МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

УДК 519.7 DOI 10.17223/20710410/71/3

ВИЗАНТИЙСКОЕ СОГЛАШЕНИЕ И ШИРОКОВЕЩАТЕЛЬНАЯ ПЕРЕДАЧА

С. М. Рацеев

Ульяновский государственный университет, г. Ульяновск, Россия

E-mail: ratseevsm@mail.ru

Византийское соглашение и широковещательная передача являются двумя фундаментальными протоколами и важнейшими строительными блоками в безопасных многосторонних вычислениях, поэтому повышение их эффективности представляет интерес как для теоретической, так и для практической криптографии. В данной обзорной работе приводятся протоколы византийского соглашения, протоколы широковещательной передачи, а также протоколы расширения для протоколов византийского соглашения и широковещательной передачи.

Ключевые слова: *византийское соглашение, широковещательная передача, криптографический протокол.*

BYZANTINE AGREEMENT AND BYZANTINE BROADCAST

S. M. Ratseev

Ulyanovsk State University, Ulyanovsk, Russia

Byzantine agreement and byzantine broadcast are the two most fundamental problems and essential building blocks in secure multiparty computations, and improving their efficiency is of interest to both theorists and practitioners. In this survey, we describe the most important constructions of Byzantine agreement and Byzantine broadcast, as well as their extension protocols.

Keywords: *Byzantine agreement, Byzantine broadcast, cryptographic protocol.*

Введение

Задача (византийской) широковещательной передачи заключается в том, что некоторый назначенный участник (отправитель) отправляет сообщение всем участникам, причём все (честные) получатели должны получить одинаковое сообщение, несмотря на то, что некоторые нечестные участники могут вести себя произвольным образом. Аналогично, византийское соглашение (Byzantine agreement) позволяет всем (честным) участникам, у каждого из которых имеется входное сообщение, определить одно и то же выходное сообщение.

Наиболее важна задача построения эффективных протоколов византийского соглашения и широковещательной передачи для входных сообщений большой длины, поскольку такие протоколы широко используются в качестве строительных блоков

безопасных многосторонних вычислений [1]. Простым решением для сообщения длины l является применение l протоколов трансляции (византийского соглашения) для каждого бита этого сообщения в отдельности. Такой подход имеет коммуникационную сложность $\Omega(ln^2)$ бит, где n — число участников. Более эффективным решением для сообщения большой длины l является применение протоколов расширения для протоколов византийского соглашения и широковещательной передачи. В этом случае можно достичь коммуникационной сложности $O(ln)$ бит.

В данной работе исследуются как протоколы византийского соглашения и широковещательной передачи, так и протоколы расширения для этих протоколов. Рассматриваются информационно-теоретически и криптографически безопасные протоколы. Криптографическая система (протокол) обладает свойством *информационно-теоретической безопасности*, если ни один противник не может взломать систему, независимо от того, насколько он силён, т. е. он может обладать неограниченными вычислительными возможностями. Если же криптографическая система удовлетворяет требованиям *криптографической* (*вычислительной*) *безопасности*, то это означает, что она безопасна только до тех пор, пока противник располагает ограниченными вычислительными ресурсами. Часто это тот случай, когда используются такие инструменты, как асимметричные шифры, с которыми связана вычислительная проблема, которую противник не должен быть в состоянии решить, чтобы гарантировать безопасность системы.

Оракулом в теории вычислений называется внешнее (по отношению к алгоритмам) устройство, которое в ответ на запрос произвольного алгоритма выдаёт значение некоторой функции на этом запросе. При этом как обращение к оракулу, так и получение от него ответа занимают один такт работы алгоритма.

1. Византийское соглашение

Задачу византийского соглашения можно определить следующим образом. Пусть имеется n участников $P_1, \ldots, P_n$, среди которых t могут быть нечестными и контролироваться противником. Участники соединены попарно защищёнными и аутентифицированными каналами. У каждого участника P_i имеется своё входное значение $x_i \in \{0,1\}^*$. Цель протокола состоит в том, чтобы все честные участники согласовали общее выходное значение y.

Определение 1 (византийское соглашение). Протокол, в котором изначально каждый участник P_i имеет входное значение (сообщение) $x_i \in \{0,1\}^*$ и который завершается при вычислении выходного значения y_i, является протоколом *византийского соглашения*, устойчивого к активному противнику, контролирующему до t участников (т. е. t-безопасное византийское соглашение), если выполняются следующие свойства:

1) *достоверность* (validity): если все честные участники имеют на входе x, то каждый честный участник P_i имеет выходное значение $y_i = x$;
2) *договоренность* (agreement): любые два честных участника P_i и P_j определяют одинаковое выходное значение $y_i = y_j$, т. е. выходные значения всех честных участников одинаковы.

Заметим, что t-безопасное византийское соглашение имеет смысл только при $t < n/2$.

Рассмотрим протокол византийского соглашения BGP (Berman, Garay, Perry) [2] для однобитовых входных данных с полиномиальной коммуникационной сложностью. Под *коммуникационной сложностью* передачи данных протокола понимается общее количество битов, отправленных/полученных честными участниками во время вы-

полнения протокола (при этом учитываются только те биты, которые должны быть получены в соответствии со спецификацией протокола). Под *раундовой сложностью* понимается количество раундов, необходимых протоколу для завершения работы.

Протокол 1 (протокол BGP).
Пусть каждый участник P_i получает на вход бит $x_i \in \{0,1\}$.
Цикл: $k = 1, \ldots, t+1$:

— Р а у н д 1. Каждый участник P_i передает свой бит x_i всем остальным участникам. В конце раунда участник P_i для $b = 0, 1$ определяет

$$C_i^b = \begin{cases} 1, & \text{если } P_i \text{ получил } b \text{ от не менее чем } n-t \text{ участников,} \\ 0 & \text{иначе.} \end{cases}$$

Заметим, что при вычислении C_i^b учитывается и собственное входное значение x_i участника P_i.

— Р а у н д 2. Каждый участник P_i передаёт C_i^0 и C_i^1 всем остальным участникам. Пусть C_{ji}^b — значение, полученное участником P_i от P_j. В конце раунда участник P_i определяет

$$D_i^b = |\{j : C_{ji}^b = 1\}|, \quad b = 0, 1.$$

Участник P_i определяет своё (промежуточное) выходное значение y_i:

$$y_i = \begin{cases} 1, & D_i^1 > t, \\ 0, & D_i^1 \leqslant t. \end{cases}$$

— Р а у н д 3. Участник P_k (k — номер итерации цикла) передаёт значение y_k всем остальным участникам. После этого каждый участник P_i переопределяет своё значение y_i следующим образом: если $D_i^{y_i} < n - t$, то P_i переопределяет $y_i := y_k$; в противном случае y_i остаётся прежним.
Каждый участник P_i определяет $x_i := y_i$ (как входное значение для следующего шага итерации).

Теорема 1 [2]. При $t < n/3$ протокол 1 является t-безопасным протоколом византийского соглашения с коммуникационной сложностью $O(n^3)$ бит.

2. Широковещательная передача

Широковещательная передача позволяет участнику отправлять одно и то же сообщение всем участникам, и все участники уверены, что они получили одинаковые сообщения. Предполагать наличие широковещательного канала разумно только в ограниченных условиях, например, когда участники географически близки и могут использовать радиоволны. В большинстве случаев, особенно при выполнении протокола через Интернет, участники должны реализовывать широковещательный канал по каналам «точка — точка» (point-to-point network). Например, эмуляцию широковещательных протоколов можно создавать, используя каналы типа «точка — точка», при наличии инфраструктуры открытых ключей и электронных подписей. Один из таких протоколов приведён далее.

Отказоустойчивая широковещательная передача позволяет участнику распределять некоторое значение (сообщение) между набором участников, не доверяющих друг другу, которые попарно соединены каналами типа «точка — точка». Формальным требованием широковещательного протокола является то, что в конце протокола

все участники должны договориться о распределённом значении между всеми участниками. При этом должно быть гарантировано выполнение договорённости, даже если участник, который распределяет некоторое значение, является нечестным. Протоколы безопасных многосторонних вычислений обычно разрабатываются с учётом наличия широковещательных каналов. Однако в реальных сетях взаимодействие между участниками обычно происходит только с помощью каналов типа «точка — точка» и широковещательная передача должна эмулироваться с помощью защищённого протокола широковещательной передачи.

Определение 2 (широковещательная передача). Протокол для участников $\mathcal{P} = \{P_1, \ldots, P_n\}$, в котором назначенный участник (дилер) $D \in \mathcal{P}$ имеет некоторое (входное) значение M, называется протоколом *широковещательной передачи*, который является устойчивым к действиям активного противника (т. е. t-безопасной широковещательной передачи), контролирующего до t участников, если выполнены следующие условия:

1) *достоверность* (validity): если дилер честный, то все честные участники в качестве своего итогового (выходного) значения определят значение дилера M;
2) *договоренность* (agreement): даже если дилер нечестный, то итоговые (выходные) значения всех честных участников будут одинаковыми.

Пусть n участников $P_1, \ldots, P_n$ синхронно обменивается данными по защищённым и аутентифицированным каналам в полностью подключенной сети «точка — точка». Если канал, соединяющий двух участников, является защищённым, то противник ничего не может узнать о сообщениях, которыми обмениваются эти два участника. Аутентифицированный канал гарантирует, что никто не сможет изменить сообщение, передаваемое по каналу, во время его передачи. Отметим, что защищённые и аутентифицированные каналы могут быть реализованы с использованием криптографических примитивов, таких, как шифрование и электронная подпись.

Замечание 1. Если $t < n/2$, то на основе протокола t-безопасной широковещательной передачи можно построить протокол t-безопасного византийского соглашения. В этом случае каждый участник транслирует свое входное значение. Тогда каждый участник в качестве выходного значения определяет то значение, которое во время n трансляций повторяется чаще всего.

Обратно, на основе протокола t-безопасного византийского соглашения можно построить протокол t-безопасной широковещательной передачи. В этом случае участник (отправитель) D передаёт сообщение M всем участникам. После этого все участники запускают протокол византийского соглашения, в котором входным значением является значение, полученное от D.

Замечание 2. Наиболее популярным является предположение об аутентифицированной настройке, например, наличие инфраструктуры с открытыми ключами (Public-Key Infrastructure, PKI) для n участников. В этом случае все участники имеют набор из n открытых ключей для схемы подписи, где i-й ключ соответствует i-му участнику. Каждый честный участник имеет секретный ключ, сгенерированный честным путём, связанный с его собственным открытым ключом. Нечестные участники могут генерировать свои ключи произвольно.

При отсутствии аутентифицированной настройки t-безопасная широковещательная передача и t-безопасное византийское соглашение существуют тогда и только тогда, когда $t < n/3$ [2, 3]. Примером такого протокола византийского соглашения явля-

ется протокол 1 (BGP), на основе которого можно построить протокол широковещательной передачи без аутентифицированной настройки с учётом замечания 1.

При наличии аутентифицированной настройки при $t < n$ возможна t-безопасная широковещательная передача (протокол 2), а при $t < n/2$ возможно t-безопасное византийское соглашение (протокол 2 с учётом замечания 1).

Аутентифицированная настройка существует в двух вариантах: *информационно-теоретическая* и *криптографическая*. Информационно-теоретическая безопасность достигается с помощью использования псевдоподписи (pseudo-signature scheme) [4]. Криптографическая (вычислительная) безопасность достигается с помощью использования безопасной схемы электронной подписи, которую (почти) невозможно подделать.

Рассмотрим протокол широковещательной передачи, который устойчив к активному противнику, контролирующему до t участников. Наличие активного противника означает, что любой нечестный участник, находящийся под контролем противника, может произвольно отклоняться от предписанного протокола.

Модифицированный протокол Долева — Стронга (Dolev — Strong).

Пусть $P_1, \ldots, P_n$ — участники протокола широковещательной передачи; $D \in \{P_1, \ldots, P_n\}$ — дилер, который собирается транслировать бит (сообщение) $m \in \{0, 1\}$; Sign — некоторый алгоритм электронной подписи; sk_i — секретный ключ электронной подписи участника P_i, $i = 1, \ldots, n$. Приведём модифицированную версию протокола Долева — Стронга [5] в синхронной настройке для аутентифицированной широковещательной передачи для случая $t < n$.

Протокол 2 (модифицированный протокол Долева — Стронга).

Пусть дилер D обладает сообщением $m \in \{0, 1\}$.

— Э т а п 1 (действия дилера D и каждого участника P_i)
  — Дилер D отправляет всем участникам P_i подписанное сообщение $(m, \mathrm{Sign}_{sk_D}(m))$.
  — Каждый участник P_i определяет и инициализирует три множества: $AS_i = SET_0^i = SET_1^i = \varnothing$, где AS_i — одобренное участником P_i множество значений (Accepted Set); SET_0^i, SET_1^i — множества подписанных разными участниками сообщений $m = 0$ и $m = 1$ соответственно.

— Э т а п 2 (действия каждого участника P_i)
  В раундах $r = 1, \ldots, N + 1$ выполняются следующие шаги (в начале раунда с номером $r = 1$ все участники получили от дилера соответствующее сообщение):
  — Пусть участник P_i в r-м раунде получил от некоторого участника P_j сообщение (x, SET), где $x \in \{0, 1\}$; $SET = \{\mathrm{Sign}_{sk_{i_1}}(x), \ldots, \mathrm{Sign}_{sk_{i_s}}(x)\}$ — множество подписей, поставленных под сообщением x различными участниками $P_{i_1}, \ldots, P_{i_s}$, включая дилера D, причём $s \geqslant r$. Тогда участник P_i переопределяет свои множества следующим образом: $AS_i := AS_i \cup \{x\}$, $SET_x^i := SET_x^i \cup SET$.
  — Если $s < r$ или в сообщении (x, SET) нет подписи дилера D, то полученное сообщение игнорируется.
  — Если добавленного в множество AS_i сообщения x в этом множестве ранее не было (к началу r-го раунда), то участник P_i подписывает это сообщение $\mathrm{Sign}_{sk_i}(x)$, после чего всем остальным участникам передаёт сообщение $(x, SET_x^i \cup \{\mathrm{Sign}_{sk_i}(x)\})$.

— Э т а п 3 (действия каждого участника P_i)
  Если $AS_i = \{1\}$, то участник P_i в качестве своего итогового значения m_i определяет $m_i = 1$, в противном случае $m_i = 0$.

Теорема 2 [5]. Если $t < n$ и $N = t$, то модифицированный протокол Долева — Стронга является протоколом широковещательной передачи даже при наличии активного противника, контролирующего до t участников.

Протокол Долева — Стронга. Обобщим протокол 2 до случая сообщения m произвольной длины l.

Протокол 3 (протокол Долева — Стронга).
Пусть дилер D обладает сообщением $m \in \{0,1\}^l$.

— Э т а п 1 (действия дилера D и каждого участника P_i)
  — Дилер D отправляет всем участникам P_i подписанное сообщение $(m, \mathrm{Sign}_{sk_D}(m))$.
  — Каждый участник P_i определяет и инициализирует множество $AS_i = \varnothing$ — одобренное участником P_i множество сообщений.

— Э т а п 2 (действия каждого участника P_i)
  В раундах $r = 1, \ldots, N+1$ выполняются следующие шаги (в начале раунда с номером $r = 1$ все участники получили от дилера соответствующее сообщение):
  — Пусть участник P_i в r-м раунде получил от участника P_j сообщение $(x, \mathrm{Sign}_{sk_{i_1}}(x), \ldots, \mathrm{Sign}_{sk_{i_s}}(x))$, где $\{\mathrm{Sign}_{sk_{i_1}}(x), \ldots, \mathrm{Sign}_{sk_{i_s}}(x)\}$ — множество подписей, поставленных под сообщением x различными участниками $P_{i_1}, \ldots, P_{i_s}$, включая дилера D, причём $s \geqslant r$. Если $|AS_i| < 2$, то участник P_i переопределяет множество $AS_i := AS_i \cup \{x\}$, подписывает x своей подписью $\mathrm{Sign}_{sk_i}(x)$ и передаёт сообщение $(x, \mathrm{Sign}_{sk_{i_1}}(x), \ldots, \mathrm{Sign}_{sk_{i_s}}(x), \mathrm{Sign}_{sk_i}(x))$ всем остальным участникам.
  — Если $|AS_i| = 2$, или $s < r$, или в сообщении $(x, \mathrm{Sign}_{sk_{i_1}}(x), \ldots, \mathrm{Sign}_{sk_{i_s}}(x))$ нет подписи дилера D, то полученное сообщение игнорируется.

— Э т а п 3 (действия каждого участника P_i)
  Если $AS_i = \{x\}$ — одноэлементное множество, то участник P_i в качестве своего итогового значения m_i определяет $m_i = x$, в противном случае $m_i = 0$.

Теорема 3 [6]. Если $t < n$ и $N = t$, то протокол Долева — Стронга является протоколом широковещательной передачи даже при наличии активного (адаптивного) противника, контролирующего до t участников.

3. Широковещательная передача с прерыванием

В случае нечестного большинства и активного противника существуют протоколы безопасных многосторонних вычислений, которые не обладают свойствами гарантированного получения результатов и справедливости [7]. Следовательно, многие такие протоколы просто прерываются при обнаружении обмана, реализуя безопасность с прерыванием (security with abort) [8]. В частности, это означает, что либо протокол завершается успешно и каждый участник получит свои выходные данные, либо протокол прерывается, причём это может произойти даже после того, как противник узнал результаты вычислений, что может стать серьёзной проблемой в некоторых приложениях.

Учитывая соглашение о безопасности с прерыванием, широковещательный канал может быть эффективно реализован с использованием стандартного 2-раундового протокола *эхо-трансляции* [8], который приведён далее.

Определение 3 (широковещательная передача с прерыванием). Протокол для участников $\mathcal{P} = \{P_1, \ldots, P_n\}$, в котором назначенный участник (дилер) $D \in \mathcal{P}$ имеет

некоторое (входное) значение M, называется протоколом *широковещательной передачи с прерыванием*, который является устойчивым к действиям активного противника, контролирующего до t участников, если выполнены следующие условия:

1) *достоверность* (validity): если дилер честный, то каждый честный участник в качестве своего итогового (выходного) значения определит либо значение дилера M, либо $\perp$;
2) *договоренность* (agreement): если некоторый честный участник в качестве своего итогового значения определит $\widetilde{M}$, то каждый честный участник в качестве своего итогового значения определит либо $\widetilde{M}$, либо $\perp$;
3) *нетривиальность* (non-triviality): если все участники являются честными (включая дилера), то все участники в качестве своих итоговых значений определят M.

Протокол 4 (трансляция с прерыванием).

Пусть дилер D обладает сообщением M, предназначенным для трансляции.

— Дилер D передаёт сообщение M каждому участнику.
— Обозначим через M_i сообщение, полученное участником P_i от дилера D на предыдущем шаге. Если участник P_i не получил ничего от дилера на предыдущем шаге, то полагается $M_i = \perp$.

Каждый участник P_i передаёт сообщение M_i всем остальным участникам.

— Обозначим через M_{ji} сообщение, полученное участником P_i от участника P_j на предыдущем шаге. Участник P_i в качестве выходного значения определяет M_i, если для любого $j = 1, \ldots, n$, $j \neq i$, выполнено $M_{ji} = M_i$. В противном случае участник P_i определяет $\perp$.

Утверждение 1 [8]. Протокол 4 является протоколом широковещательной передачи с прерыванием, который является устойчивым к действиям активного противника, контролирующего до $t < n$ участников.

4. Протокол расширения для случая $t < n/3$

Исследуем протоколы расширения для византийского соглашения и широковещательной передачи. Это связано с задачей эффективной трансляции и византийским соглашением для длинных входных сообщений, поскольку такие протоколы широко используются. Простым решением трансляции l-битного сообщения является трансляция каждого бита в отдельности. Такой подход требует коммуникационной сложности $\Omega(ln^2)$ бит, где n — число участников, так как коммуникационная сложность для однобитного сообщения составляет $\Omega(n^2)$ бит [9]. Протоколы расширения для широковещательной передачи и византийского соглашения представляют собой конструкции для длинных сообщений, построенные на основе соответствующих протоколов для коротких сообщений (в частности, однобитных) и каналов связи типа «точка — точка».

Рассмотрим протокол византийского соглашения для случая $t < n/3$, который является оптимальным как для коммуникационной, так и для раундовой сложности [10]. Данный протокол имеет информационно-теоретическую безопасность, причём вероятность ошибки равна нулю (error-free), т. е. является детерминированным. Он основан на методах теории кодирования и теории графов. В протоколе использован алгоритм поиска (n, t)-звезды и коды Рида — Соломона.

Поиск (n, t)-звезды

Определим структуру данных под названием (n, t)-*звезда*, которую иногда будем называть просто звездой.

Пусть G является неориентированным графом с множеством вершин $\{P_1, \ldots, P_n\}$; $\mathcal{C} \subseteq \mathcal{D} \subseteq \{P_1, \ldots, P_n\}$ — некоторые подмножества вершин графа. Пара $(\mathcal{C}, \mathcal{D})$ называется (n,t)-*звездой*, если $|\mathcal{C}| \geqslant n-2t$, $|\mathcal{D}| \geqslant n-t$ и для любых $P_i \in \mathcal{C}$, $P_j \in \mathcal{D}$ ребро (P_i, P_j) принадлежит графу G.

Кликой в неориентированном графе $G = (V, E)$ называется подмножество вершин $\mathcal{C} \subseteq V$, для которого для любых двух вершин в $\mathcal{C}$ существует ребро, их соединяющее. Из определения (n,t)-звезды $(\mathcal{C}, \mathcal{D})$ следует, что $\mathcal{C}$ является кликой. Понятие звезды обобщает понятие клики. Если надмножество $\mathcal{D}$ множества $\mathcal{C}$ является кликой, то $(\mathcal{C}, \mathcal{D})$ является звездой.

В работе [11] приведён эффективный алгоритм проверки наличия звезды в графе при условии, что граф содержит клику размера не менее $n-t$ (алгоритм 1).

Напомним несколько понятий. Граф $\overline{G} = (V, \overline{E})$ называется дополнительным графом к графу $G = (V, E)$, если

$$\overline{E} = \{(u,v) : u, v \in V, u \neq v, (u,v) \notin E\},$$

т. е. дополнительный граф $\overline{G}$ содержит те же вершины, что и граф G, и любые две различные вершины в нем смежны в том и только в том случае, когда эти вершины не смежны в графе G. Множество вершин $U \subseteq V$ называется независимым, если никакие две его вершины не смежны. Множество рёбер $M \subseteq E$ называется паросочетанием, если никакие два его ребра не имеют общей вершины. Максимальное паросочетание в графе G — это такое паросочетание M, в котором количество входящих в его состав рёбер является максимальным среди всех паросочетаний в графе G. Две вершины называются соседями (или смежными вершинами), если в графе имеется ребро, их соединяющее.

Алгоритм 1. Эффективный алгоритм поиска звезды в графе

Вход: граф $G = (\{1, \ldots, n\}, E)$, параметр t.
Выход: звезда $(\mathcal{C}, \mathcal{D})$ или сообщение об её отсутствии.

1: Пусть $\overline{G} = (\{1, \ldots, n\}, \overline{E})$ — дополнительный граф; M — максимальное паросочетание в $\overline{G}$, найденное, например, с помощью алгоритма Эдмондса; N — множество всех вершин графа G, входящих в паросочетание M; $\overline{N} = \{1, \ldots, n\} \setminus N$.
2: Пусть T — множество таких вершин из $\overline{N}$, для каждой из которых найдутся вершины $j, k \in \{1, \ldots, n\}$, такие, что треугольник $(i,j), (i,k), (j,k)$ принадлежит графу $\overline{G}$ и $(j,k) \in M$: $T = \{i \in \overline{N} : \exists j, k\, (j,k) \in M, (i,j), (i,k) \in \overline{G}\}$. Пусть $\mathcal{C} = \overline{N} \setminus T$.
3: Пусть B — множество вершин из N, для которых найдутся смежные вершины в $\overline{G}$: $B = \{j \in N : \exists i \in \mathcal{C}\, (i,j) \in \overline{G}\}$; $\mathcal{D} = \{1, \ldots, n\} \setminus B$.
4: **Если** $|\mathcal{C}| \geqslant n-2t$ и $|\mathcal{D}| \geqslant n-t$, **то**
5: **Вернуть** звезду $(\mathcal{C}, \mathcal{D})$,
6: **иначе**
7: **Вернуть** сообщение об отсутствии звезды.

Коды Рида — Соломона

В протоколе византийского соглашения используется обобщённый $[n, t+1, n-t]$-код Рида — Соломона [12] над полем $F = \mathrm{GF}(2^m)$, $n \leqslant 2^m$. Каждый элемент поля F можно представить в виде двоичного вектора длины m. С помощью обобщённого кода Рида — Соломона информационное сообщение над полем F длины $t+1$ кодируется в сообщение длины n над полем F.

Протокол византийского соглашения

Приведём протокол расширения для византийского соглашения [10].

Протокол 5 (протокол византийского соглашения для $t < n/3$).
Пусть каждый участник P_i обладает сообщением $m_i \in \{0,1\}^l$.
Оракул: оракул для широковещательной передачи сообщений небольшой длины.
Каждый участник P_i производит следующие действия:

1) l-Битное сообщение m_i разбивается на $t+1$ блоков $m_i = (m_{i0}, m_{i1}, \ldots, m_{it})$, где каждый блок m_{ij} имеет длину $l/(t+1)$ бит. С помощью обобщённого $[n, t+1, n-t]$-кода Рида — Соломона сообщение $(m_{i0}, m_{i1}, \ldots, m_{it})$ кодируется в кодовое сообщение $(s_{i1}, \ldots, s_{in})$. Значение s_{ii} передаётся всем участникам. Значение s_{ij} передаётся участнику P_j, $j = 1, \ldots, n$.
2) Двоичный вектор v_i длины n заполняется следующим образом:
$$v_i[j] = \begin{cases} 1, & s_{ij} = s_{jj} \text{ и } s_{ii} = s_{ji}, \\ 0 & \text{иначе}, \end{cases}$$
где s_{jj} и s_{ji} получено от P_j, $j = 1, \ldots, n$. С помощью оракула вектор v_i транслируется всем участникам.
3) Строится граф G с множеством вершин $\{P_1, \ldots, P_n\}$. Ребро (P_j, P_k) добавляется в граф, если $v_j[k] = 1$ и $v_k[j] = 1$ (j и k могут совпадать). Для графа G вызывается алгоритм 1. Возможны два случая:
   а) алгоритм 1 возвратил звезду $(\mathcal{C}, \mathcal{D})$. Вычисляется множество вершин $\mathcal{F}$ графа G, каждая из которых имеет не менее $t+1$ соседей из множества $\mathcal{C}$. Вычисляется множество вершин $\mathcal{E}$ графа G, каждая из которых имеет не менее $2t+1$ соседей из множества $\mathcal{F}$. Если $|\mathcal{E}| \geqslant 2t+1$, то $\mathcal{P}_{\text{sm}} := \mathcal{E}$ (“sm” означает “same message”). Если $|\mathcal{E}| < 2t+1$, то участник P_i полагает своим выходным значением заранее определённое значение $m^* \in \{0,1\}^l$ и прерывает протокол;
   б) звезда в графе G не найдена. Участник P_i полагает выходным значением заранее определённое значение $m^* \in \{0,1\}^l$ и прерывает протокол.
4) Пусть s_i — такое значение из множества $\{s_{ji} : P_j \in \mathcal{P}_{\text{sm}}\}$, которое в нём встречается чаще всего. Значение s_i передаётся всем остальным участникам.
5) Пусть $(s_1, \ldots, s_n)$ — вектор, в котором s_j получено от P_j, $j = 1, \ldots, n$. К нему применяется алгоритм декодирования для обобщённого $[n, t+1, n-t]$-кода Рида — Соломона, который возвращает информационное сообщение $(m_0, \ldots, m_t)$. Это сообщение определяется в виде выходного результата участника P_i.

Обозначим через $\mathcal{B}(s)$ коммуникационную сложность трансляции двоичного сообщения длины s. Простой конструкцией является трансляция (короткого) сообщения по одному биту. В этом случае коммуникационная сложность составляет $s\mathcal{B}(1)$ бит. Например, для протокола Долева — Стронга $\mathcal{B}(1) = O(n^2 + kn^3)$. Аналогичным образом через $\mathcal{A}(s)$ будем обозначать коммуникационную сложность византийского соглашения для двоичного сообщения длины s.

Теорема 4 [10]. Протокол 5 является (информационно-теоретически) t-безопасным протоколом византийского соглашения со следующими условиями: число раундов равно 3, коммуникационная сложность — $O\big(ln + n^2\mathcal{B}(1)\big)$ бит.

Замечание 3. Протокол широковещательной передачи получается на основе протокола византийского соглашения. В этом случае сначала отправитель передаёт всем участникам некоторое сообщение, а потом запускается протокол византийского соглашения.

5. Универсальная хеш-функция

Пусть $K = \{0, 1, \ldots, 2^\kappa - 1\}$ — множество ключей. Рассмотрим семейство функций $\mathcal{U} = \{U_k : k \in K\}$, где для каждого $k \in K$ функция U_k отображает элементы некоторого множества X в $\{0,1\}^\kappa$. Семейство $\mathcal{U}$ называется ε-*универсальным*, если для любых двух различных сообщений $m_1, m_2 \in X$ выполнено

$$\frac{|\{k \in K : U_k(m_1) = U_k(m_2)\}|}{|K|} \leqslant \varepsilon.$$

Данное условие (комбинаторного) определения универсальной хеш-функции можно записать на языке вероятностей:

$$\Pr[k \leftarrow K : U_k(m_1) = U_k(m_2)] \leqslant \varepsilon.$$

ε-Универсальную хеш-функцию можно построить следующим образом. Пусть $X = \{0,1\}^l$, $K = \mathrm{GF}(2^\kappa)$. Каждое сообщение $m \in X$ интерпретируется как многочлен f_m над $\mathrm{GF}(2^\kappa)$ степени не более $\lceil l/\kappa \rceil - 1$. Тогда значение хеш-функции U_k определяется следующим образом: $U_k(m) = f_m(k)$. Так как два различных многочлена степени не более $\lceil l/\kappa \rceil - 1$ могут совпадать не более чем в $\lceil l/\kappa \rceil - 1$ точках, для любых $m_1, m_2 \in X$, $m_1 \neq m_2$, выполнено

$$\frac{|\{k \in GF(2^\kappa) \mid U_k(m_1) = U_k(m_2)\}|}{2^\kappa} \leqslant \frac{\lceil l/\kappa \rceil - 1}{2^\kappa} \leqslant 2^{-\kappa} l.$$

Таким образом, построенное семейство $\mathcal{U}$ является $2^{-\kappa}l$-универсальной хеш-функцией.

6. Протоколы расширения для случая $t < n/2$

Рассмотрим протоколы расширения для византийского соглашения и широковещательной передачи для случая $t < n/2$. Коммуникационная сложность для однобитного сообщения составляет $\Omega(n^2)$ бит, поэтому если такой протокол использовать для каждого бита сообщения длины l, то получим коммуникационную сложность $\Omega(ln^2)$. Поэтому для длинных сообщений более практичны протоколы расширения для широковещательной передачи и византийского соглашения, которые имеют меньшую коммуникационную сложность, нежели $\Omega(ln^2)$. Более того, для достаточно больших l коммуникационная сложность таких протоколов составляет $O(ln)$ бит.

6.1. Информационно-теоретически безопасный протокол

Протокол [13] состоит из трёх этапов, каждый из которых «приближает» участников к соглашению. Протокол может быть прерван на любом из первых двух этапов при обнаружении (доказуемых) несоответствий между данными, предоставленными честными участниками. В этом случае каждый участник выбирает выходное сообщение по умолчанию, обозначаемое $\perp$. Если выполнение протокола дошло до третьего этапа, то в конце протокола все участники получат свои выходные значения. В протоколе используется ε-универсальная хеш-функция $\mathcal{U} = \{U_k : k \in K\}$. Пусть $\mathcal{P} = \{P_1, \ldots, P_n\}$.

Протокол 6 (протокол византийского соглашения для $t < n/2$).
Пусть каждый участник P_i обладает сообщением $m_i \in \{0,1\}^l$.
Оракул: оракул для широковещательной передачи сообщений небольшой длины.

Этап проверок

1) Каждый участник P_i выбирает случайным образом ключ k_i для хеш-функции U_k и вычисляет $h_i = (k_i, U_{k_i}(m_i))$, которое транслируется всем участникам.

2) Каждый участник P_j создает двоичный вектор v_j длины n следующим образом:

$$v_j[i] = \begin{cases} 1, & U_{k_i}(m_j) = U_{k_i}(m_i), \\ 0 & \text{иначе}, \end{cases} \qquad i = 1, \ldots, n.$$

Должно быть выполнено $v_j[j] = 1$, $j = 1, \ldots, n$. Каждый участник P_j транслирует вектор v_j.

3) Если не менее $n-t$ транслируемых векторов являются одинаковыми и в каждом из таких одинаковых векторов v_j выполнено $v_j[j] = 1$, то этот вектор обозначим через v, а через $\mathcal{P}_{\text{sm}}$ — множество участников, которые транслировали вектор v. Если транслировалось менее $n - t$ одинаковых векторов, то протокол прерывается.

Этап укрупнения

Пусть $\phi : \mathcal{P} \setminus \mathcal{P}_{\text{sm}} \to \mathcal{P}_{\text{sm}}$ — некоторая инъективная функция. С её помощью участник $\phi(P_j) \in \mathcal{P}_{\text{sm}}$ помогает участнику $P_j \in \mathcal{P} \setminus \mathcal{P}_{\text{sm}}$ получить корректное сообщение m:

1) Для каждого участника $P_j \in \mathcal{P} \setminus \mathcal{P}_{\text{sm}}$ участник $P_i = \phi(P_j)$ передаёт сообщение m_i участнику P_j, который обозначает полученное сообщение через $\widetilde{m}_j$.
2) Каждый участник $P_j \in \mathcal{P} \setminus \mathcal{P}_{\text{sm}}$ выбирает случайным образом ключ k_j, вычисляет и транслирует сообщение $(k_j, U_{k_j}(\widetilde{m}_j))$.
3) Каждый участник $P_i \in \mathcal{P}_{\text{sm}}$ создаёт двоичный вектор v_i длины $|\mathcal{P} \setminus \mathcal{P}_{\text{sm}}|$ следующим образом:

$$v_i[j] = \begin{cases} 1, & U_{k_j}(m_i) = U_{k_j}(\widetilde{m}_j), \\ 0 & \text{иначе}, \end{cases} \qquad j = 1, \ldots, |\mathcal{P} \setminus \mathcal{P}_{\text{sm}}|.$$

Каждый участник P_i транслирует вектор v_i.

4) Если не менее $n-t$ транслируемых векторов являются одинаковыми, то обозначим этот вектор через v, а через $\mathcal{P}_{\text{rej}} \subseteq (\mathcal{P} \setminus \mathcal{P}_{\text{sm}})$ — множество всех участников P_j, для которых j-я компонента вектора v равна нулю. Пусть

$$\mathcal{P}_{\text{ok}} = \mathcal{P} \setminus \mathcal{P}_{\text{rej}} \setminus \{\phi(P_j) : P_j \in \mathcal{P}_{\text{rej}}\}.$$

Заметим, что множество

$$\mathcal{P}_{\text{conf}} = \mathcal{P}_{\text{rej}} \cup \{\phi(P_j) : P_j \in \mathcal{P}_{\text{rej}}\} = \{P_j, \phi(P_j) : P_j \in \mathcal{P}_{\text{rej}}\}$$

состоит из участников, не менее половины которых являются нечестными, т. е. из каждой пары участников P_j и $\phi(P_j)$, $P_j \in \mathcal{P}_{\text{rej}}$, хотя бы один нечестный.
Каждый участник $P_i \in \mathcal{P}_{\text{sm}} \cap \mathcal{P}_{\text{ok}}$ определяет имеющееся у него сообщение m_i как выходное, а каждый участник $P_i \in \mathcal{P}_{\text{ok}} \setminus \mathcal{P}_{\text{sm}}$ определяет своё выходное сообщение $m_i = \widetilde{m}_i$. Если транслировалось менее $n - t$ одинаковых векторов, то протокол прерывается.

Этап утверждения сообщений

1) Каждый участник $P_i \in \mathcal{P}_{\text{ok}}$ вычисляет $d = \lceil (|\mathcal{P}_{ok}| + 1)/2 \rceil$, $c = \lceil (l+1)/d \rceil$ и многочлен f_m, где $m = (m_0, m_1, \ldots, m_{d-1})$, $f_m(x) = m_0 + m_1 x + \ldots + m_{d-1} x^{d-1}$. После этого P_i передаёт $y_i = f_m(i)$ (вектор длины c) каждому участнику множества $\mathcal{P} \setminus \mathcal{P}_{\text{ok}}$.
2) Каждый участник $P_i \in \mathcal{P}_{\text{ok}}$ выбирает случайным образом ключ k_i и передаёт набор $\big(k_i, U_{k_i}(f_m(1)), \ldots, U_{k_i}(f_m(n))\big)$ каждому участнику множества $\mathcal{P} \setminus \mathcal{P}_{\text{ok}}$.

3) Каждый участник $P_j \in \mathcal{P} \setminus \mathcal{P}_{\text{ok}}$ для каждого y_i, полученного от $P_i \in \mathcal{P}_{\text{ok}}$, делает следующее: участник P_j обладает $|\mathcal{P}_{\text{ok}}|$ наборами вида

$$\big(k_i, U_{k_i}(f_m(1)), \ldots, U_{k_i}(f_m(n))\big), \quad P_i \in \mathcal{P}_{\text{ok}}.$$

Значения $U_{k_i}(y_i)$ сравниваются с $U_{k_i}(f_m(i))$. Если не менее $d = \lceil (|\mathcal{P}_{\text{ok}}| + 1)/2 \rceil$ наборов содержат значения вида $U_{k_i}(y_i)$, то значение y_i принимается, в противном случае оно отвергается. После этого P_j интерполирует многочлен f_m по не менее d значениям y_i, восстанавливая при этом некоторое сообщение $\widetilde{m}_j$.

Опишем этапы протокола 6.

Этап проверок. Участники сравнивают между собой сообщения m_i с помощью сравнения образов хеш-функции и совместно определяют подмножество участников $\mathcal{P}_{\text{sm}}$, чьи сообщения равны между собой (вернее, равны образы хеш-функции). Этот этап может быть прерван при обнаружении несоответствий между сообщениями честных участников. Пусть $\mathcal{P}_{\text{hon}}$ — все честные участники. Более формально, этап проверок удовлетворяет следующим требованиям:

— если все честные участники $P_i \in \mathcal{P}_{\text{hon}}$ обладают одинаковыми входными сообщениями $m_i = m$, то этап проверок не будет прерван;
— все честные участники $P_i \in \mathcal{P}_{\text{hon}} \cap \mathcal{P}_{\text{sm}}$ обладают одинаковыми входными сообщениями $m_i = m$;
— если все честные участники $P_i \in \mathcal{P}_{\text{hon}}$ обладают одинаковыми входными сообщениями $m_i = m$, то все они принадлежат множеству $\mathcal{P}_{\text{sm}}$, т. е. $\mathcal{P}_{\text{hon}} \subseteq \mathcal{P}_{\text{sm}}$.

Этап укрупнения. Участники из множества $\mathcal{P}_{\text{sm}}$ помогают другим участникам (множества $\mathcal{P} \setminus \mathcal{P}_{\text{sm}}$) получить правильное сообщение. Это приводит к множеству участников $\mathcal{P}_{\text{ok}}$, состоящему из участников с одинаковыми сообщениями, причём большинство участников $\mathcal{P}_{\text{ok}}$ являются честными. Данный этап также может быть прерван при обнаружении несоответствий между сообщениями честных участников. Этап укрупнения удовлетворяет следующим требованиям:

— если все честные участники попали в множество $\mathcal{P}_{\text{sm}}$ на этапе проверок (т. е. выполнено $\mathcal{P}_{\text{hon}} \subseteq \mathcal{P}_{\text{sm}}$), то этап укрупнения завершается без прерывания;
— все честные участники $P_i \in \mathcal{P}_{\text{hon}} \cap \mathcal{P}_{\text{ok}}$ обладают одинаковыми выходными сообщениями $m_i = m$;
— выходные сообщения участников множества $\mathcal{P}_{\text{hon}} \cap \mathcal{P}_{\text{sm}}$ совпадают с сообщением m;
— большинство участников множества $\mathcal{P}_{\text{ok}}$ являются честными, т. е. выполнено неравенство $|\mathcal{P}_{\text{ok}} \cap \mathcal{P}_{\text{hon}}| > |\mathcal{P}_{\text{ok}}|/2$. Это следует из того, что не менее половины участников множества $\mathcal{P}_{\text{conf}} = \{P_j, \phi(P_j) : P_j \in \mathcal{P}_{\text{rej}}\}$ являются нечестными.

Этап утверждения сообщений. Участники множества $\mathcal{P} \setminus \mathcal{P}_{\text{ok}}$ должны иметь возможность получить сообщение m, которым владеют участники множества $\mathcal{P}_{\text{ok}}$. Но при этом участники из множества $\mathcal{P} \setminus \mathcal{P}_{\text{ok}}$ не должны запрашивать сообщения от любых участников множества $\mathcal{P}_{\text{ok}}$. Это связано с тем, что нечестные участники могут злоупотребить этой возможностью, запросив сообщения у каждого участника из $\mathcal{P}_{\text{ok}}$, что приведёт к коммуникационной сложности $\Omega(ln^2)$ бит. Для уменьшения этой сложности применён трюк с использованием многочлена.

Пусть $l \approx c \cdot d$ для подходящих c и d. Представим сообщение m в виде $m = (m_0, m_1, \ldots, m_{d-1})$, где m_i имеет длину c бит, $i = 0, 1, \ldots, d-1$. Рассмотрим поле $F = \text{GF}(2^c)$. Сообщению m поставим в соответствие многочлен $f_m(x) = m_0 + m_1 x + \ldots + m_{d-1} x^{d-1} \in \in F[x]$. Для восстановления многочлена $f_m(x)$ (следовательно, сообщения m) необходимо знать d значений многочлена f_m в различных точках. Пусть $d = \lceil (|\mathcal{P}_{\text{ok}}| + 1)/2 \rceil$,

$c = \lceil (l+1)/d \rceil$. В данном случае к сообщению добавляется дополнительный бит. Каждый участник $P_i \in \mathcal{P}_{\rm ok}$ передаёт значение $f_m(i)$ каждому участнику из множества $\mathcal{P} \setminus \mathcal{P}_{\rm ok}$. Каждый участник $P_j \in \mathcal{P} \setminus \mathcal{P}_{\rm ok}$ (с помощью участников из множества $\mathcal{P}_{\rm ok}$) может отличить корректные значения $f_m(i)$ от некорректных и в конечном итоге интерполировать многочлен f_m, получая при этом сообщение m.

Теорема 5 [13]. Протокол 6 является (информационно-теоретически) t-безопасным протоколом византийского соглашения с коммуникационной сложностью $O\big(ln + n^3k + (n^2+nk)\mathcal{B}(1)\big)$ бит, где k — параметр безопасности.

6.2. Криптографически безопасный протокол

Рассмотрим протокол, который является модификацией протокола 6 и имеет немного меньшую коммуникационную сложность, но обладает криптографической (вычислительной) безопасностью [10]. Пусть H — функция хеширования, устойчивая к обнаружению коллизий.

Протокол 7 (протокол византийского соглашения для $t < n/2$).
Пусть каждый участник P_i обладает сообщением $m_i \in \{0,1\}^l$.
Оракул: оракул для широковещательной передачи сообщений небольшой длины.

Этап проверок. Каждый участник P_i делает следующее.

1) Вычисляется значение функции хеширования $h_i = H(m_i)$, которое транслируется всем участникам.
2) Происходит проверка, сколько одинаковых свёрток транслируется. Если их не менее $n-t$, то обозначим эту свёртку через h, а через $\mathcal{P}_{\rm sm}$ — множество участников, которые транслировали свертку h. Если транслировалось менее $n-t$ одинаковых свёрток, то протокол прерывается.

Этап договорённости. Пусть $\phi : \mathcal{P} \setminus \mathcal{P}_{\rm sm} \to \mathcal{P}_{\rm sm}$ — некоторая инъективная функция. Например, участники множества $\mathcal{P} \setminus \mathcal{P}_{\rm sm}$ перебираются по порядку следования их индексов, которые отображаются в соответствующих участников множества $\mathcal{P}_{\rm sm}$, которые тоже перебираются по порядку следования индексов. С помощью этой функции участнику $P_j \in \mathcal{P} \setminus \mathcal{P}_{\rm sm}$ получить корректное сообщение m помогает участник $\phi(P_j) \in \mathcal{P}_{\rm sm}$.

Каждый участник P_i делает следующее:

1) Если $P_i \in \mathcal{P}_{\rm sm}$, то он определяет своё сообщение m_i как выходное.
2) Если $P_i \in \mathcal{P}_{\rm sm}$ и $P_i = \phi(P_j)$, то P_i передаёт сообщение m_i участнику P_j. Обозначим полученное участником P_j сообщение через $\widetilde{m}_j$.
3) Если $P_i \in \mathcal{P} \setminus \mathcal{P}_{\rm sm}$ и он получил на предыдущем шаге от участника P_j сообщение $\widetilde{m}_i = m_j$, где $P_j = \phi(P_i)$, то P_i проверяет, выполнено ли равенство $H(\widetilde{m}_i) = h$. Если равенство выполнено, то P_i транслирует значение 1 и определяет своё выходное сообщение $m_i = \widetilde{m}_i$, в противном случае P_i транслирует значение 0.
4) Строится множество $\mathcal{P}_{\rm conf}$, которое состоит из пар вида $\{P_j, \phi(P_j)\}$, $P_j \in \mathcal{P} \setminus \mathcal{P}_{\rm sm}$, причём P_j транслировал 0. Пусть $\mathcal{P}_{\rm ok} = \mathcal{P} \setminus \mathcal{P}_{\rm conf}$, $d = \lceil (|\mathcal{P}_{\rm ok}| + 1)/2 \rceil$, $c = \lceil (l+1)/d \rceil$.
5) Если $P_i \in \mathcal{P}_{\rm ok}$, то сообщение m_i преобразуется в многочлен $f_i \in {\rm GF}(2^c)$ степени не более $d-1$. Вычисляется значение $y_i = f_i(i)$ и вектор $H_i = \big(H(f_i(1)), \ldots, H(f_i(n))\big)$. Набор (y_i, H_i) передаётся каждому $P_j \in \mathcal{P} \setminus \mathcal{P}_{\rm sm}$, который транслировал значение 0.
6) Если $P_i \in \mathcal{P} \setminus \mathcal{P}_{\rm sm}$ и P_i транслировал 0, то для каждого y_j, полученного от $P_j \in \mathcal{P}_{\rm ok}$, делается следующее: участник P_i обладает $|\mathcal{P}_{\rm ok}|$ наборами вида $\big(H(f_j(1)), \ldots, H(f_j(n))\big)$, $P_j \in \mathcal{P}_{\rm ok}$. Значения $H(y_j)$ сравниваются с $H(f_j(j))$.

Если не менее $d = \lceil(|\mathcal{P}_{\text{ok}}| + 1)/2\rceil$ наборов содержат значения вида $H(y_j)$, то значение y_j принимается, в противном случае оно отвергается. После этого P_i интерполирует многочлен f по не менее d значениям y_j, восстанавливая некоторое сообщение $\widetilde{m}_i$, которое является его выходным значением.

Этап проверок обладает следующими свойствами:

— если все честные участники P_i начнут этап проверок с одинаковым сообщением $m_i = m$, то этап не будет прерван, причём все честные участники попадут в множество $\mathcal{P}_{\text{sm}}$;
— все честные участники множества $\mathcal{P}_{\text{sm}}$ обладают одинаковыми входными сообщениями.

Этап договоренности обладает следующими свойствами:

— большинство участников множества $\mathcal{P}_{\text{ok}}$ являются честными;
— выходные значения честных участников множеств $\mathcal{P}_{\text{sm}}$ и $\mathcal{P}_{\text{ok}}$ одинаковы;
— все честные участники обладают одинаковыми выходными сообщениями.

Теорема 6 [10]. Протокол 7 (за исключением пренебрежимо малой вероятности от параметра k) является (криптографически) t-безопасным протоколом византийского соглашения с коммуникационной сложностью $O\big(ln + n^3k + nk\mathcal{B}(1)\big)$ бит, где k — параметр безопасности.

7. Протоколы расширения широковещательной передачи для случая $t < n$

Рассмотрим криптографически (вычислительно) и информационно-теоретически безопасные протоколы расширения широковещательной передачи в случае $t < n$ [14]. На высоком уровне обе конструкции работают следующим образом: длинное сообщение (которое требуется транслировать) разбивается на блоки, а к каждому блоку применяется специальный протокол для трансляции блока.

В протоколах используется система контроля для разрешения споров между участниками (когда участники обладают разными значениями). Пусть Δ — множество неупорядоченных пар участников, причём $\{P_i, P_j\} \in \Delta$ тогда и только тогда, когда у P_i и P_j происходит спор по поводу того, чьё сообщение является правильным. В начале протокола множество Δ пустое. Во время выполнения протокола при возникновении спора соответствующая пара участников добавляется в это множество. Будем говорить, что множество Δ является *допустимым*, если для каждой пары $\{P_i, P_j\} \in \Delta$ участники P_i и P_j находятся в споре.

7.1. Криптографически безопасный протокол

Сначала рассмотрим протокол CryptoBlockBC, который предназначен для трансляции некоторого блока данных. Он вызывается в протоколе CryptoBC, где сообщение длины l разбивается на блоки и для каждого блока вызывается протокол CryptoBlockBC. В начале протокола CryptoBC множество Δ пусто. Если при вызове протокола CryptoBlockBC некоторая пара $\{P_i, P_j\}$ попадает в Δ, то она там остаётся до окончания протокола CryptoBC, т. е. множество Δ является глобальной переменной относительно протоколов CryptoBlockBC. Это значит, что если у участников P_i и P_j возник спор относительно значения некоторого блока, то этот спор продолжается и при вызове CryptoBlockBC для других блоков некоторого сообщения длины l.

В протоколе CryptoBlockBC используется устойчивая к коллизиям функция хеширования H. Пусть $\mathcal{P} = \{P_1, \ldots, P_n\}$ — множество участников; $\mathcal{P}_{\text{sm}}$ — множество участников, у которых сообщения m_i совпадают.

Протокол 8 (протокол CryptoBlockBC(m)).
Пусть дилер D обладает сообщением m.
Оракул: оракул для широковещательной передачи сообщений небольшой длины.

1) Каждый участник P_i инициализирует множество $\mathcal{P}_{\rm sm} = \{D\}$.
2) Дилер D транслирует значение хеш-функции $h = H(m)$.
3) Цикл: пока существует хотя бы одна пара различных участников $P_i, P_j \in \mathcal{P}$ с условием $P_i \in \mathcal{P}_{\rm sm}$, $P_j \in \mathcal{P} \setminus \mathcal{P}_{\rm sm}$ и $\{P_i, P_j\} \notin \Delta$, делается следующее:
   а) участник P_i передаёт сообщение m_i участнику P_j; обозначим полученное участником P_j сообщение через m_j;
   б) если $H(m_j) = h$, то участник P_j транслирует значение 1, в противном случае — значение 0;
   в) если P_j транслирует 1, то все участники добавляют участника P_j в множество $\mathcal{P}_{\rm sm}$, в противном случае все участники добавляют пару $\{P_i, P_j\}$ в множество Δ.
4) Каждый участник $P_i \in \mathcal{P}_{\rm sm}$ определяет в качестве выходного значения сообщение m_i. Каждый участник $P_i \in \mathcal{P} \setminus \mathcal{P}_{\rm sm}$ определяет в качестве выходного значения $\perp$.

Лемма 1 [14]. Пусть Δ — допустимое множество пар участников на момент вызова протокола CryptoBlockBC для блока m, $\Delta_{\rm e}$ — соответствующее множество на момент окончания протокола CryptoBlockBC для сообщения m, H — устойчивая к коллизиям функция хеширования с параметром безопасности k (длина свёртки). Тогда протокол CryptoBlockBC является t-безопасным протоколом широковещательной передачи, множество $\Delta_{\rm e}$ является допустимым, число раундов протокола CryptoBlockBC равно $O(n+d)$, коммуникационная сложность составляет $\mathcal{B}(k)+(n+d)(|m|+\mathcal{B}(1))$ бит, где $d = |\Delta_e| - |\Delta|$; $|m|$ — длина блока m.

Теперь рассмотрим протокол CryptoBC. Пусть q — некоторый параметр, $q \leqslant l$.

Протокол 9 (протокол CryptoBC).
Пусть дилер D обладает сообщением $m \in \{0,1\}^l$.
Оракул: оракул для широковещательной передачи сообщений небольшой длины.

1) Участники инициализируют множество Δ в виде пустого множества.
2) Дилер D разбивает сообщение m на q блоков $m = (m_1, \ldots, m_q)$.
3) Для $i = 1, \ldots, q$ вызывается протокол CryptoBlockBC(m_i). Пусть m_j^i — выходное сообщение участника P_j после i-го шага, $i = 1, \ldots, q$.
4) Каждый участник P_j в качестве итогового выходного сообщения определяет следующее: если для некоторого i выполнено $m_j^i = \perp$, то $m_j = \perp$, в противном случае $m_j = (m_j^1, \ldots, m_j^q)$.

Ввиду леммы 1 коммуникационная сложность протокола CryptoBC не превышает

$$\sum_{i=1}^{q}\Big(\mathcal{B}(k) + (n+d_i)(l/q + \mathcal{B}(1))\Big) = q\mathcal{B}(k) + \Big(qn + \sum_{i=1}^{q} d_i\Big)(l/q + \mathcal{B}(1)) \text{ бит.}$$

Так как $\sum\limits_{i=1}^{q} d_i \leqslant n^2$, коммуникационная сложность ограничена сверху числом $q\mathcal{B}(k) + (qn + n^2)(l/q + \mathcal{B}(1))$. При $q = n$ коммуникационная сложность ограничена сверху числом $2ln + n\mathcal{B}(k) + 2n^2\mathcal{B}(1)$.

Так как число раундов протокола CryptoBlockBC при i-м вызове равно $O(n + d_i)$, причём $\sum\limits_{i=1}^{q} d_i \leqslant n^2$, то число раундов протокола CryptoBC равно $O(n^2)$.

Таким образом, получаем следующее утверждение:

Теорема 7. При $t < n$ протокол CryptoBC при $q = n$ является (вычислительно) t-безопасным протоколом широковещательной передачи для сообщения длины l бит с числом раундов $O(n^2)$ и коммуникационной сложностью $O\big(ln + (n^2 + nk)\mathcal{B}(1)\big)$ бит, где k — параметр безопасности.

7.2. Информационно-теоретически безопасный протокол

Как и для случая криптографически безопасного протокола, все участники во время протокола ITBlockBC делятся на подмножества $\mathcal{P}_{\text{sm}}$ и $\mathcal{P} \setminus \mathcal{P}_{\text{sm}}$. Отличие от предыдущего случая в том, что множество $\mathcal{P}_{\text{sm}}$ не растёт монотонно, так как во время протокола ITBlockBC один и тот же участник может быть добавлен/удалён из множества $\mathcal{P}_{\text{sm}}$ несколько раз. При очередной итерации цикла происходит попытка перевести участника из $\mathcal{P} \setminus \mathcal{P}_{\text{sm}}$ в $\mathcal{P}_{\text{sm}}$. Пусть $\{P_i, P_j\}$ — некоторая пара участников, для которых $P_i \in \mathcal{P}_{\text{sm}}$, $P_j \in \mathcal{P} \setminus \mathcal{P}_{\text{sm}}$, $\{P_i, P_j\} \notin \Delta$. Участник P_i передаёт участнику P_j сообщение m_i. После этого участник P_j должен проверить, что полученное сообщение совпадает с сообщениями, которые имеются у участников множества $\mathcal{P}_{\text{sm}}$. Для этого P_j транслирует значение ключа k_j для ε-универсальной хеш-функции U_k, а дилер транслирует значение $h_j = U_{k_j}(m)$. Если участник P_j честно выбирает k_j случайным равновероятным образом, то с подавляющей вероятностью честные участники получат разные значения хеш-функции, если у них будут сообщения, отличающиеся от m. Если участник из $\mathcal{P}_{\text{sm}} \cup \{P_j\} \setminus \{D\}$ для своего сообщения получит значение хеш-функции, равное h_j, то он транслирует значение 1, иначе — 0. В данном случае не требуется, чтобы дилер D транслировал результат своей проверки, так как честный дилер всегда транслирует значение 1. Если все участники множества $\mathcal{P}_{\text{sm}} \cup \{P_j\} \setminus \{D\}$ транслируют 1, то участник P_j добавляется в множество $\mathcal{P}_{\text{sm}}$, в противном случае хотя бы один участник множества $\mathcal{P}_{\text{sm}} \cup \{P_j\}$ не обладает корректным сообщением, поэтому происходит поиск новых споров участников.

Важным отличием от криптографического случая является то, что споры могут возникать не только между P_i и P_j, но и между любыми двумя участниками в $\mathcal{P}_{\text{sm}}$. Чтобы найти такие споры, нужно знать историю формирования множества $\mathcal{P}_{\text{sm}}$. Для этого множество T содержит такие (упорядоченные) пары участников (P_i, P_j), для которых P_j получил от участника P_i сообщение m_i.

Протокол 10 (протокол ITBlockBC(m)).
Пусть дилер D обладает сообщением m.
Оракул: оракул для широковещательной передачи сообщений небольшой длины.

1) Каждый участник P_i инициализирует множества $\mathcal{P}_{\text{sm}} = \{D\}$ и $T = \varnothing$.
2) Цикл: пока существует хотя бы одна пара участников $P_i, P_j \in \mathcal{P}$ с условием $P_i \in \mathcal{P}_{\text{sm}}$, $P_j \in \mathcal{P} \setminus \mathcal{P}_{\text{sm}}$ и $\{P_i, P_j\} \notin \Delta$, делается следующее:
   а) участник P_i передаёт сообщение m_i участнику P_j. Обозначим полученное участником P_j сообщение через m_j. Пара (P_i, P_j) добавляется в T;
   б) участник P_j выбирает случайным равновероятным образом ключ $k_j \in K$ и транслирует его. После этого дилер D транслирует $h_j = U_{k_j}(m)$;
   в) каждый участник $P_k \in \mathcal{P}_{\text{sm}} \cup \{P_j\} \setminus \{D\}$ проверяет, выполнено ли равенство $U_{k_j}(m_k) = h_j$. Участник P_k транслирует 1, если равенство выполнено, в противном случае транслируется 0;
   г) если все участники множества $\mathcal{P}_{\text{sm}} \cup \{P_j\} \setminus \{D\}$ транслируют значение 1, то участник P_j добавляется в множество $\mathcal{P}_{\text{sm}}$, в противном случае делается следующее:

— для каждой пары вида $(P_k, P_r) \in T$, для которой P_k транслировал значение 1 (P_k может быть равен D), а участник P_r — значение 0, пара $\{P_k, P_r\}$ добавляется в множество Δ;
— $\mathcal{P}_{\text{sm}} := \{D\}$, $T := \varnothing$.

3) Каждый участник $P_i \in \mathcal{P}_{\text{sm}}$ определяет в качестве выходного значения сообщение m_i. Каждый участник $P_i \in \mathcal{P} \setminus \mathcal{P}_{\text{sm}}$ определяет в качестве выходного значения $\perp$.

Лемма 2 [14]. Пусть Δ — допустимое множество пар участников на момент вызова протокола ITBlockBC для сообщения (блока) m, Δ_{e} — соответствующее множество на момент окончания протокола ITBlockBC для сообщения m, $\mathcal{U}$ — универсальная функция хеширования с параметром безопасности κ. Тогда протокол ITBlockBC является t-безопасным протоколом широковещательной передачи, множество Δ_{e} является допустимым, число раундов протокола ITBlockBC равно $O(n + nd)$, коммуникационная сложность составляет $(n + nd)\big(|m| + 2\mathcal{B}(\kappa) + n\mathcal{B}(1)\big)$ бит, где $d = |\Delta_e| - |\Delta|$; $|m|$ — длина блока m.

Приведём протокол ITBC для широковещательной передачи сообщения длины l; он аналогичен протоколу CryptoBC.

Протокол 11 (протокол ITBC).
Пусть дилер D обладает сообщением $m \in \{0,1\}^l$.
Оракул: оракул для широковещательной передачи сообщений небольшой длины.

1) Участники инициализируют множество Δ в виде пустого множества.
2) Дилер D разбивает сообщение m на q блоков $m = (m_1, \ldots, m_q)$.
3) Для $i = 1, \ldots, q$ вызывается протокол ITBlockBC(m_i). Пусть m_j^i — выходное сообщение участника P_j после i-го шага, $i = 1, \ldots, q$.
4) Каждый участник P_j в качестве итогового выходного сообщения определяет следующее: если для некоторого i выполнено $m_j^i = \perp$, то $m_j = \perp$, в противном случае $m_j = (m_j^1, \ldots, m_j^q)$.

Ввиду леммы 2 коммуникационная сложность протокола ITBC не превышает

$$\sum_{i=1}^{q}(n + nd_i)\big(l/q + 2\mathcal{B}(\kappa) + n\mathcal{B}(1)\big) = n\Big(q + \sum_{i=1}^{q} d_i\Big)\big(l/q + 2\mathcal{B}(\kappa) + n\mathcal{B}(1)\big) \quad \text{бит.}$$

Так как $\sum\limits_{i=1}^{q} d_i \leqslant n^2$, то коммуникационная сложность ограничена сверху числом $n(q + n^2)\big(l/q + 2\mathcal{B}(\kappa) + n\mathcal{B}(1)\big)$. При $q = n^2$ коммуникационная сложность ограничена сверху числом $2ln + 2n^3\big(2\mathcal{B}(\kappa) + n\mathcal{B}(1)\big)$.

Так как число раундов протокола ITBlockBC при i-м вызове равно $O(n + nd_i)$, причём $\sum\limits_{i=1}^{q} d_i \leqslant n^2$, то число раундов протокола ITBC равно $O(n^3)$.

Теорема 8 [14]. При $t < n$ протокол ITBC при $q = n^2$ является (информационно-теоретически) t-безопасным протоколом широковещательной передачи для сообщения длины l бит с числом раундов $O(n^3)$ и коммуникационной сложностью $O\big(ln + (n^4 + n^3\kappa)\mathcal{B}(1)\big)$ бит, где κ — параметр безопасности.

Заключение

В таблице приведены параметры протоколов расширения для византийского соглашения и широковещательной передачи в синхронной настройке, где n — число участ-

ников; t — максимальное число нечестных участников; l — длина сообщения; k — параметр безопасности.

Порог	Безопасность	Протокол	**Коммуникационная сложность**	**Литература**
$t < n/3$	Информационно-теоретическая	Виз. соглашение, широковещание	$O\big(ln + n^2\mathcal{B}(1)\big)$	[10]
$t < n/3$	Информационно-теоретическая	Виз. соглашение, широковещание	$O\big(ln + n\mathcal{B}(1) + n^3\big)$	[15]
$t < n/2$	Информационно-теоретическая	Виз. соглашение, широковещание	$O\big(ln + n^3k + (n^2 + nk)\mathcal{B}(1)\big)$	[13]
$t < n/2$	Криптографическая	Виз. соглашение, широковещание	$O\big(ln + nk\mathcal{B}(1) + kn^3\big)$	[10]
$t < n/2$	Криптографическая	Виз. соглашение, широковещание	$O\big(ln + k\mathcal{A}(1) + kn^2\big)$	[15]
$t < (1-\varepsilon)n$	Криптографическая	Широковещание	$O\big(ln + k\mathcal{B}(1) + kn^2 + n^3\big)$	[15]
$t < n$	Криптографическая	Широковещание	$O\big(ln + (nk + n^3\log n)\mathcal{B}(1)\big)$	[10]
$t < n$	Криптографическая	Широковещание	$O\big(ln + (n^2 + nk)\mathcal{B}(1)\big)$	[14]
$t < n$	Информационно-теоретическая	Широковещание	$O\big(ln + (n^4 + n^3k)\mathcal{B}(1)\big)$	[14]

ЛИТЕРАТУРА

1. *Рацеев С. М.* Криптография. Безопасные многосторонние вычисления: учеб. пособие для вузов. 2-е изд., испр. и доп. СПб.: Лань, 2025. 540 с.
2. *Berman P., Garay J. A., and Perry K. J.* Bit-optimal distributed consensus // R. Baeza-Yates and U. Manber (eds.). Computer Science. Boston, MA: Springer, 1992. P. 313–322.
3. *Lamport L., Shostak R., and Pease M.* The Byzantine generals problem // ACM Trans. Program. Lang. Syst. 1982. V. 4. No. 3. P. 382–401.
4. *Pfitzmann B. and Waidner M.* Information-Theoretic Pseudosignatures and Byzantine Agreement for $t < n/3$. Technical Report RZ 2882. IBM Research, 1996.
5. *Kumaresan R.* Broadcast and Verifiable Secret Sharing: New Security Models and Round Optimal Constructions. PhD Thesis. University of Maryland at College Park, 2012.
6. *Dolev D. and Strong H. R.* Authenticated algorithms for Byzantine agreement // SIAM J. Computing. 1983. V. 12. No. 4. P. 656–666.
7. *Cleve R.* Limits on the security of coin flips when half the processors are faulty // Proc. STOC'86. Berkeley, California, USA, 1986. P. 364–369.
8. *Goldwasser S. and Lindell Y.* Secure computation without agreement // J. Cryptology. 2005. V. 18. No. 3. P. 247–287.
9. *Dolev D. and Reischuk R.* Bounds on information exchange for Byzantine agreement // Proc. PODS'82. Ottawa, Canada, 1982. P. 132–140.
10. *Ganesh C. and Patra A.* Optimal extension protocols for Byzantine broadcast and agreement // Distrib. Comput. 2021. V. 34. P. 59–77.
11. *Ben-Or M., Canetti R., and Goldreich O.* Asynchronous secure computation // Proc. STOC'93. N.Y., USA, 1993. P. 52–61.
12. *Рацеев С. М.* Элементы высшей алгебры и теории кодирования: учеб. пособие для вузов. 2-е изд., испр. и доп. СПб.: Лань, 2023. 684 с.
13. *Fitzi M. and Hirt M.* Optimally efficient multi-valued Byzantine agreement // Proc. PODC'06. Denver, Colorado, USA, 2006. P. 163–168.
14. *Hirt M. and Raykov P.* Multi-valued Byzantine broadcast: the $t < n$ case // LNCS. 2014. V. 8874. P. 448–465.
15. *Nayak K., Ren L., Shi E., et al.* Improved Extension Protocols for Byzantine Broadcast and Agreement. arXiv:2002.11321. `https://arxiv.org/abs/2002.11321`. 2020.

REFERENCES

1. *Ratseev S. M.* Kriptografiya. Bezopasnye mnogostoronnie vychisleniya [Cryptography. Secure Multiparty Computation]. St. Petersburg, Lan Publ., 2025. 540 p. (in Russian)
2. *Berman P., Garay J. A., and Perry K. J.* Bit-optimal distributed consensus. R. Baeza-Yates and U. Manber (eds.). Computer Science, Boston, MA, Springer, 1992, pp. 313–322.
3. *Lamport L., Shostak R., and Pease M.* The Byzantine generals problem. ACM Trans. Program. Lang. Syst., 1982, vol. 4, no. 3, pp. 382–401.
4. *Pfitzmann B. and Waidner M.* Information-Theoretic Pseudosignatures and Byzantine Agreement for $t < n/3$. Technical Report RZ 2882, IBM Research, 1996.
5. *Kumaresan R.* Broadcast and Verifiable Secret Sharing: New Security Models and Round Optimal Constructions. PhD Thesis, University of Maryland at College Park, 2012.
6. *Dolev D. and Strong H. R.* Authenticated algorithms for Byzantine agreement. SIAM J. Computing, 1983, vol. 12, no. 4, pp. 656–666.
7. *Cleve R.* Limits on the security of coin flips when half the processors are faulty. Proc. STOC'86, Berkeley, California, USA, 1986, pp. 364–369.
8. *Goldwasser S. and Lindell Y.* Secure computation without agreement. J. Cryptology, 2005, vol. 18, no. 3, pp. 247–287.
9. *Dolev D. and Reischuk R.* Bounds on information exchange for Byzantine agreement. Proc. PODS'82, Ottawa, Canada, 1982, pp. 132–140.
10. *Ganesh C. and Patra A.* Optimal extension protocols for Byzantine broadcast and agreement. Distrib. Comput., 2021, vol. 34, pp. 59–77.
11. *Ben-Or M., Canetti R., and Goldreich O.* Asynchronous secure computation. Proc. STOC'93, N.Y., USA, 1993, pp. 52–61.
12. *Ratseev S. M.* Elementy vysshey algebry i teorii kodirovaniya [Elements of Higher Algebra and Coding Theory]. St. Petersburg, Lan Publ., 2023. 684 p. (in Russian)
13. *Fitzi M. and Hirt M.* Optimally efficient multi-valued Byzantine agreement. Proc. PODC'06, Denver, Colorado, USA, 2006, pp. 163–168.
14. *Hirt M. and Raykov P.* Multi-valued Byzantine broadcast: the $t < n$ case. LNCS, 2014, vol. 8874, pp. 448–465.
15. *Nayak K., Ren L., Shi E., et al.* Improved Extension Protocols for Byzantine Broadcast and Agreement. arXiv:2002.11321. `https://arxiv.org/abs/2002.11321`, 2020.