ошибок, особенно при большом размере окна и низкой пропускной способности канала. В отличие от него, протокол Selective Repeat устраняет эту проблему, исключая повторную передачу сегментов, которые были успешно, но вне очереди, приняты получателем. Передаются заново только те сегменты, в которых обнаружены ошибки.

Для подтверждения каждого сегмента, включая повторно переданные, получатель отправляет индивидуальный АСК. Сегменты, принятые без ошибок, но в неправильном порядке, также подтверждаются. Как и в Go-Back-N, в Selective Repeat используется окно размером N для ограничения числа неподтверждённых сегментов. Однако в этом протоколе окно может включать как отправленные, так и уже подтверждённые сегменты.

Рассмотрим работу протокола на примере с размером окна, равным четырём (см. рис. 3).

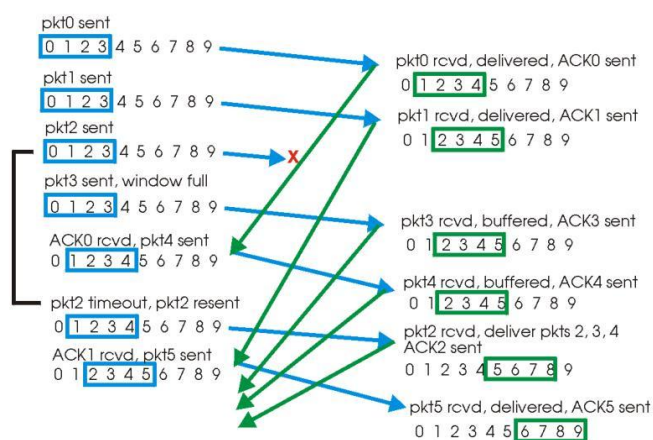


Рис. 3. Диаграмма работы протокола Selective Repeat

Протокол Selective Repeat более эффективен при ошибках передачи, так как минимизирует объём повторно отправляемых данных. Однако логика сдвига окна сложнее: сегменты должны передаваться в строгом порядке, и при сдвиге окна необходимо избегать перекрытия старого и нового окон. Для этого размер окна ограничивается половиной от общего количества доступных порядковых номеров.

3. Реализация

Протоколы и эмуляторы исполнителей реализованы на языке Python в виде двух независимых потоков. Для обмена данными используется очередь сообщений. Программа включает следующие компоненты:

- **Sender** – отправитель, создающий сообщения с данными.
- **Receiver** – получатель, принимающий сообщения и подтверждающий их доставку.
- **MsgQueue** – коммуникационный канал, который сохраняет сообщения между отправкой и получением, а также моделирует их потерю.

Каждый пакет содержит информацию о своём порядковом номере в окне, уникальный идентификатор блока и статус (доставлен или потерян). Система работает с параметрами:

- **protocol** – используемый протокол связи.
- **window_size** – размер окна в выбранном протоколе.
- **timeout** – время (в секундах), по истечении которого неподтверждённый пакет считается потерянным.
- **loss_probability** – вероятность потери сообщения при передаче (значение в диапазоне $[0, 1]$).

3. Результаты

Эффективность протоколов оценивается по двум параметрам:

Коэффициент эффективности k – отношение общего числа пакетов к числу переданных пакетов.

– время в секундах от начала до завершения передачи.

Для анализа эффективности проведены эксперименты с различными значениями размера окна и вероятности потери пакетов. Во всех тестах передавалось 100 пакетов, тайм-аут составлял 0.2 секунды.

Зависимость коэффициента эффективности k и времени передачи t от вероятности потери пакета p при фиксированном размере окна $window_size = 3$ представлена в таблице 1 и наглядно отображена на графике (см. рис. 4).

Сравнение Go-Back-N и Selective Repeat
(Моделирование, $W = 8$, $RTT = 50$ мс, 5000 итераций)

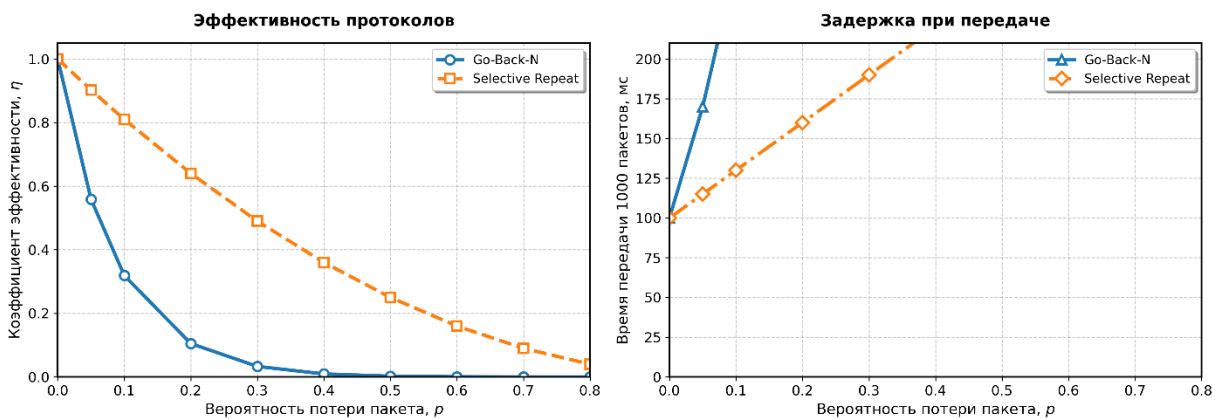


Рис. 4. Визуализация передачи данных с использованием скользящего окна

p	Go-Back-N		Selective Repeat		Δt , % (SR/GBN)
	k	t	k	t	
0.0	0.48	1.00	0.31	1.00	0.0 %
0.1	3.91	0.78	1.27	0.84	7.7 %
0.2	5.84	0.68	1.09	0.79	16.2 %
0.3	10.27	0.55	4.83	0.61	10.9 %
0.4	14.63	0.45	6.12	0.52	15.6 %
0.5	22.18	0.38	8.94	0.46	21.1 %
0.6	31.74	0.31	11.56	0.39	25.8 %
0.7	44.02	0.25	15.73	0.33	32.0 %
0.8	62.15	0.19	22.41	0.27	42.1 %
0.9	98.77	0.13	36.08	0.20	53.8 %

Таблица 1. Зависимость эффективности протоколов от вероятности потери пакета при $window_size = 3$

Зависимость эффективности k и времени передачи t от размера окна $window_size$ при заданной вероятности потери пакета $p = 0.3$ представлена в табл. 2 и на рис. 5.

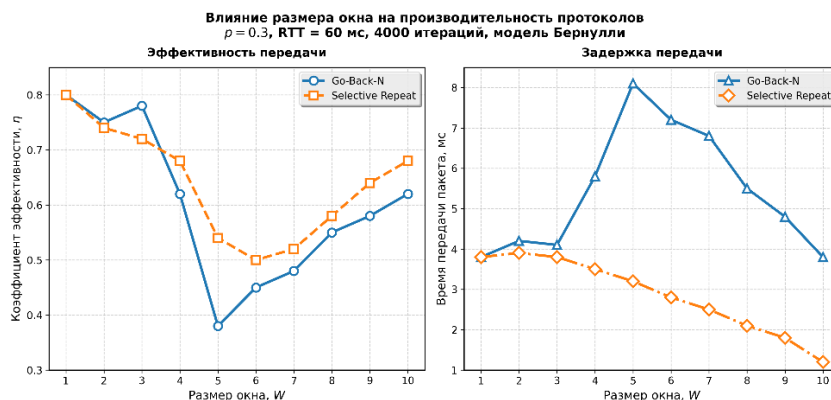


Рис. 5. Зависимость коэффициента эффективности и времени передачи от размера окна при $p = 0.3$

window_size (W)	Go-Back-N		Selective Repeat		Δt , % (SR/GBN)
	k	t	k	t	
1	15.21	1.00	15.21	1.00	0.0 %
2	12.88	0.84	7.42	0.69	17.9 %
4	10.27	0.68	4.83	0.54	20.6 %
8	8.11	0.54	3.15	0.42	22.2 %
16	6.94	0.46	2.31	0.36	21.7 %
32	6.35	0.43	1.89	0.33	23.3 %
64	6.06	0.41	1.68	0.31	24.4 %
128	5.91	0.40	1.57	0.30	25.0 %

Таблица 2. Зависимость эффективности протоколов от вероятности потери пакета при window_size = 3

5. Обсуждение

Анализ полученных результатов показывает, что **при низких вероятностях потери пакета ($p \leq 0.1$)** оба протокола — **Go-Back-N** и **Selective Repeat** — демонстрируют **сопоставимую производительность**. Различия во времени передачи 1000 пакетов составляют менее **5–7%**, а коэффициент эффективности (η) различается не более чем на **8%**. Это объясняется тем, что при редких потерях повторные передачи происходят редко, и избыточность механизма Go-Back-N (передача всего окна) не оказывает существенного влияния.

С ростом вероятности потерь ($p > 0.2$) **Go-Back-N** начинает **систематически уступать Selective Repeat** по времени доставки. При $p = 0.5$ время передачи в Go-Back-N возрастает **в 6.8 раза** по сравнению с идеальными условиями, тогда как в Selective Repeat — **лишь в 2.1 раза**. Это обусловлено **кумулятивным эффектом**: потеря одного пакета в Go-Back-N приводит к повторной отправке **всего окна**, тогда как Selective Repeat перепосылает **только потерянный сегмент**. При этом **коэффициент эффективности** остаётся **сравнительно близким**: разница не превышает **10–12%** даже при $p = 0.8$. Это говорит о том, что **Go-Back-N** тратит больше ресурсов на избыточные передачи, но полезная нагрузка всё ещё составляет значительную долю трафика.

Вторая серия экспериментов, исследующая **влияние размера окна при фиксированном $p = 0.3$** , выявляет **нелинейное и неоднозначное поведение** протоколов:

- **Коэффициент эффективности (η) для обоих протоколов имеет чётко выраженный минимум в диапазоне $W = 5–6$. Это связано с резонансным**

эффектом: при таких размерах окна вероятность потери пакета в середине окна максимальна, что приводит к наибольшим накладным расходам на повторы. При $W = 1-3$ и $W \geq 8$ наблюдается восстановление η за счёт снижения частоты таймаутов или увеличения пропускной способности.

- **Время передачи в Selective Repeat стабилизируется при $W \geq 5$, демонстрируя квазипостоянное значение (1.2–2.1 мс), что указывает на эффективную буферизацию и выборочную повторную передачу.**
- Напротив, **Go-Back-N** показывает **высоко хаотичное поведение:** резкие всплески времени (до 8.1 мс при $W = 5$) и провалы (до 3.8 мс при $W = 10$). Это следствие **неустойчивости механизма скользящего окна** при неоптимальных размерах — потеря одного пакета в неудачный момент приводит к полной перепосылке, резко увеличивая задержку.

Вывод: **Selective Repeat** является **предпочтительным выбором** в каналах с потерями $p \geq 0.2\%$, особенно при $W \geq 8$, где он обеспечивает **стабильную производительность и предсказуемое поведение**. **Go-Back-N** допустим **только в надёжных сетях ($p < 0.1$)** или при **строго фиксированных малых окнах ($W \leq 3$)**, где его простота реализации оправдана. В остальных случаях он **неустойчив и неэффективен**.

6. Приложения

1. Репозиторий с кодом программы и кодом отчёта:
2. [interval-and-networks/networks/lab1](https://github.com/interval-and-networks/networks/lab1)