

УДК 621.391 : 519.72 : 004.7

© 2021 г. К.Г. Бенерджи, М.К. Гупта

**КОМПРОМИССНОЕ СООТНОШЕНИЕ МЕЖДУ СТОИМОСТЬЮ  
ХРАНЕНИЯ И ВОССТАНОВЛЕНИЯ ДЛЯ НЕОДНОРОДНЫХ  
РАСПРЕДЕЛЕННЫХ СИСТЕМ ХРАНЕНИЯ ДАННЫХ<sup>1</sup>**

Рассматриваются неоднородные распределенные системы хранения данных (РСХД), имеющие переменную степень реконструкции, где каждый узел системы имеет свою собственную ширину восстановления и свой собственный объем памяти. В частности, устройство сбора данных может реконструировать файл с помощью некоторых  $k$  узлов системы, а в случае отказа узла систему можно восстановить по некоторому множеству активных узлов. С помощью границы минимального разреза исследуется фундаментальное компромиссное соотношение между стоимостью хранения и восстановления для нашей модели неоднородной РСХД. Кроме того, задача формулируется как оптимизационная задача двухкритериального линейного программирования для различных неоднородных РСХД. Для некоторых РСХД показано, что полученная граница минимального разреза точна.

*Ключевые слова:* облачное хранилище, коды для распределенного хранения данных, неоднородная распределенная система хранения данных, информационный поток, компромиссное соотношение между стоимостью хранения и восстановления.

**DOI:** 10.31857/S0555292321010022**§ 1. Введение**

Облачное хранилище – это распределенная система хранения данных (РСХД), в которой информация хранится в избыточном виде в различных узлах в виде закодированных пакетов. Для получения файла необходимо установить соединение с определенными узлами системы. В случае отказа узла его можно восстановить с помощью других узлов системы. Для таких РСХД требуется оптимизировать различные параметры системы, такие как объем памяти, ширина восстановления, доступность, надежность, безопасность и масштабируемость. Подобные РСХД используются во многих коммерческих системах, таких как Facebook, Yahoo!, IBM, Amazon и Microsoft Windows Azure [1–4].

В однородных РСХД (где каждый узел имеет одинаковый объем памяти и одинаковую степень восстановления) [5] закодированные пакеты данных из файла объема  $B$  распределяются между  $n$  узлами (каждый узел имеет объем памяти  $\alpha$ ) таким образом, что установив соединение с любыми  $k$  ( $< n$ ) узлами, можно получить весь файл. В случае отказа произвольного узла система восстанавливается путем загрузки  $\beta$  пакетов из любых  $d$  ( $< n$ ) активных узлов, называемых вспомогательными [5]. В таких системах надежность можно обеспечить или простым дублированием, или

<sup>1</sup> Предварительная английская версия данной статьи доступна по адресу <https://arXiv.org/abs/1503.02276>.

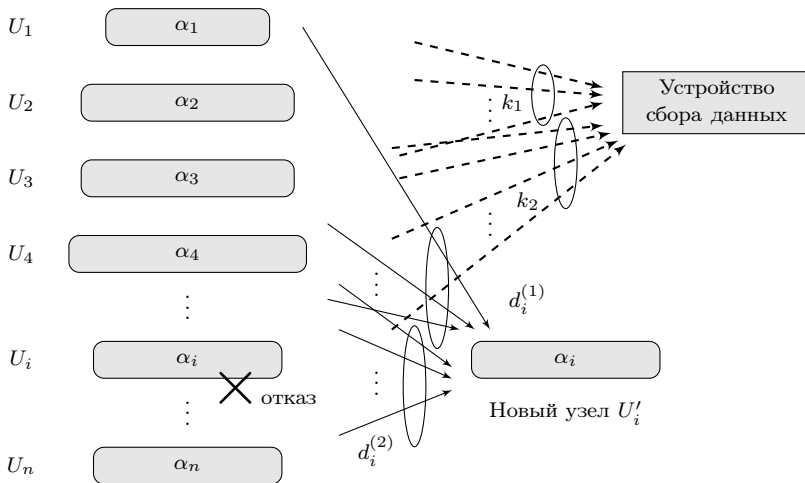


Рис. 1. Модель неоднородной РСХД. Здесь каждый узел имеет свои собственные объем памяти  $\alpha_i$ ,  $i = 1, 2, \dots, n$ , и ширину восстановления. В этой системе переменная степень реконструкции для устройства сбора данных равна  $k_t$ ,  $t \in \{1, 2\}$ . Степень восстановления при отказе произвольного узла  $U_i$  также неодинакова. Отказавший узел  $U_i$  восстанавливается с помощью некоторых  $d_i^{(t)}$  узлов

кодированием пакетов сообщений. В случае простого дублирования неэффективно используется объем памяти. С другой стороны, кодирование пакетов данных с помощью МДР-кодов (кодов с максимальным достижимым расстоянием) со стиранием приводит к неэффективности минимизации ширины восстановления в процессе восстановления узлов. Для оптимизации этих конфликтующих параметров в основополагающей работе [5] были введены регенерирующие коды (regenerating codes). В [6, 7] зависимость между объемом памяти узла  $\alpha$  и шириной восстановления  $d/\beta$  изучалась с помощью построения кривых компромисса для регенерирующих кодов. Все точки на кривой компромисса можно получить с помощью линейных сетевых кодов над конечными полями [8, 9]. По кривым компромисса минимизацией обоих параметров в разном порядке получают регенерирующие коды с минимальной шириной восстановления (minimum bandwidth regenerating codes) и регенерирующие коды с минимальным объемом памяти (minimum storage regenerating codes) [6]. Компромиссное соотношение между объемом памяти и шириной восстановления в задаче точного восстановления узлов изучалось в [10]. В [11] была вычислена нижняя граница пропускной способности разреза на ширину восстановления для конкретной постановки задачи с переменными параметрами в однородных РСХД. В [12] дан прекрасный обзор некоторых существующих результатов и моделей восстановления для РСХД. В недавней работе [13] изучено компромиссное соотношение между объемом памяти и шириной восстановления для линейных регенерирующих кодов с точным восстановлением узлов при  $k = d = n - 1$ .

Неоднородные РСХД более близки к реальным сценариям, где свойства узлов хранения данных не обязательно одинаковы в различных аспектах из-за различий в географической среде, стоимости запоминающих устройств и т.д. Многие такие неоднородные РСХД рассматривались в недавних работах [14–16]. В [17–19] изучалась задача распределения памяти с точки зрения максимизации вероятности успешного восстановления. Для неоднородных РСХД стоимость восстановления можно снижать, позволяя вспомогательным узлам кодировать слова, получаемые из других узлов [20]. В [21–25] исследовалось компромиссное соотношение между объемом памяти и шириной восстановления для обобщенных регенерирующих кодов

и было показано, что каждая точка на кривой компромисса достижима. В обобщенном регенерирующем коде множество узлов разбито на две части так, что все узлы в одной части имеют одинаковые параметры  $(\alpha_i, d_i, \beta_i)$ ,  $i = 1, 2$ . В [26] была вычислена граница пропускной способности для неоднородных РСХД с переменной шириной восстановления и постоянной степенью восстановления. В [27] эта граница была вычислена для неоднородных РСХД с переменной шириной восстановления, где восстановление узлов производится с помощью некоторых конкретных вспомогательных узлов. В [28] исследовалось компромиссное соотношение между стоимостью хранения в системе и стоимостью восстановления системы для неоднородных РСХД с переменной стоимостью хранения и восстановления. В [29] были предложены селективные регенерирующие коды, и для них получено компромиссное соотношение между объемом памяти на узел и шириной восстановления, где селективные регенерирующие коды – это регенерирующие коды, в которых вспомогательные узлы для отказавшего узла выбираются разумным способом, уменьшающим ширину восстановления. В [30, 31] анализировалось компромиссное соотношение между объемом памяти и шириной восстановления узла для задачи точного восстановления узлов. В [32] дана конструкция для внутренних точек нормированного компромиссного соотношения. В [33] улучшаются границы для регенерирующих кодов с точным восстановлением узлов. В [34] обсуждались улучшения для регенерирующих кодов с помощью разумного выбора вспомогательных узлов при отказе произвольного узла.

В настоящей статье рассматривается неоднородная РСХД, в которой файл размера  $B$  распределяется по  $n$  узлам, каждый из которых имеет различный объем памяти. Реконструкция файла ведется гибким образом, когда в любой момент времени устройство сбора данных может для реконструкции файла устанавливать соединение с  $k_t$  узлами для некоторого  $t$ . Следовательно, для заданного файла *степень реконструкции*  $k_t$  является переменной. С другой стороны, в случае отказа любого узла  $U_i$ ,  $i \in \{1, 2, \dots, n\}$ , его можно восстановить, загружая пакеты из некоторых  $d_i^{(t)}$  узлов. Следовательно, *степень восстановления*  $d_i^{(t)}$  также переменна относительно числа узлов. Восстановление отказавшего узла может выполняться двумя способами – точное восстановление и функциональное. Если пакеты, создаваемые в процессе восстановления являются точными копиями утерянных, это называется *точным восстановлением*. С другой стороны, если восстанавливаемые пакеты являются некоторыми функциями от утерянных, то восстановление называется *функциональным*. Модель такой неоднородной РСХД показана на рис. 1. Устройство сбора данных реконструирует распространяемый файл, устанавливая соединение с  $k_t$  узлами,  $t \in \{1, 2\}$ . При этом отказавший узел  $U_i$  восстанавливается с помощью некоторых  $d_i^{(t)}$  узлов.

Пример такой неоднородной РСХД показан на рис. 2. В этой системе файл  $B$  разделен на четыре информационных пакета  $x_1, x_2, x_3$  и  $x_4$ . Эти информационные пакеты закодированы одиннадцатью пакетами, являющимися их линейными комбинациями:  $y_1 = x_1, y_2 = x_2, y_3 = x_3, y_4 = x_1 + x_2, y_5 = x_4, y_6 = x_1 + x_2, y_7 = x_1, y_8 = x_3, y_9 = x_2 + x_4, y_{10} = x_2$  и  $y_{11} = x_1 + x_4$ . Закодированные пакеты  $y_m, m = 1, 2, \dots, 11$ , распределены по пяти узлам, так что пакеты  $y_1$  и  $y_2$  хранятся в узле  $U_1$ , пакеты  $y_3$  и  $y_4$  – в узле  $U_2$ , пакеты  $y_5$  и  $y_6$  – в узле  $U_3$ , пакеты  $y_7, y_8$  и  $y_9$  – в узле  $U_4$  и оставшиеся два пакета – в узле  $U_5$ . Очевидно, узлы имеют объем памяти  $\alpha_i = 2$  для  $i = 1, 2, 3, 5$  и  $\alpha_4 = 3$ . В этом примере, если отказывает узел  $U_5$ , то его можно восстановить путем загрузки пакетов  $y_7$  и  $y_9$  из узла  $U_4$ . Поскольку восстановленные пакеты являются функциями от утерянных, то это – функциональное восстановление. С другой стороны, узел  $U_5$  может быть восстановлен точно путем загрузки пакетов  $y_1, y_2$  и  $y_5$  из узлов  $U_1$  и  $U_3$  и решения  $y_{10} = y_2$  и  $y_{11} = y_1 + y_5$ .

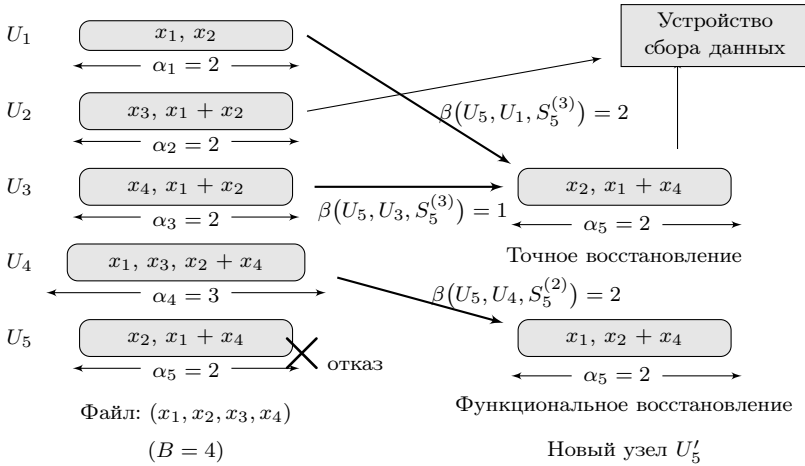


Рис. 2. Файл размером в 4 единицы данных ( $= B$ ) разбит на 11 закодированных пакетов над полем  $\mathbb{F}_q$ . Эти пакеты разделены между 5 ( $= n$ ) узлами таким образом, что любое устройство сбора данных может загрузить весь файл, установив соединение с не более чем 3 ( $= k$ ) узлами. В этой неоднородной РСХД  $\alpha = (\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5) = (2, 2, 2, 3, 2)$ . Отказавший узел может быть восстановлен с помощью не более чем 2 ( $= d$ ) узлов. Функциональное и точное восстановление показаны для отказа узла  $U_5$  с помощью множеств выживания  $S_5^{(2)} = \{U_4\}$  и  $S_5^{(3)} = \{U_1, U_3\}$ , где множество выживания – это множество, состоящее из нескольких вспомогательных узлов. Множество выживания  $S_5^{(3)}$  не представлено в табл. 1, поскольку  $S_5^{(1)} \subsetneq S_5^{(3)}$

Под РСХД с параметрами  $(n, k, d)$  будем подразумевать систему хранения с  $n$  узлами, степенью реконструкции  $k$  и степенью восстановления  $d$ , такую что

- каждый узел содержит часть информации о файле данных,
- любое устройство сбора данных может реконструировать весь файл данных, загружая пакеты из  $k$  ( $< n$ ) узлов, и
- отказавший узел восстанавливается с помощью некоторых  $d$  вспомогательных узлов.

**Сводка результатов.** В статье вычислена граница минимального разреза для рассматриваемых неоднородных РСХД. Для таких неоднородных РСХД сформулирована оптимизационная двухкритериальная задача линейного программирования с ограничениями, накладываемыми границей минимального разреза. Решения этой задачи представлены в виде кривой компромисса между стоимостью хранения в системе и стоимостью восстановления системы. В неоднородной РСХД стоимостью хранения в системе и стоимостью восстановления системы называются, соответственно, средняя стоимость хранения и средняя стоимость восстановления одной единицы данных в узле. Построены некоторые кривые компромисса и проведено их сравнение с компромиссным соотношением для неоднородных РСХД, рассмотренных в [28], и однородных РСХД, изучавшихся в [7]. Для поставленной задачи двухкритериальной оптимизации изучены некоторые специальные случаи. Вычислительная сложность нахождения параметров для кодов, достигающих фундаментальной границы, очень велика. Поэтому рассмотрен специальный случай с постоянными трафиком восстановления и степенью реконструкции. Кроме того, получено соотношение на параметры кодов, достигающих границы. Для некоторых конкретных значений параметров с помощью графов построены оптимальные коды (достигающие фундаментальной границы).

**Структура статьи.** Статья организована следующим образом. В §2 приводятся необходимые предварительные сведения и описывается рассматриваемая модель. В §3 изучается граница минимального разреза для этой модели. При ограничениях, накладываемых границей минимального разреза, формулируется задача двухкритериальной линейной оптимизации, с помощью которой строится кривая компромисса между стоимостью хранения и восстановления на узел. В §4 исследуется фундаментальная граница для конкретной неоднородной РСХД. В заключительном §5 даются общие замечания.

## § 2. Предварительные сведения

Основное внимание уделяется рассмотрению неоднородных РСХД с неодинаковыми параметрами узлов. Для произвольной неоднородной РСХД определим множество  $\mathcal{A}_t = \{U_i : i \in I \subset \{1, 2, \dots, n\}\}$ , состоящее из узлов, в которых содержится достаточно пакетов для получения всего файла. Множество  $\mathcal{A}_t$  назовем *реконструирующим множеством* и будем использовать обозначение  $|I| = |\mathcal{A}_t| = k_t$ . Для множества  $\mathcal{A} = \{\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_t, \dots\}$ , состоящего из всех различных реконструирующих множеств для произвольной неоднородной РСХД, будем всегда предполагать, что это множество  $\mathcal{A}$  конечно. Следовательно,  $\exists \omega \in \mathbb{N}$ , такое что  $|\mathcal{A}| = \omega$ . Далее, для неоднородной РСХД *степенью реконструкции*  $k_t$  называется мощность реконструирующего множества  $\mathcal{A}_t$ , т.е.  $k_t = |\mathcal{A}_t|$  и  $k = \max\{k_t : t \in \{1, 2, \dots, \omega\}\}$ . В примере, показанном на рис. 2,  $\mathcal{A} = \{\mathcal{A}_i : i = 1, 2, \dots, 7\}$ , где  $\mathcal{A}_1 = \{U_1, U_2, U_3\}$ ,  $\mathcal{A}_2 = \{U_1, U_2, U_5\}$ ,  $\mathcal{A}_3 = \{U_1, U_4\}$ ,  $\mathcal{A}_4 = \{U_2, U_4\}$ ,  $\mathcal{A}_5 = \{U_2, U_5\}$ ,  $\mathcal{A}_6 = \{U_3, U_4\}$  и  $\mathcal{A}_7 = \{U_4, U_5\}$ . Таким образом,  $(k_1, k_2, k_3, k_4, k_5, k_6, k_7) = (3, 3, 2, 2, 2, 2, 2)$ , и поэтому  $k = 3$ .

Если в неоднородной РСХД отказывает узел  $U_i, i \in \{1, 2, \dots, n\}$ , то из некоторых активных узлов, называемых вспомогательными, загружаются необходимые пакеты и образуется новый узел, скажем,  $U'_i$ . Этот новый узел  $U'_i$  занимает место отказавшего узла  $U_i$ , и таким образом система восстанавливается. В частности, множество таких вспомогательных узлов называется *множеством выживания* для отказавшего узла  $U_i$ . Заметим, что для всякого отказавшего узла может существовать более одного множества выживания. Для узла  $U_i$  обозначим число его различных множеств выживания через  $\tau_i$ . Будем обозначать множество выживания с номером  $\ell$  через  $S_i^{(\ell)} = \{U_j : j \in J \subset \{1, 2, \dots, n\} \setminus \{i\}\}$ , где  $\ell \in \{1, 2, \dots, \tau_i\}$  [27]. Если отказавший узел  $U_i$  восстанавливается с помощью узлов из множества выживания  $S_i^{(\ell)}$ , то *степень восстановления* узла  $U_i$  равна  $d_i^{(\ell)} = |J| = |S_i^{(\ell)}|$ . Для неоднородной РСХД, представленной на рис. 2, множества выживания перечислены в табл. 1. В этом примере видно, что если отказывает узел  $U_5$ , то его можно восстановить, установив соединение либо с узлами  $U_1$  и  $U_3$ , либо с  $U_1$  и  $U_4$ . Следовательно, множества выживания для узла  $U_5$  – это  $S_5^{(1)} = \{U_1, U_3\}$  и  $S_5^{(2)} = \{U_1, U_4\}$ , как и указано в табл. 1. Заметим также, что в этой таблице перечислены только те множества выживания, которые не покрывают никакое другое множество выживания для того же самого отказавшего узла. Это условие, в частности, гарантирует активное участие каждого узла из любого множества выживания в процессе восстановления.

Для всякого отказавшего узла  $U_i$ , если система восстанавливается с помощью узлов из какого-либо конкретного множества выживания  $S_i^{(\ell)}$ , то число информационных пакетов, загружаемых из узла  $U_j \in S_i^{(\ell)}$ , обозначается через  $\beta(U_i, U_j, S_i^{(\ell)}) > 0$ . Отметим, что положительность значения  $\beta(U_i, U_j, S_i^{(\ell)})$  соответствует активному участию каждого вспомогательного узла из множества выживания  $S_i^{(\ell)}$ . Например, на рис. 2 для восстановления отказавшего узла  $U_4$  загружаются все пакеты из узлов  $U_2 \in S_4^{(2)}$  и  $U_5 \in S_4^{(2)}$ , и недостающие пакеты  $y_7, y_8$  и  $y_9$  получаются из них как

Множества выживания и ширина восстановления для неоднородной РСХД, представленной на рис. 2

Узлы $U_i$	Множества выживания $S_i^{(\ell)}$	Ширина восстановления $\gamma(U_i, S_i^{(\ell)})$	Число множеств выживания $\tau_i$
$U_1$	$S_1^{(1)} = \{U_2, U_4\}$ $S_1^{(2)} = \{U_2, U_5\}$ $S_1^{(3)} = \{U_3, U_4\}$ $S_1^{(4)} = \{U_3, U_5\}$ $S_1^{(5)} = \{U_4, U_5\}$	$\gamma(U_1, S_1^{(1)}) = 2$ $\gamma(U_1, S_1^{(2)}) = 2$ $\gamma(U_1, S_1^{(3)}) = 2$ $\gamma(U_1, S_1^{(4)}) = 2$ $\gamma(U_1, S_1^{(5)}) = 2$	5
$U_2$	$S_2^{(1)} = \{U_1, U_4\}$ $S_2^{(2)} = \{U_3, U_4\}$ $S_2^{(3)} = \{U_4, U_5\}$	$\gamma(U_2, S_2^{(1)}) = 3$ $\gamma(U_2, S_2^{(2)}) = 2$ $\gamma(U_2, S_2^{(3)}) = 3$	3
$U_3$	$S_3^{(1)} = \{U_1, U_4\}$ $S_3^{(2)} = \{U_1, U_5\}$ $S_3^{(3)} = \{U_4, U_5\}$	$\gamma(U_3, S_3^{(1)}) = 3$ $\gamma(U_3, S_3^{(2)}) = 3$ $\gamma(U_3, S_3^{(3)}) = 3$	3
$U_4$	$S_4^{(1)} = \{U_1, U_2, U_3\}$ $S_4^{(2)} = \{U_2, U_5\}$	$\gamma(U_4, S_4^{(1)}) = 4$ $\gamma(U_4, S_4^{(2)}) = 4$	2
$U_5$	$S_5^{(1)} = \{U_1, U_3\}$ $S_5^{(2)} = \{U_1, U_4\}$	$\gamma(U_5, S_5^{(1)}) = 3$ $\gamma(U_5, S_5^{(2)}) = 3$	2

$y_7 = y_4 - y_{10}$ ,  $y_8 = y_3$  и  $y_9 = y_{11} + 2y_{10} - y_4$ . Следовательно,  $\beta(U_4, U_5, S_4^{(2)}) = 2$  и  $\beta(U_4, U_2, S_4^{(2)}) = 2$ .

Далее, дадим следующие формальные определения. Для отказавшего узла  $U_i$  и множества выживания  $S_i^{(\ell)}$  введем вектор

$$\beta_i^{(\ell)} = (\beta(U_i, U_{j_1}, S_i^{(\ell)}), \beta(U_i, U_{j_2}, S_i^{(\ell)}), \dots, \beta(U_i, U_{j_m}, S_i^{(\ell)}))$$

числа пакетов, которые загружаются из вспомогательных узлов множества выживания  $S_i^{(\ell)} = \{U_{j_1}, U_{j_2}, \dots, U_{j_m}\}$  мощности  $m$ , т.е.  $m = d_i^{(\ell)} = |S_i^{(\ell)}|$ . Для отказавшего узла  $U_i$  введем вектор  $\beta_i = (\beta_i^{(1)}, \beta_i^{(2)}, \dots, \beta_i^{(\tau_i)})$  из всех  $\beta_i^{(\ell)}$ , т.е.  $\beta = (\beta_1, \beta_2, \dots, \beta_n)$ .

Если отказавший узел  $U_i$ ,  $i \in \{1, 2, \dots, n\}$ , восстанавливается с помощью узлов из множества выживания  $S_i^{(\ell)}$ , то *шириной восстановления* для узла  $U_i$  (обозначаемой через  $\gamma(U_i, S_i^{(\ell)})$ ) называется общее число пакетов, загружаемых всеми узлами из множества выживания  $S_i^{(\ell)}$ , т.е.

$$\gamma(U_i, S_i^{(\ell)}) = \sum_{j: U_j \in S_i^{(\ell)}} \beta(U_i, U_j, S_i^{(\ell)}).$$

Для примера из рис. 2, если узел  $U_4$  отказывает и затем восстанавливается с помощью узлов из множества выживания  $S_4^{(2)} = \{U_2, U_5\}$ , то ширина восстановления для узла  $U_4$  равна  $\gamma(U_4, S_4^{(2)}) = \beta(U_4, U_2, S_4^{(2)}) + \beta(U_4, U_5, S_4^{(2)}) = 2 + 2 = 4$  единицы.

*Замечание 1.* Мы рассматриваем только отказы единичных узлов, поскольку случай одновременного отказа нескольких узлов можно рассматривать как последовательность единичных отказов.

Далее, неоднородную РСХД можно формально определить следующим образом.

Определение 1. Неоднородной РСХД с параметрами  $(n, \mathbf{k}, \mathbf{d}, \boldsymbol{\alpha}, \boldsymbol{\beta}, B)$  называется система хранения с  $n$  узлами, такая что

- в каждом узле  $U_i$  ( $i = 1, 2, \dots, n$ ) содержится часть информации о файле, закодированная в виде данных объема  $\alpha_i$ ,
- любое устройство сбора данных может полностью реконструировать файл путем загрузки пакетов из некоторых  $k_t$  узлов ( $k_t$  – мощность реконструирующего множества  $\mathcal{A}_t$  с номером  $t \in \{1, 2, \dots, \omega\}$ ), и
- отказавший узел  $U_i$  восстанавливается с помощью загрузки  $\sum_{U_j \in S_i^{(\ell)}} \beta(U_i, U_j, S_i^{(\ell)})$

пакетов из вспомогательных узлов множества выживания  $S_i^{(\ell)}$ ,  $\ell \in \{1, 2, \dots, \tau_i\}$ , где

$$\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_n),$$

$$\mathbf{d} = (\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n), \quad \mathbf{d}_i = (|S_i^{(1)}|, |S_i^{(2)}|, \dots, |S_i^{(\tau_i)}|),$$

$$\boldsymbol{\beta}_i^{(\ell)} = (\beta(U_i, U_{j_1}, S_i^{(\ell)}), \beta(U_i, U_{j_2}, S_i^{(\ell)}), \dots, \beta(U_i, U_{j_m}, S_i^{(\ell)})), \quad m = |S_i^{(\ell)}|,$$

$$\boldsymbol{\beta}_i = (\boldsymbol{\beta}_i^{(1)}, \boldsymbol{\beta}_i^{(2)}, \dots, \boldsymbol{\beta}_i^{(\tau_i)}), \quad \boldsymbol{\beta} = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \dots, \boldsymbol{\beta}_n).$$

Чтобы построить кривую компромисса между объемом памяти и шириной восстановления для любой однородной РСХД, в [7] решалась задача оптимизации с ограничениями, накладываемыми границей минимального разреза между параметрами. Эта граница была вычислена путем анализа графа информационных потоков в данной однородной РСХД [7]. Аналогичным способом можно вывести границу минимального разреза и построить кривую компромисса для любой неоднородной РСХД. В настоящей статье граница минимального разреза выводится с помощью графа информационных потоков, как описано в [27, 28], где граф информационных потоков – это взвешенный ориентированный ациклический граф  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  с множеством вершин  $\mathcal{V}$  и множеством ребер  $\mathcal{E}$ .

Для произвольной неоднородной РСХД с заданным реконструирующим множеством  $\mathcal{A}_t = \{U_{\lambda_1}, U_{\lambda_2}, \dots, U_{\lambda_{k_t}}\}$  ее граф информационных потоков  $\mathcal{G}$  показан на рис. 3, где рассматриваются отказы всех узлов  $U_{\lambda_1}, U_{\lambda_2}, \dots, U_{\lambda_{k_t}}$  одного за другим в указанном порядке и их восстановление с помощью множеств выживания  $S_{\lambda_1}^{(\ell_{\lambda_1})}, S_{\lambda_2}^{(\ell_{\lambda_2})}, \dots, S_{\lambda_{k_t}}^{(\ell_{\lambda_{k_t}})}$ , где  $\ell_{\lambda_i} \in \{1, 2, \dots, \tau_{\lambda_i}\}$ . Для любого реконструирующего множества  $\mathcal{A}_t$  граф делится на  $k_t + 3$  уровней, начиная с уровня  $-1$  и заканчивая уровнем  $k_t + 1$ . На уровне  $-1$  в графе содержится узел-источник “ $s$ ”, а на уровне  $k_t + 1$  – узел (устройство) сбора данных “ $D$ ”. Типичный узел  $U_{\lambda_i}$  неоднородной РСХД,  $i = 1, 2, \dots, n$ , представлен в графе в виде пары вершин “ $\text{In}_{\lambda_i}$ ” и “ $\text{Out}_{\lambda_i}$ ” из  $\mathcal{V}$ , соединенных ребром, т.е.  $(\text{In}_{\lambda_i}, \text{Out}_{\lambda_i}) \in \mathcal{E}$ , где  $\lambda_i$  – упорядоченные с помощью некоторой перестановки номера узлов. Объем памяти  $\alpha_{\lambda_i}$  узла  $U_{\lambda_i}$  представлен в виде веса  $w(\text{In}_{\lambda_i}, \text{Out}_{\lambda_i})$  ребра  $(\text{In}_{\lambda_i}, \text{Out}_{\lambda_i}) \in \mathcal{E}$ . В графе  $\mathcal{G}$ , представленном на рис. 3, на уровне 0 содержится  $2n$  вершин, обозначенных  $\text{In}_{\lambda_i}$  и  $\text{Out}_{\lambda_i}$ , ассоциированных с узлами  $U_{\lambda_i}$ ,  $i \in \{1, 2, \dots, n\}$ .

В такой неоднородной РСХД отказавший узел  $U_{\lambda_i}$ ,  $i \in \{1, 2, \dots, n\}$ , восстанавливается путем образования нового узла  $U'_{\lambda_i}$ . В графе информационных потоков узел  $U'_{\lambda_i}$  представляется новой парой узлов  $\text{In}'_{\lambda_i}$  и  $\text{Out}'_{\lambda_i}$ ,  $(\text{In}'_{\lambda_i}, \text{Out}'_{\lambda_i}) \in \mathcal{E}$ , такой что  $w(\text{In}'_{\lambda_i}, \text{Out}'_{\lambda_i}) = \alpha_{\lambda_i}$ . На каждом уровне  $j$ ,  $j = 1, 2, \dots, k_t$ , в графе потоков содержится одна пара узлов  $\text{In}'_{\lambda_j}$  и  $\text{Out}'_{\lambda_j}$ . Как показано на рис. 3, в неоднородной РСХД при отказе узла  $U_{\lambda_j}$  система восстанавливается с помощью загрузки данных объема  $\beta(U_{\lambda_1}, U_{\mu_j}, S_{\lambda_1}^{(\ell_{\lambda_1})})$  из узла  $U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} = \{U_{\mu_j} : j = 1, 2, \dots, d_{\lambda_i}^{(\ell_{\lambda_i})}\}$ , где  $\mu_j$  – некоторая перестановка номеров узлов. Для каждого отдельного восста-





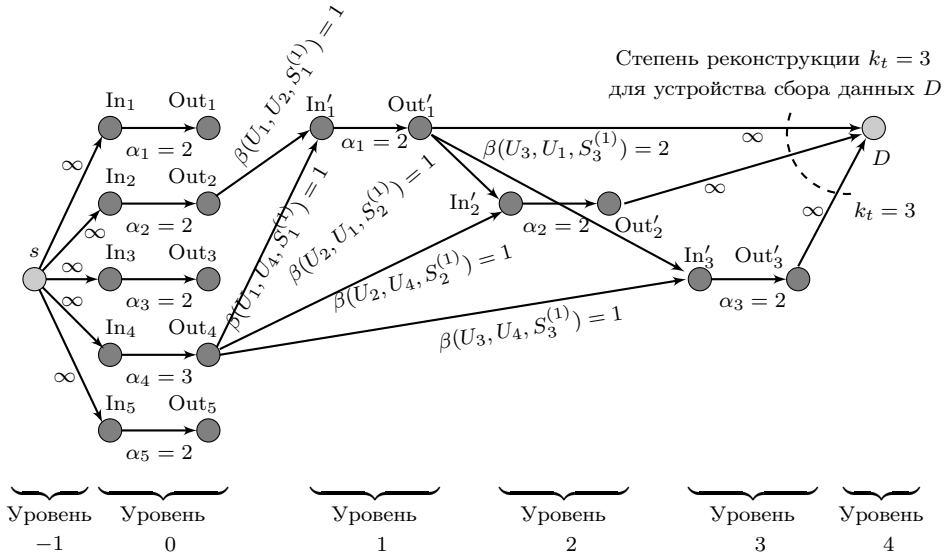


Рис. 4. Граф информационных потоков для неоднородной РСХД, представленной на рис. 2, для конкретного устройства сбора данных, устанавливающего соединение с узлами из реконструирующего множества  $\mathcal{A}_1 = \{U_1, U_2, U_3\}$ . Этот конкретный граф информационных потоков построен для последовательности множеств выживания  $\langle S_1^{(1)}, S_2^{(1)}, S_3^{(1)} \rangle$  (см. определение 4 для произвольной неоднородной РСХД)

В работах [7, 26–28] граница минимального разреза вычисляется с помощью анализа потоков, идущих по графу информационных потоков от узла-источника  $s$  до устройства сбора данных  $D$ , для любой РСХД. В настоящей статье выполняется аналогичный анализ информационных потоков для неоднородной РСХД. Поэтому определим поток на графе информационных потоков следующим образом.

**Определение 2** (информационный поток). Функция  $f: \mathcal{E} \rightarrow [0, \infty) \subset \mathbb{R}$  называется информационным потоком, или просто потоком, на графе информационных потоков  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , если выполнены следующие условия:

1. (ограничение по пропускной способности): для каждого ребра  $(x, y) \in \mathcal{E}$  имеет место неравенство  $f((x, y)) \leq c((x, y))$ , где  $c((x, y)) = w(x, y)$  и  $c((x, y))$  – пропускная способность ребра  $(x, y)$ , и
2. (ограничение по сохранению потока): для каждой вершины  $y \in \mathcal{V} \setminus \{s, t\}$  справедливо равенство

$$\sum_{x: (x,y) \in \mathcal{E}} f((x,y)) = \sum_{z: (y,z) \in \mathcal{E}} f((y,z)).$$

Дальнейшие подробности и примеры таких функций можно найти в [35, 36].

Для заданного графа информационных потоков  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  значение потока, поступающего на устройство сбора данных  $D$ , определяется как общая сумма потоков, идущих по ребрам  $(x, D) \in \mathcal{E}$  для всех возможных  $x \in \mathcal{V}$ . Для сетей максимальное возможное значение потока, приходящего в  $D$ , описывается теоремой о минимальном разрезе и максимальном потоке [35–37]. Согласно этой теореме максимальное возможное значение  $\text{max-flow}(s, D)$  потока, идущего по сети от источника  $s$  до конкретного устройства сбора данных  $D$ , равно минимуму величины  $\text{cut-capacity}(s, D)$ ,

где

$$\min \text{cut-capacity}(s, D) = \min_{\substack{\text{cut}(\mathcal{X}, \overline{\mathcal{X}}) \\ s \in \mathcal{X}, D \in \overline{\mathcal{X}} \\ \overline{\mathcal{X}} = \mathcal{V} \setminus \mathcal{X}}} \text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}}).$$

Заметим, что через  $\text{cut}(\mathcal{X}, \overline{\mathcal{X}})$  обозначено множество всех ребер, имеющих один конец в множестве  $\mathcal{X}$ , а другой в множестве  $\overline{\mathcal{X}}$ , таких что при удалении всех этих ребер число компонент графа  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  увеличится. Здесь  $\text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}})$  – сумма пропускных способностей всех ребер из множества  $\text{cut}(\mathcal{X}, \overline{\mathcal{X}})$ . Для конкретного устройства сбора данных  $D$ , соединяющегося с узлами  $U_{\lambda_i}$  из множества  $\mathcal{A}_t \in \mathcal{A}$ , имеется  $|\mathcal{A}_t|! \prod_{i=1}^{|\mathcal{A}_t|} \tau_{\lambda_i}$  различных графов информационных потоков. Для каждого графа информационных потоков  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  узел  $D$  может получить весь файл  $B$ , поэтому

$$B \leq \min_{\mathcal{G}} \text{max-flow}(s, D).$$

По теореме о минимальном разрезе и максимальном потоке суммарный поток через разрез равен пропускной способности разреза. Таким образом,

$$B \leq \min_{\mathcal{G}} \min \text{cut-capacity}(s, D). \quad (1)$$

В [7] анализ потоков для графа информационных потоков был произведен с помощью выбора топологического порядка на отказывающих узлах, с которыми также соединяется устройство сбора данных для получения всего файла. В настоящей статье для анализа потоков мы определяем некоторые последовательности узлов и соответствующие множества выживания для нашей модели. Приведем эти определения.

**Определение 3** (множество последовательностей узлов). Множество всевозможных последовательностей узлов из реконструирующего множества  $\mathcal{A}_t \in \mathcal{A}$  называется множеством последовательностей узлов и обозначается через  $\mathcal{A}(\mathcal{A}_t) = \{ \langle U_{\lambda_i} \rangle_{i=1}^{|\mathcal{A}_t|} : U_{\lambda_i} \in \mathcal{A}_t \}$ , где  $\langle U_{\lambda_i} \rangle_{i=1}^{|\mathcal{A}_t|}$  – последовательность различных узлов множества  $\mathcal{A}_t \in \mathcal{A}$ . Очевидно,  $|\mathcal{A}(\mathcal{A}_t)| = |\mathcal{A}_t|!$ .

Например, на рис. 2 имеем  $\mathcal{A}(\mathcal{A}_3) = \{ \langle U_1, U_4 \rangle, \langle U_4, U_1 \rangle \}$ , где  $\mathcal{A}_3 = \{ U_1, U_4 \}$ .

**Определение 4** (последовательность выживания). Для любого реконструирующего множества  $\mathcal{A}_t \in \mathcal{A}$  можно определить последовательность множеств выживания  $S_{\lambda_i}^{(\ell)}$ ,  $i = 1, 2, \dots, |\mathcal{A}_t|$ ,  $\ell = 1, 2, \dots, \tau_{\lambda_i}$ , такую что  $U_{\lambda_i} \in \mathcal{A}_t$ . Последовательность выживания, соответствующую последовательности узлов  $\langle U_{\lambda_i} \rangle_{i=1}^{|\mathcal{A}_t|} \in \mathcal{A}(\mathcal{A}_t)$ , можно обозначать через  $\langle S_{\lambda_i}^{(\ell)} \rangle_{i=1}^{|\mathcal{A}_t|}$ .

На рис. 2 последовательность  $\langle S_1^{(3)}, S_4^{(2)} \rangle$  – одна из возможных последовательностей выживания среди десяти имеющихся для последовательности узлов  $\langle U_1, U_4 \rangle$ .

**Определение 5** (множество последовательностей выживания). Множество всех последовательностей выживания, соответствующих последовательности узлов  $\langle U_{\lambda_i} \rangle_{i=1}^{|\mathcal{A}_t|}$ , определяется следующим образом:

$$\mathcal{S}(\langle U_{\lambda_i} \rangle_{i=1}^{|\mathcal{A}_t|}) = \left\{ \langle S_{\lambda_i}^{(\ell)} \rangle_{i=1}^{|\mathcal{A}_t|} : \ell \in \{1, 2, \dots, \tau_{\lambda_i}\} \right\}.$$

Очевидно,  $|\mathcal{S}(\langle U_{\lambda_i} \rangle_{i=1}^{|\mathcal{A}_t|})| = \left( \prod_{i=1}^{|\mathcal{A}_t|} \tau_{\lambda_i} \right)!$ .

Например, на рис. 2 имеем  $\{\langle S_1^{(\ell_1)}, S_4^{(\ell_2)} \rangle : \ell_1 = 1, 2, 3, 4, 5 \text{ и } \ell_2 = 1, 2\} = \mathcal{S}(\langle U_1, U_4 \rangle)$  и т.д.

В [28] была получена кривая компромисса между стоимостью хранения в системе и стоимостью восстановления системы для произвольной неоднородной РСХД с одинаковыми степенями реконструкции. Аналогично можно получить кривую компромисса между стоимостью хранения в системе и стоимостью восстановления системы для модели неоднородной РСХД, рассматриваемой в настоящей статье. Определим для нашей модели понятия стоимости хранения в системе, стоимости восстановления узла и стоимости восстановления системы следующим образом.

**Определение 6** (стоимость хранения в системе). Общая величина стоимости хранения единицы данных в неоднородной РСХД с параметрами  $(n, \mathbf{k}, \mathbf{d}, \boldsymbol{\alpha}, \boldsymbol{\beta}, B)$  называется стоимостью хранения в системе  $C_s(\boldsymbol{\alpha})$ , где  $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_n)$  – вектор объема памяти,  $\mathbf{s} = (s_1, s_2, \dots, s_n)$  – вектор стоимости хранения,  $\alpha_i$  – объем памяти узла  $U_i$ , а  $s_i$  – стоимость хранения единицы данных в узле  $U_i$ ,  $i = 1, 2, \dots, n$ . Очевидно,

$$C_s(\boldsymbol{\alpha}) = \frac{1}{B} \sum_{j=1}^n s_j \alpha_j.$$

**Определение 7** (стоимость восстановления узла). Средняя величина стоимости восстановления узла  $U_i$ ,  $i \in \{1, 2, \dots, n\}$ , в неоднородной РСХД с параметрами  $(n, \mathbf{k}, \mathbf{d}, \boldsymbol{\alpha}, \boldsymbol{\beta}, B)$  называется стоимостью восстановления узла  $r(\beta_i)$ , соответствующей вектору стоимости восстановления  $\mathbf{r} = (r_1, r_2, \dots, r_n)$ , т.е.

$$r(\beta_i) = \frac{1}{B\tau_i} \sum_{\ell=1}^{\tau_i} \sum_{j: U_j \in S_i^{(\ell)}} r_j \beta(U_i, U_j, S_i^{(\ell)}),$$

где  $r_j$  – стоимость загрузки единицы данных из узла  $U_j$  в процессе восстановления. Вектором восстановления узлов называется  $r(\boldsymbol{\beta}) = (r(\beta_1), r(\beta_2), \dots, r(\beta_n))$ .

**Определение 8** (стоимость восстановления системы). Стоимость восстановления системы  $C_r(\boldsymbol{\beta})$  – это общая величина стоимости восстановления всех узлов в неоднородной РСХД с параметрами  $(n, \mathbf{k}, \mathbf{d}, \boldsymbol{\alpha}, \boldsymbol{\beta}, B)$ . На математическом языке

$$C_r(\boldsymbol{\beta}) = \sum_{j=1}^n r(\beta_j).$$

В следующем параграфе мы приводим некоторые результаты и анализ для границы минимального разреза для модели неоднородной РСХД, рассматриваемой в настоящей статье.

### § 3. Результаты

Мы показываем, что для нашей модели минимальным возможным значением переменной степени реконструкции является нижняя граница мощности любого множества разреза, разделяющего узел-источник и узел сбора данных. Для неоднородной РСХД граница минимального разреза вычислена в теореме 1. С использованием этой границы минимального разреза мы показываем, что для неоднородной РСХД размер файла должен быть меньше границы минимального разреза. Используя эту конкретную границу в качестве ограничения, мы формулируем оптимизационную задачу двухкритериального линейного программирования для минимизации стоимости хранения в системе и стоимости восстановления системы для рассматриваемой неоднородной модели. Для этой задачи мы вычисляем семейство решений,

подставляя некоторые числовые значения параметров системы. По числовому параметру строится кривая компромисса между стоимостью хранения в системе и стоимостью восстановления системы. Эта кривая сравнивается с кривой компромисса для однородной РСХД [7] и кривой компромисса для неоднородной РСХД из [28].

В неоднородной РСХД информация, доставляемая на устройство сбора данных  $D$ , зависит от минимума величины cut-capacity( $s, D$ ). В теореме 1 дается нижняя граница на  $\min \text{cut-capacity}(s, D)$ .

**Теорема 1** (граница минимального разреза). *Для заданной неоднородной РСХД с устройством сбора данных  $D$  и переменной степенью реконструкции  $k_t$  справедливо неравенство  $\min \text{cut-capacity}(s, D) \geq Q$ , где*

$$Q = \min_{\mathcal{A}_t \in \mathcal{A}} \min_{\langle U_{\lambda_i} \rangle_{i=1}^{k_t}} \sum_{\substack{i=1 \\ U_{\lambda_i} \in \mathcal{A}_t}}^{k_t} \min \left\{ \alpha_{\lambda_i}, \min_{\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}} \sum_{\mu_j} \beta(U_{\lambda_i}, U_{\mu_j}, S_{\lambda_i}^{(\ell_{\lambda_i})}) \right\}, \quad (2)$$

причем  $|\mathcal{A}_t| = k_t$ ,  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t} \in \mathcal{A}(\mathcal{A}_t)$ ,  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t} \in \mathcal{S}(\langle U_{\lambda_i} \rangle_{i=1}^{k_t})$ ,  $\ell_i \in \{1, 2, \dots, \tau_i\}$ , а индекс  $\mu_j$  соответствует узлу  $U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, \dots, U_{\lambda_{i-1}}\}$ . Остальные обозначения в этом выражении имеют обычный смысл, описанный ранее.

**Доказательство.** Рассмотрим неоднородную РСХД с параметрами  $(n, \mathbf{k}, \mathbf{d}, \alpha, \beta, B)$  и заданным реконструирующим множеством  $\mathcal{A}_t = \{U_{\lambda_1}, U_{\lambda_2}, \dots, U_{\lambda_{k_t}}\}$ . Если все узлы  $U_{\lambda_1}, U_{\lambda_2}, \dots, U_{\lambda_{k_t}}$  отказывают один за другим в порядке, заданном фиксированной последовательностью  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t}$ , то для  $\ell_i \in \{1, 2, \dots, \tau_i\}$  эти узлы восстанавливаются с помощью множеств выживания  $S_{\lambda_1}^{(\ell_{\lambda_1})}, S_{\lambda_2}^{(\ell_{\lambda_2})}, \dots, S_{\lambda_{k_t}}^{(\ell_{\lambda_{k_t}})}$ . Таким образом, процесс восстановления идет в том же порядке, т.е. в порядке множеств выживания, заданном последовательностью выживания  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}$ . Для неоднородной РСХД рассмотрим граф информационных потоков  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ . В любом графе информационных потоков  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  имеется узел-источник  $s$ , узел сбора данных  $D$ , причем устройство сбора данных  $D$  устанавливает соединение со всеми узлами из реконструирующего множества  $\mathcal{A}_t$ , и поэтому степень реконструкции равна  $k_t$ . В этой неоднородной РСХД отказавший узел  $U_i$  может быть восстановлен с узлов из некоторого множества выживания  $S_i^{(\ell_i)}$ , где  $\ell_i \in \{1, 2, \dots, \tau_i\}$ .

Пусть  $\mathcal{X} \subset \mathcal{V}$ ,  $\overline{\mathcal{X}} = \mathcal{V} \setminus \mathcal{X}$ ,  $s \in \mathcal{X}$  и  $D \in \overline{\mathcal{X}}$ , так что существует некоторое непустое подмножество  $\text{cut}(\mathcal{X}, \overline{\mathcal{X}}) \subset \mathcal{E}$ . Далее, если  $\mathcal{X} = \mathcal{V} \setminus \{D\}$ , то  $\text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}}) \rightarrow \infty$ . Аналогично, если  $\mathcal{X} = \{s\}$ , то опять же  $\text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}}) \rightarrow \infty$ . Следовательно,  $\min \text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}})$  будет получен при всех  $\text{Out}'_j \in \overline{\mathcal{X}}$  и  $\text{In}_i \in \mathcal{X}$ , поскольку это даст конечное значение величины  $\text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}})$ , где  $i \in \{1, 2, \dots, n\}$  и  $j \in \{1, 2, \dots, k_t\}$ .

Граф информационных потоков  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  является ориентированным ациклическим графом, поэтому его можно представить в топологическом порядке его вершин. Для этого топологического порядка последовательности отказов узла и соответствующие последовательности множеств выживания должны быть организованы согласно определениям, данным в предыдущих параграфах. С этой целью предположим, что устройство сбора данных  $D$  устанавливает соединение со всеми узлами множества  $\mathcal{A}_t \in \mathcal{A}$  и реконструирует файл  $B$ . Множество  $\mathcal{A}(\mathcal{A}_t)$  представляет собой набор всевозможных последовательностей узлов из  $\mathcal{A}_t \in \mathcal{A}$ . Последовательность  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t} \in \mathcal{A}(\mathcal{A}_t)$  описывает порядок отказа узлов конкретного множества  $\mathcal{A}_t$ . Напомним, что множество всевозможных последовательностей выживания  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}$ , соответствующих последовательности узлов  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t}$ , обозначается через  $\mathcal{S}(\langle U_{\lambda_i} \rangle_{i=1}^{k_t})$ .

Для заданной последовательности узлов  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t}$  с заданной последовательностью выживания  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}$  проведем следующий анализ.

Для вершины  $\text{Out}'_{\lambda_1} \in \overline{\mathcal{X}}$ , соответствующей первому узлу в последовательности узлов  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t}$ , возможны следующие два случая.

- Если  $\text{In}'_{\lambda_1} \in \mathcal{X}$ , то ребро  $(\text{In}'_{\lambda_1}, \text{Out}'_{\lambda_1}) \in \text{cut}(\mathcal{X}, \overline{\mathcal{X}})$ . Следовательно, вклад в величину  $\text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}})$  равен  $\alpha_{\lambda_1}$ .
- Если  $\text{In}'_{\lambda_1} \in \overline{\mathcal{X}}$ , то ребра  $(\text{Out}_{\mu_j}, \text{In}'_{\lambda_1}) \in \text{cut}(\mathcal{X}, \overline{\mathcal{X}})$ , где  $U_{\mu_j} \in S_{\lambda_1}^{(\ell_{\lambda_1})}$  и  $S_{\lambda_1}^{(\ell_{\lambda_1})} \in \langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}$  для любого  $\ell_{\lambda_1} \in \{1, 2, \dots, \tau_{\lambda_1}\}$ . Следовательно, в этом случае вклад в величину  $\text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}})$  равен

$$\sum_{\mu_j: U_{\mu_j} \in S_{\lambda_1}^{(\ell_{\lambda_1})}} \beta(U_{\lambda_1}, U_{\mu_j}, S_{\lambda_1}^{(\ell_{\lambda_1})}).$$

Итак, вклад узла  $U_{\lambda_1}$  в  $\min \text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}})$  равен

$$\min \left\{ \alpha_{\lambda_1}, \sum_{\mu_j: U_{\mu_j} \in S_{\lambda_1}^{(\ell_{\lambda_1})}} \beta(U_{\lambda_1}, U_{\mu_j}, S_{\lambda_1}^{(\ell_{\lambda_1})}) \right\}.$$

Если узел  $U_p$  ( $p = 1, 2, \dots, n$ ) отказывает в системе, то все узлы некоторого множества выживания  $S_p^{(\ell_p)}$  породят новый узел  $U'_p$  с теми же характеристиками.

Далее в доказательстве будем использовать обозначение  $U_p$ ,  $p \in \{1, 2, \dots, n\}$ , вместо  $U'_p$ , поскольку характеристики обоих узлов  $U_p$  и  $U'_p$  одинаковы, и в каждый момент времени существует один из них.

Вычислим вклад узла  $U_{\lambda_i} \in \langle U_{\lambda_i} \rangle_{i=1}^{k_t}$  в величину  $\min \text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}})$  в общем случае. Предположим, что  $\text{Out}'_{\lambda_i} \in \overline{\mathcal{X}}$ . Далее возможны следующие два случая.

- Если  $\text{In}'_{\lambda_i} \in \mathcal{X}$ , то ребро  $(\text{In}'_{\lambda_i}, \text{Out}'_{\lambda_i}) \in \text{cut}(\mathcal{X}, \overline{\mathcal{X}})$ . Следовательно, вклад в величину  $\text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}})$  равен  $\alpha_{\lambda_i}$ .
- Если  $\text{In}'_{\lambda_i} \in \overline{\mathcal{X}}$ , то все возможные ребра  $(\text{Out}_{\mu_j}, \text{In}'_{\lambda_i})$ , такие что  $U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, U_{\lambda_2}, \dots, U_{\lambda_{i-1}}\}$  и  $S_{\lambda_i}^{(\ell_{\lambda_i})} \in \langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}$  для любого  $\ell_{\lambda_i} \in \{1, 2, \dots, \tau_{\lambda_i}\}$ , входят в множество  $\text{cut}(\mathcal{X}, \overline{\mathcal{X}})$ . Ребра  $(\text{Out}_{\lambda_j}, \text{In}'_{\lambda_i})$ , соответствующие узлам  $U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, U_{\lambda_2}, \dots, U_{\lambda_{i-1}}\}$ , впервые входят с уровня 0 в  $\text{cut}(\mathcal{X}, \overline{\mathcal{X}})$ . Ребра  $(\text{Out}'_{\lambda_m}, \text{In}'_{\lambda_i})$  следует исключить, так как они рассматривались ранее на уровне  $m$ , где  $m \in \{1, 2, \dots, i-1\}$ , таком что  $U_{\lambda_m} \in S_{\lambda_i}^{(\ell_{\lambda_i})}$ . Следовательно, в этом случае вклад в величину  $\text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}})$  равен

$$\sum_{\mu_j: U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, U_{\lambda_2}, \dots, U_{\lambda_{i-1}}\}} \beta(U_{\lambda_i}, U_{\mu_j}, S_{\lambda_i}^{(\ell_{\lambda_i})}).$$

Итак, вклад узла  $U_{\lambda_i}$  в величину  $\min \text{cut-capacity}(\mathcal{X}, \overline{\mathcal{X}})$  равен

$$\min \left\{ \alpha_{\lambda_i}, \sum_{\mu_j: U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, U_{\lambda_2}, \dots, U_{\lambda_{i-1}}\}} \beta(U_{\lambda_i}, U_{\mu_j}, S_{\lambda_i}^{(\ell_{\lambda_i})}) \right\}.$$

Если устройство сбора данных  $D$  устанавливает соединение с каждым из узлов  $U_{\lambda_i} \in \mathcal{A}_t$ ,  $i \in \{1, 2, \dots, k_t\}$ , то для заданной последовательности узлов  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t}$ ,

соответствующей заданной последовательности выживания  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}$ , вклад в величину  $\min \text{cut-capacity}(\mathcal{X}, \bar{\mathcal{X}})$  равен

$$\sum_{i=1}^{k_t} \min \left\{ \alpha_{\lambda_i}, \sum_{\mu_j: U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, U_{\lambda_2}, \dots, U_{\lambda_{i-1}}\}} \beta(U_{\lambda_i}, U_{\mu_j}, S_{\lambda_i}^{(\ell_{\lambda_i})}) \right\}.$$

Далее, для заданного реконструирующего множества  $\mathcal{A}_t$  можно найти значение  $\min \text{cut-capacity}(s, D)$  для фиксированного  $D$ , вычисляя минимум по всевозможным значениям величины  $\text{cut-capacity}(\mathcal{X}, \bar{\mathcal{X}})$ , вычисленным для всех возможных последовательностей узлов  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t}$  во всех возможных соответствующих последовательностях выживания  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}$ , т.е.

$$\min_{\langle U_{\lambda_i} \rangle_{i=1}^{k_t}} \min_{\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}} \sum_{U_{\lambda_i} \in \mathcal{A}_t} \min \left\{ \alpha_{\lambda_i}, \sum_{\mu_j: U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, \dots, U_{\lambda_{i-1}}\}} \beta(U_{\lambda_i}, U_{\mu_j}, S_{\lambda_i}^{(\ell_{\lambda_i})}) \right\}. \quad (3)$$

Однако индекс  $\lambda_i$  объема памяти узла определяется узлами в последовательности узлов. Поэтому величина, указанная в (3), равна

$$\min_{\langle U_{\lambda_i} \rangle_{i=1}^{k_t}} \sum_{U_{\lambda_i} \in \mathcal{A}_t} \min \left\{ \alpha_{\lambda_i}, \min_{\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}} \sum_{\mu_j: U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, \dots, U_{\lambda_{i-1}}\}} \beta(U_{\lambda_i}, U_{\mu_j}, S_{\lambda_i}^{(\ell_{\lambda_i})}) \right\}.$$

Следовательно, для данной неоднородной РСХД величина  $\text{cut-capacity}(\mathcal{X}, \bar{\mathcal{X}})$  равна

$$\min_{\mathcal{A}_t \in \mathcal{A}} \min_{\langle U_{\lambda_i} \rangle_{i=1}^{k_t}} \sum_{U_{\lambda_i} \in \mathcal{A}_t} \min \left\{ \alpha_{\lambda_i}, \min_{\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}} \sum_{\mu_j: U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, \dots, U_{\lambda_{i-1}}\}} \beta(U_{\lambda_i}, U_{\mu_j}, S_{\lambda_i}^{(\ell_{\lambda_i})}) \right\},$$

где  $|\mathcal{A}_t| = k_t$ ,  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t} \in \mathcal{A}(\mathcal{A}_t)$  и  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t} \in \mathcal{S}(\langle U_{\lambda_i} \rangle_{i=1}^{k_t})$ .

Эта граница точна, поскольку граница минимального разреза вычисляется как минимум по всем возможным границам разреза. Иными словами, граница минимального разреза вычисляется для всех возможных реконструирующих множеств со всеми возможными последовательностями узлов  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t}$ , соответствующими всем возможным последовательностям выживания  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}$ . Следовательно, существует хотя бы одна последовательность выживания,  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}$ , соответствующая последовательности узлов, скажем,  $\langle U_{\lambda_i}^* \rangle_{i=1}^{k_t}$ , для которой неравенство обращается в равенство, т.е. граница минимального разреза в теореме 1 является точной.  $\blacktriangle$

*Замечание 2.* Для заданной неоднородной РСХД, если произвольное устройство сбора данных соединяется с каждым узлом  $U_{\lambda_j}$  из подмножества  $\mathcal{A}_t \in \mathcal{A}$ , то общее число возможных графов информационных потоков равно

$$\sum_{\mathcal{A}_t \in \mathcal{A}} \left( |\mathcal{A}_t|! \prod_{j=1}^{|\mathcal{A}_t|} \tau_{\lambda_j} \right).$$

В частности, для конкретного графа информационных потоков общее число вычислительных сравнений  $|\mathcal{A}_t|$ . Поэтому общее число вычислительных сравнений для

неоднородной РСХД равно

$$\sum_{\mathcal{A}_t \in \mathcal{A}} \left( |\mathcal{A}_t| (|\mathcal{A}_t|!) \prod_{j=1}^{|\mathcal{A}_t|} \tau_{\lambda_j} \right).$$

Следовательно, можно сказать, что временная сложность вычисления границы минимального разреза для неоднородной РСХД равна

$$O \left( |\mathcal{A}_t|! \prod_{j=1}^{|\mathcal{A}_t|} \tau_{\lambda_j} \right).$$

С помощью теоремы 1 можно вычислить минимальные объем памяти узла и ширину восстановления, необходимые для хранения файла размера  $B$ . Другими словами, верхнюю границу для хранения файла размера  $B$  дает следующая

*Лемма 1. Если файл размера  $B$  хранится в некоторой неоднородной РСХД с параметрами  $(n, \mathbf{k}, \mathbf{d}, \boldsymbol{\alpha}, \boldsymbol{\beta}, B)$ , то*

$$B \leq \mathcal{Q}, \tag{4}$$

где величина  $\mathcal{Q}$  определена в (2), а остальные используемые обозначения имеют обычный смысл, описанный ранее.

*Доказательство.* Любой узел сбора данных  $D$  должен быть способен реконструировать весь файл размера  $B$ . Следовательно, максимальное значение информационного потока, приходящего в любой узел сбора данных, должно быть не меньше  $B$ . Дальнейшее доказательство вытекает из теоремы о минимальном разрезе и максимальном потоке, теоремы 1 и неравенства (1).  $\blacktriangle$

Величина  $\min \text{cut-capacity}(s, D)$  для графа информационных потоков, приведенного на рис. 4, равна

$$\begin{aligned} & \min\{\alpha_1, \beta(U_1, U_2, S_1^{(1)}) + \beta(U_1, U_4, S_1^{(1)})\} + \min\{\alpha_2, \beta(U_2, U_4, S_2^{(1)})\} + \\ & + \min\{\alpha_3, \beta(U_3, U_4, S_3^{(1)})\} = 2 + 1 + 1 = 4 \end{aligned}$$

единицы. Теперь можно составить задачу оптимизации для нахождения минимальных стоимости хранения в системе и стоимости восстановления системы при ограничении, что максимальная возможная информация, поступающая в узел сбора данных  $D$ , не меньше  $B$ .

**Задача 1.**

Минимизировать:  $[C_s(\boldsymbol{\alpha}), C_r(\boldsymbol{\beta})]$

при условиях

Неравенство (4);

$\alpha_i \geq 0$ ;

$\beta(U_i, U_j, S_i^{(\ell)}) \geq 0$ ;

где  $i = 1, 2, \dots, n$ ,  $\ell = 1, 2, \dots, \tau_i$  и  $U_j \in S_i^{(\ell)}$  для некоторого  $j \in \{1, 2, \dots, n\} \setminus \{i\}$ .

*Замечание 3.* Можно вычислить кривую компромисса между стоимостью восстановления и стоимостью хранения с помощью оптимизационной задачи 1 как для точного, так и для функционального восстановления, используя множества выживания как наборы тех вспомогательных узлов, которые участвуют в точном или функциональном восстановлении отказавших узлов соответственно.

Построение оптимальных значений для обеих целевых функций двухкритериальной оптимизационной задачи 1 дает кривую компромисса между  $C_s(\alpha)$  и  $C_r(\beta)$ . В настоящей статье оптимизационная задача 1 решается методом взвешенных сумм для некоторых численных примеров. Некоторые частные случаи оптимизационной задачи 1 рассматриваются в нижеследующих пунктах.

**3.1. Некоторые частные случаи.** Неоднородную РСХД, рассмотренную в этой статье, можно приводить к следующим случаям при некоторых соответствующих ограничениях.

1) (постоянная реконструкция): Если произвольное устройство сбора данных может получить файл, загружая данные из ровно  $k$  узлов для любой их комбинации из  $n$  узлов, то ограничительное неравенство (4) для оптимизационной задачи 1 имеет дополнительное свойство  $k_t = k$  для  $t = 1, 2, \dots, \omega$ .

Задача 2.

Минимизировать:  $[C_s(\alpha), C_r(\beta)]$

при условиях

$$B \leq \min_{\langle U_{\lambda_i} \rangle_{i=1}^k} \sum_{i=1}^k \min \left\{ \alpha_{\lambda_i}, \min_{\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^k} \sum_{\mu_j: U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, \dots, U_{\lambda_{i-1}}\}} \beta(U_{\lambda_i}, U_{\mu_j}, S_{\lambda_i}^{(\ell_{\lambda_i})}) \right\};$$

$$0 \leq \alpha_1 \leq \alpha_2 \leq \dots \leq \alpha_n;$$

где  $\mu_j$  – номер узла  $U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, \dots, U_{\lambda_{i-1}}\}$  для  $i = 1, 2, \dots, k$ .

2) (постоянная степень восстановления): Пусть для неоднородной РСХД отказавший узел можно восстановить с помощью *любой*  $d$  из оставшихся  $n - 1$  узлов. При таком предположении ограничительное неравенство (4) для оптимизационной задачи 1 сводится к следующему:

Задача 3.

Минимизировать:  $[C_s(\alpha), C_r(\beta)]$

при условиях

$$B \leq \min_{\mathcal{A}_t \in \mathcal{A}} \sum_{\substack{i=1 \\ U_{\lambda_i} \in \mathcal{A}_t}}^{k_t} \min \left\{ \alpha_{\lambda_i}, \sum_{\mu_j} \beta(U_{\lambda_i}, U_{\mu_j}, S_{\lambda_i}^{(\ell_{\lambda_i})}) \right\};$$

$$0 \leq \alpha_1 \leq \alpha_2 \leq \dots \leq \alpha_n;$$

$$1 \leq \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_{k_t} \leq n;$$

где  $\mu_j$  – номер узла  $U_{\mu_j} \in S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \{U_{\lambda_1}, \dots, U_{\lambda_{i-1}}\}$ , такого что  $\{U_{\lambda_1}, \dots, U_{\lambda_{i-1}}\} \subset S_{\lambda_i}^{(\ell_{\lambda_i})}$ ,  $i = 1, 2, \dots, k_t$ , и некоторого  $j \in \{1, 2, \dots, d\}$ .

В этом случае  $|S_m^{(\ell_m)}| = d$  и  $\tau_m = \binom{n-1}{d}$  для  $m = 1, 2, \dots, n$ . Здесь значение величины  $\min \text{cut-capacity}(s, D)$  задается последовательностью узлов  $\langle U_{\lambda_i} \rangle_{i=1}^{k_t} \in \mathcal{A}(\mathcal{A}_t)$ , соответствующей последовательности выживания  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^{k_t}$ , такой что  $\alpha_{\lambda_1} \leq \alpha_{\lambda_2} \leq \dots \leq \alpha_{\lambda_{k_t}}$  и  $\{U_{\lambda_1}, U_{\lambda_2}, \dots, U_{\lambda_{i-1}}\} \subset S_{\lambda_i}^{(\ell_{\lambda_i})}$ .

3) (постоянный объем загрузки для восстановления): В этом случае предполагается, что объем загрузки из любого вспомогательного узла для восстановления системы постоянен и равен, скажем,  $\beta$ . Тогда оптимизационная задача 1 при таком ограничении имеет дополнительные свойства  $\beta(U_i, U_j, S_i^{(\ell_i)}) = \beta$ ,  $\beta \geq 0$ ,  $i = 1, 2, \dots, n$ , для всевозможных  $j \in \{1, 2, \dots, n\} \setminus \{i\}$  и  $\ell_i = 1, 2, \dots, \tau_i$ .



Задача 4.

Минимизировать:  $[C_s(\alpha), C_r(\beta)]$

при условиях

$$B \leq \min_{\mathcal{A}_t \in \mathcal{A}} \min_{\langle U_{\lambda_i} \rangle_{i=1}^{k_t}} \sum_{i=1}^{k_t} \min_{U_{\lambda_i} \in \mathcal{A}_t} \left\{ \alpha_{\lambda_i}, \min_{\langle S_{\lambda_i}^{(\ell)} \rangle_{i=1}^{k_t}} \left| S_{\lambda_i}^{(\ell)} \setminus \{U_{\lambda_1}, \dots, U_{\lambda_{i-1}}\} \right| \beta \right\};$$

$$0 \leq \alpha_1 \leq \alpha_2 \leq \dots \leq \alpha_n; \beta \geq 0.$$

4) (однородная РСХД): Неоднородная РСХД сводится к однородной, если все параметры в ней постоянны. Таким образом, предположим, что эффективная степень реконструкции для любого устройства сбора данных равна  $k$ , а объем памяти каждого узла равен  $\alpha$ . Кроме того, пусть отказавший узел может быть восстановлен с помощью *любых*  $d$  из оставшихся  $n-1$  узлов путем загрузки  $\beta$  пакетов из каждого вспомогательного узла. При таких условиях ограничительное неравенство (4) для оптимизационной задачи 1 сводится к следующему:

Задача 5.

Минимизировать:  $[C_s(\alpha), C_r(\beta)]$

при условиях

$$B \leq \sum_{i=1}^k \min\{\alpha, (d-i-1)\beta\};$$

$$\alpha \geq 0;$$

$$\beta \geq 0.$$

5) (прочее): Рассмотренную в статье модель неоднородной РСХД можно сводить к еще более специальным РСХД при некоторых соответствующих условиях на ограничения. Например, неоднородные РСХД с постоянной реконструкцией и с постоянной степенью восстановления (случаи 1) и 2) соответственно) совместно приводят к случаю неоднородной РСХД, рассмотренной в [28].

Можно искать решения двухкритериальной оптимизационной задачи 1 для некоторых численных значений и по этим решениям строить кривую компромисса для нее. Можно сравнить полученную кривую компромисса с кривой компромисса для существующей неоднородной РСХД, исследованной в [28]. Итак, в следующем параграфе мы вычисляем некоторые оптимальные решения для численных значений параметров нашей модели и сравниваем ее с однородной моделью из [7] и неоднородной моделью из [28].

**3.2. Численные результаты.** Для оптимизационной задачи 1 мы решали задачи линейного программирования с одной целевой функцией. Эта целевая функция выбиралась как линейная комбинация двух целевых функций оптимизационной задачи 1. Затем эти задачи линейного программирования были решены с выбором различных отношений коэффициентов линейной комбинации в пределах от  $10^{-3}$  до  $10^3$ . Построение компромиссных соотношений и решения задач линейного программирования были выполнены с помощью программных средств MATLAB и `lp_solve` [38].

На рис. 5 построены четыре кривые компромисса между стоимостью восстановления системы  $C_r$  и стоимостью хранения в системе  $C_s$  для соответствующих РСХД. В частности, одна кривая на рис. 5 построена для однородной РСХД из [7], еще одна – для неоднородной РСХД из [28], а остальные две – для неоднородных РСХД, рассмотренных в настоящей статье. А именно, одна из двух последних кривых имеет минимальную эффективную степень реконструкции  $k_{\min}$ , равную 2,

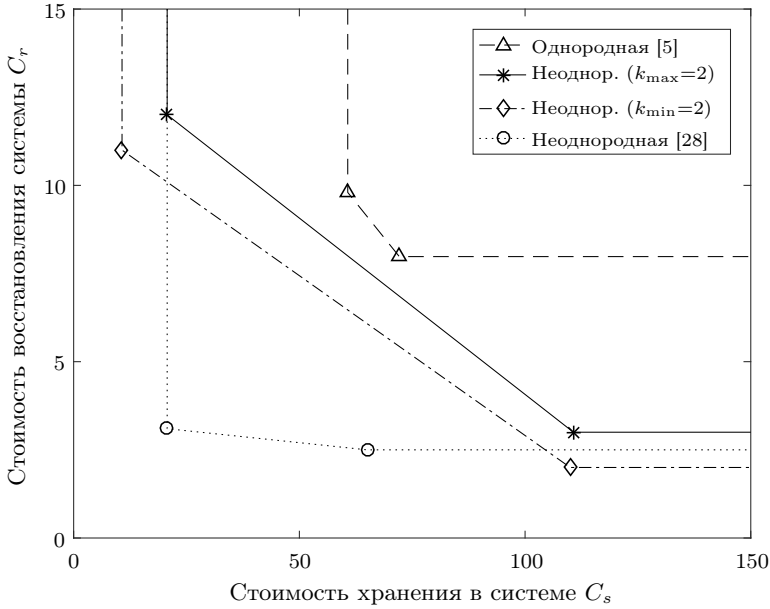


Рис. 5. Кривая оптимального компромисса между стоимостью восстановления системы  $C_r$  и стоимостью хранения в системе  $C_s$  для различных РСХД

а вторая – максимальную эффективную степень восстановления  $k_{\max}$ , равную 2. Общие для всех рассматриваемых РСХД параметры следующие:  $n = 4$ ,  $B = 1$  единица,  $s = (1 \ 10 \ 10 \ 100)$  и  $r = (10 \ 1 \ 1 \ 1)$ . Однородная РСХД и неоднородная РСХД из [28] имеют степень реконструкции  $k = 2$  и степень восстановления  $d = 3$ . Обе оставшиеся неоднородные РСХД имеют множества выживания  $S_1^{(1)} = \{U_2, U_3, U_4\}$ ,  $S_2^{(1)} = \{U_1, U_4\}$ ,  $S_3^{(1)} = \{U_1, U_2\}$  и  $S_4^{(1)} = \{U_2, U_3\}$ .

Из рис. 5 видно, что наша модель неоднородной РСХД имеет более оптимальные стоимости хранения и восстановления, чем однородная РСХД, рассмотренная в [7]. Несмотря на то, что характеристики нашей неоднородной модели и неоднородной модели, исследованной в [28], различны, мы получили несколько более оптимальных точек для нашей модели, как показано на рис. 5. В п. 3.1 было показано, что неоднородную РСХД из [28] можно получить из нашей модели, накладывая в ней определенные условия.

*Замечание 4.* Заметим, что уменьшение размера произвольного файла  $B$  до 1 приводит к тому, что соответствующие целочисленные решения могут перестать быть целочисленными. Поэтому на конкретных кривых компромисса можно также рассматривать и нецелочисленные решения двухкритериальной оптимизационной задачи 1.

Теперь рассмотрим пример, приведенный на рис. 2. В этом примере средняя степень реконструкции равна 2,286 для данной неоднородной РСХД. Средние степени восстановления для узлов равны (2, 2, 2, 2,5, 2), и среднее значение этих средних степеней восстановления равно 2,1. Для отказов узлов в РСХД, рассматриваемой в этом примере, значения ширины восстановления приведены в табл. 1. Таким образом, для вектора стоимости хранения  $s = (1, 1, 1, 1, 1)$  и вектора стоимости восстановления  $r = (1, 1, 1, 1, 1)$  стоимость хранения в системе и стоимость восстановления системы равны 2,750 и 3,667 единиц соответственно. Поэтому стоимость хранения на узел и стоимость восстановления на узел равны 0,550 и 0,733 единиц соответствен-

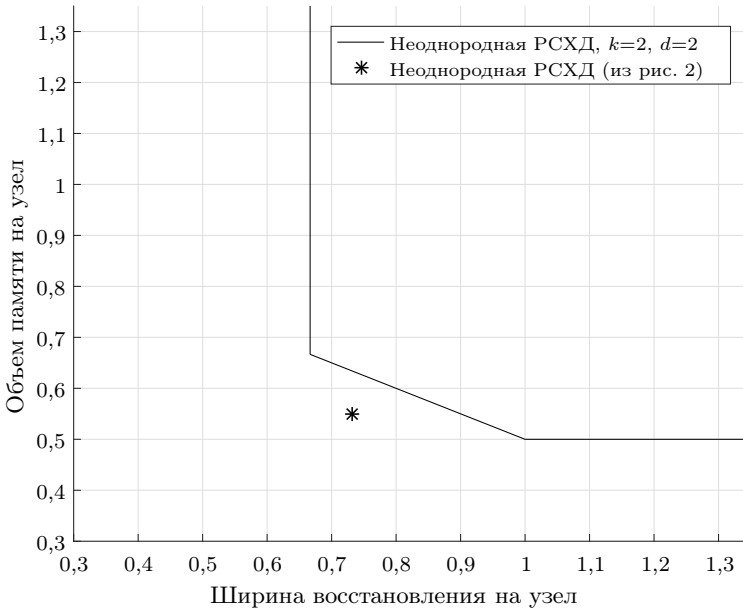


Рис. 6. Кривая компромисса между объемом памяти на узел и шириной восстановления на узел для однородной РСХД в сравнении со значениями этих параметров для примера неоднородной РСХД, приведенной на рис. 2

но. Далее, кривая компромиссного соотношения для однородной РСХД со степенью реконструкции 2 и степенью восстановления 2 построена на рис. 6. Этот рисунок приведен для сравнения примера (рис. 2) неоднородной РСХД с однородной РСХД, имеющей  $k = 2$  и  $d = 2$ .

#### § 4. Анализ границы

**4.1. Модель.** В неоднородной РСХД файл делится на пакеты, и эти закодированные пакеты распределяются по  $n$  различным узлам  $U_i, i = 1, 2, \dots, n$ , где каждый узел имеет объем памяти  $\alpha_i$  и степень восстановления  $d_i$ . Пользователь может реконструировать файл, загружая данные из любых  $k$  ( $< n$ ) узлов. Если узел  $U_i$  откажет, то тогда устройство сбора данных загрузит  $\beta$  пакетов из некоторых  $d_i$  узлов, специально выбираемых из оставшихся  $n - 1$  узлов. Эти  $d_i$  узлов называются вспомогательными узлами для отказавшего узла  $U_i$ . В таком случае ширина восстановления для узла  $U_i$  равна  $\gamma_i = d_i \beta$ . Заметим, что леммы 2 и 3 выведены для таких неоднородных РСХД.

Пример такой неоднородной РСХД показан на рис. 7. В этом примере файл размера 3 ( $= B$ ) хранится в неоднородной РСХД с параметрами ( $n = 6, k = 2$ ) и трафиком восстановления  $\beta$ , равным 1. В этой РСХД объем памяти узла  $\alpha_i$  равен 2, 2, 2, 3, 2, 2 для  $i = 1, 2, 3, 4, 5, 6$  (см. рис. 7). Заметим, что  $\alpha_i = \gamma_i = d_i$  для  $i = 1, 2, 3, 4, 5, 6$ . В [27] рассматривалось представление неоднородной РСХД в виде ациклического ориентированного графа, называемого графом информационных потоков. С помощью анализа минимального разреза в графе информационных потоков вычисляется фундаментальная граница на размер файла  $B$  для такой неоднородной РСХД. Эту границу (специальный случай леммы 1) описывает следующая

*Лемма 2 (фундаментальная граница). Для неоднородной РСХД с  $n$  узлами и степенью реконструкции  $k$  размер файла  $B$  должен удовлетворять следующему*

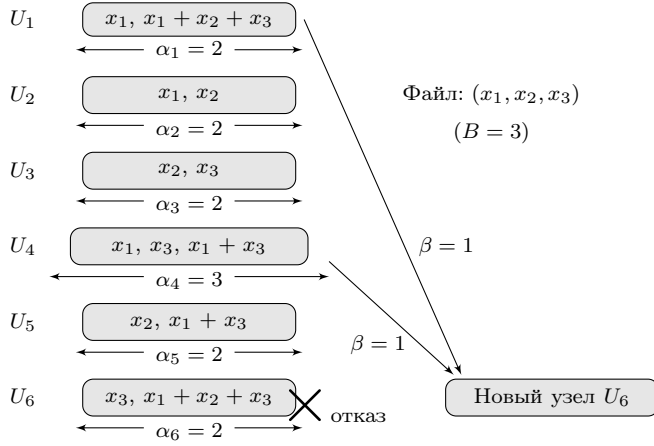


Рис. 7. Файл разделен на 3 ( $= B$ ) различных пакета  $x_1$ ,  $x_2$  и  $x_3$  над полем  $\mathbb{F}_q$ . Эти три пакета закодированы в тринадцать различных пакетов и распределены по неоднородной РСХД с шестью узлами и степенью реконструкции 2

неравенству:

$$B \leq \min_{\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^n} \left\{ \sum_{i=1}^k \min \left\{ \alpha_{\lambda_i}, \left| S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \left( \bigcup_{j=0}^{i-1} \{U_{\lambda_j}\} \right) \right| \beta \right\} \right\},$$

где  $\{U_{\lambda_0}\} = \emptyset$ ,  $0 \leq j < i \leq k$ ,  $S_{\lambda_i}^{(\ell_{\lambda_i})} \in \langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^n$  и  $\ell_{\lambda_i} \in \{1, 2, \dots, \tau_{\lambda_i}\}$ .

**Доказательство.** Для рассматриваемой неоднородной РСХД любое множество  $\mathcal{A} \subset \{U_1, U_2, \dots, U_n\}$  мощности  $k$  является реконструирующим. Значит, любая последовательность узлов длины  $k$  будет последовательностью узлов для этой неоднородной РСХД. Тем самым, любая последовательность длины  $k$ , состоящая из множеств выживания, будет также последовательностью выживания для этой неоднородной РСХД. Поэтому доказательство следует из леммы 1.  $\blacktriangle$

В [27] было показано существование кода, достигающего фундаментальной границы для таких неоднородных РСХД с параметрами  $(n, k)$ . Следовательно, можно получать оптимальные коды, уменьшая параметры, лежащие на фундаментальной границе. В следующем пункте вычисляются параметры оптимальных кодов, получаемые минимизацией объема памяти узла и ширины восстановления.

**4.2. Условия оптимальности.** Рассмотрим неоднородную РСХД с параметрами  $(n, k)$  с  $\tau_i$  множествами выживания  $S_i^{(\ell_i)}$  и степенями восстановления  $|S_i^{(\ell_i)}| = d_i$ ,  $\ell_i = 1, 2, \dots, \tau_i$ ,  $i = 1, 2, \dots, n$ . Если  $\alpha_i > |S_i^{(\ell_i)}| \beta$ , то отказавший узел  $U_i$  восстановить нельзя, поэтому  $\alpha_i \leq |S_i^{(\ell_i)}| \beta$  для всех  $i$  и  $\ell_i$ . Для оптимальности должно выполняться равенство  $\alpha_i = d_i \beta$ . Следовательно, при постоянном трафике восстановления  $\beta$  объем памяти узла  $\alpha_i$  и степень восстановления  $d_i$  должны быть пропорциональны друг другу. Рассмотрим  $c_i \in (0, 1) \subset \mathbb{R}$ , такие что  $\sum_{i=1}^n c_i = 1$  и  $\frac{c_i}{\alpha_i} = \frac{c_j}{\alpha_j}$  при  $1 \leq i < j \leq n$ . Тогда  $\alpha_i = c_i \sum_{i=1}^n \alpha_i = c_i \alpha^*$ ,  $i = 1, 2, \dots, n$ . Таким образом, параметры  $\alpha_i$  и  $c_i$  пропорциональны друг другу. Снова, поскольку  $k$  – степень реконструкции, то  $B \leq \sum_{i \in \mathcal{K}} \alpha_i = \sum_{i \in \mathcal{K}} c_i \alpha^*$  для произвольного множества  $\mathcal{K} \subset \{1, 2, \dots, n\}$ , такого что

$|\mathcal{K}| = k$ . Отсюда  $B \leq \sum_{i=1}^k c_i \alpha^*$  для  $c_1 \leq c_2 \leq \dots \leq c_n$ . Для получения оптимального случая можно уменьшить  $\alpha^*$  до  $\alpha_{\min}^*$ , такого что

$$B = \sum_{i=1}^k c_i \alpha_{\min}^* \implies \alpha_{\min}^* = B \left( \sum_{j=1}^k c_j \right)^{-1}. \quad (5)$$

Аналогично при фиксированном коэффициенте пропорциональности  $\alpha_{\min}^*$  можно минимизировать трафик восстановления  $\beta$  таким образом, чтобы граница из теоремы 2 выполнялась с равенством. Для фиксированной последовательности выживания  $\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^n$  с достаточно большим трафиком восстановления  $\beta$  неравенство  $\alpha_{\lambda_i} \leq \left| S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \left( \bigcup_{j=0}^{i-1} \{U_{\lambda_j}\} \right) \right| \beta$  выполняется для любого  $i = 1, 2, \dots, k$ . Если выбрать  $\beta = \beta_{\min}$  так, чтобы

$$\beta_{\min} = \max_{\langle S_{\lambda_i}^{(\ell_{\lambda_i})} \rangle_{i=1}^n} \left\{ \max_{1 \leq i \leq k} \left\{ \alpha_{\lambda_i} \left| S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \left( \bigcup_{j=0}^{i-1} \{U_{\lambda_j}\} \right) \right|^{-1} \right\} \right\}, \quad (6)$$

то  $\beta_{\min}$  будет минимальным значениям трафика восстановления  $\beta$ , при котором

$$\left| S_{\lambda_i}^{(\ell_{\lambda_i})} \setminus \left( \bigcup_{j=0}^{i-1} \{U_{\lambda_j}\} \right) \right| \beta_{\min} \geq \alpha_{\lambda_i}$$

для любого  $\lambda_i$  из произвольной последовательности выживания. Формально результат можно сформулировать следующим образом.

*Лемма 3. Рассмотрим неоднородную РСХД, имеющую  $n$  узлов, степень реконструкции  $k$  и множества выживания  $S_i^{(\ell_i)}$ ,  $i = 1, 2, \dots, n$ ,  $\ell_i = 1, 2, \dots, \tau_i$  для некоторого  $\tau_i \in \mathbb{Z}$ . Семейство кодов с  $\alpha_i = c_i \alpha_{\min} = d_i \beta_{\min}$  и  $\beta = \beta_{\min}$  достигает фундаментальной границы (теорема 2), где  $\alpha_{\min}$  и  $\beta_{\min}$  вычисляются по формулам (5) и (6).*

## § 5. Заключение

В статье предложена модель неоднородной РСХД с переменными степенью реконструкции, объемом памяти узла и шириной восстановления. В частности, файл может быть реконструирован с помощью определенного множества узлов, а система может быть восстановлена после отказа любого узла путем установления соединения с некоторым множеством вспомогательных узлов. Для такой неоднородной РСХД исследованы фундаментальные кривые компромисса между стоимостью восстановления системы и стоимостью хранения в системе. Для построения кривой компромисса сформулирована задача двухкритериальной оптимизации с ограничениями в виде границы минимального разреза и неотрицательности параметров этой неоднородной РСХД. Задача двухкритериальной оптимизации решена методом взвешенных сумм для некоторых численных значений параметров этой неоднородной модели. При анализе кривой компромисса наблюдается некоторые новые оптимальные точки по сравнению с существующей неоднородной моделью [28]. Рассмотренная модель близка к реальным сценариям. Наша неоднородная модель достаточно гибкая, что позволяет преобразовать ее в любую существующую неоднородную или однородную РСХД путем наложения соответствующих ограничений. Интересной задачей представлялось бы построение кодов, достигающих оптимальных точек на кривой компромисса.

Авторы благодарны рецензентам за внимательное прочтение рукописи, способствовавшее улучшению изложения.

## СПИСОК ЛИТЕРАТУРЫ

1. Amazon Elastic Compute Cloud (Amazon EC2). Web Service. Jan. 2013. Available at <http://aws.amazon.com/ec2/>.
2. Huang C., Simitci H., Xu Y., Ogun A., Calder B., Gopalan P., Li J., Yekhanin S. Erasure Coding in Windows Azure Storage // Proc. 2012 USENIX Annu. Technical Conf. (USENIX ATC'12). Boston, MA. June 13–15, 2012. P. 15–26.
3. Microsoft SkyDrive Live. Online Storage Service. Jan. 2013. Available at <https://skydrive.live.com/>.
4. Sathiamoorthy M., Asteris M., Papailiopoulos D., Dimakis A.G., Vadali R., Chen S., Borthakur D. XORing Elephants: Novel Erasure Codes for Big Data // Proc. VLDB Endow. 2013. V. 6. № 5. P. 325–336. <https://doi.org/10.14778/2535573.2488339>
5. Dimakis A.G., Godfrey P.B., Wu Y., Wainwright M.J., Ramchandran K. Network Coding for Distributed Storage Systems // Proc. 26th IEEE Annu. Joint Conf. on Computer Communications (INFOCOM'2007). Anchorage, AK, USA. May 6–12, 2007. P. 2000–2008. <https://doi.org/10.1109/INFCOM.2007.232>
6. Dimakis A.G., Godfrey P.B., Wu Y., Wainwright M.J., Ramchandran K. Network Coding for Distributed Storage Systems // IEEE Trans. Inform. Theory. 2010. V. 56. № 9. P. 4539–4551. <https://doi.org/10.1109/TIT.2010.2054295>
7. Wu Y., Dimakis A., Ramchandran K. Deterministic Regenerating Codes for Distributed Storage // Proc. 45th Annu. Allerton Conf. on Communication, Control, and Computing. Monticello, IL, USA. Sept. 26–28, 2007. V. 1. P. 242–249.
8. Wu Y. Existence and Construction of Capacity-Achieving Network Codes for Distributed Storage // Proc. 2009 IEEE Int. Symp. on Information Theory (ISIT'2009). Seoul, Korea. June 28–July 3, 2009. P. 1150–1154. <https://doi.org/10.1109/ISIT.2009.5206008>
9. Wu Y. Existence and Construction of Capacity-Achieving Network Codes for Distributed Storage // IEEE J. Sel. Areas Commun. 2010. V. 28. № 2. P. 277–288. <https://doi.org/10.1109/JSAC.2010.100217>
10. Goparaju S., El Rouayheb S., Calderbank R. New Codes and Inner Bounds for Exact Repair in Distributed Storage Systems // Proc. 2014 IEEE Int. Symp. on Information Theory (ISIT'2014). Honolulu, HI, USA. June 29–July 4, 2014. P. 1036–1040. <https://doi.org/10.1109/ISIT.2014.6874990>
11. Shah N.B., Rashmi K.V., Kumar P.V. A Flexible Class of Regenerating Codes for Distributed Storage // Proc. 2010 IEEE Int. Symp. on Information Theory (ISIT'2010). Austin, TX, USA. June 13–18, 2010. P. 1943–1947. <https://doi.org/10.1109/ISIT.2010.5513353>
12. Dimakis A.G., Ramchandran K., Wu Y., Suh C. A Survey on Network Codes for Distributed Storage // Proc. IEEE. 2011. V. 99. № 3. P. 476–489. <https://doi.org/10.1109/JPROC.2010.2096170>
13. Prakash N., Krishnan M.N. The Storage-Repair-Bandwidth Trade-off of Exact Repair Linear Regenerating Codes for the Case  $d = k = n - 1$ , <https://arXiv.org/abs/1501.03983v2> [cs.IT], 2015.
14. Kubiawicz J., Bindel D., Chen Y., Czerwinski S., Eaton P., Geels D., Gummadi R., Rhea S., Weatherspoon H., Weimer W., Wells C., Zhao B. OceanStore: An Architecture for Global-Scale Persistent Storage // ACM SIGPLAN Notices. 2000. V. 35. № 11. P. 190–201. <https://doi.org/10.1145/356989.357007>
15. Bianchi G., Melen R. Performance and Dimensioning of a Hierarchical Video Storage Network for Interactive Video Services // Eur. Trans. Telecommun. 1996. V. 7. № 4. P. 349–358. <https://doi.org/10.1002/ett.4460070407>
16. Pawar S., El Rouayheb S., Zhang H., Lee K., Ramchandran K. Codes for a Distributed Caching Based Video-on-Demand System // Conf. Rec. 46th Asilomar Conf. on Signals,

- Systems and Computers (ASILOMAR'2011). Pacific Grove, CA, USA. Nov. 6–9, 2011. P. 1783–1787. <https://doi.org/10.1109/ACSSC.2011.6190328>
17. *Ntranos V., Caire G., Dimakis A.G.* Allocations for Heterogenous Distributed Storage, <https://arXiv.org/abs/1202.1596> [cs.IT], 2012.
  18. *Li Z., Ho T., Leong D., Yao H.* Distributed Storage Allocation for Heterogeneous Systems // Proc. 51st Annu. Allerton Conf. on Communication, Control, and Computing. Monticello, IL, USA. Oct. 2–4, 2013. P. 320–326. <https://doi.org/10.1109/Allerton.2013.6736541>
  19. *Leong D., Dimakis A.G., Ho T.* Distributed Storage Allocations // IEEE Trans. Inform. Theory. 2012. V. 58. № 7. P. 4733–4752. <https://doi.org/10.1109/TIT.2012.2191135>
  20. *Gerami M., Xiao M., Skoglund M.* Optimal-Cost Repair in Multi-hop Distributed Storage Systems // Proc. 2011 IEEE Int. Symp. on Information Theory (ISIT'2011). St. Petersburg, Russia. July 31–Aug. 5, 2011. P. 1437–1441. <https://doi.org/10.1109/ISIT.2011.6033777>
  21. *Akhlaghi S., Kiani A., Ghanavati M.R.* Cost-Bandwidth Tradeoff in Distributed Storage Systems // Comput. Commun. 2010. V. 33. № 17. P. 2105–2115. <https://doi.org/10.1016/j.comcom.2010.07.022>
  22. *Akhlaghi S., Kiani A., Ghanavati M.R.* A Fundamental Trade-off between the Download Cost and Repair Bandwidth in Distributed Storage Systems // Proc. 2010 IEEE Int. Symp. on Network Coding (NetCod'2010). Toronto, ON, Canada. June 9–11, 2010. P. 97–102. <https://doi.org/10.1109/NETCOD.2010.5487685>
  23. *Yu Q., Shum K.W., Sung C.W.* Minimization of Storage Cost in Distributed Storage Systems with Repair Consideration // Proc. 2011 IEEE Global Telecommunications Conf. (GLOBECOM'2011). Houston, TX, USA. Dec. 5–9, 2011. P. 2931–2935. <https://doi.org/10.1109/GLOCOM.2011.6133729>
  24. *Pernas J., Yuen C., Gastón B., Pujol J.* Non-homogeneous Two-Rack Model for Distributed Storage Systems // Proc. 2013 IEEE Int. Symp. on Information Theory (ISIT'2013). Istanbul, Turkey. July 7–12, 2013. P. 1237–1241. <https://doi.org/10.1109/ISIT.2013.6620424>
  25. *Gastón B., Pujol J., Villanueva M.* A Realistic Distributed Storage Systems That Minimizes Data Storage and Repair Bandwidth // Proc. 2006 Data Compression Conf. (DCC'2006). Snowbird, UT, USA. Mar. 20–22, 2013. P. 491. <https://doi.org/10.1109/DCC.2013.72>
  26. *Ernwall T., El Rouayheb S., Hollanti C., Poor H.V.* Capacity and Security of Heterogeneous Distributed Storage Systems // Proc. 2013 IEEE Int. Symp. on Information Theory (ISIT'2013). Istanbul, Turkey. July 7–12, 2013. P. 1247–1251. <https://doi.org/10.1109/ISIT.2013.6620426>
  27. *Benerjee K.G., Gupta M.K.* On Heterogeneous Regenerating Codes and Capacity of Distributed Storage Systems, <https://arXiv.org/abs/1402.3801>, [cs.IT], 2014.
  28. *Yu Q., Shum K.W., Sung C.W.* Tradeoff between Storage Cost and Repair Cost in Heterogeneous Distributed Storage Systems // Trans. Emerg. Commun. Technol. 2015. V. 26. № 10. P. 1201–1211. <https://doi.org/10.1002/ett.2887>
  29. *Kiani A., Akhlaghi S.* Selective Regenerating Codes // IEEE Commun. Lett. 2011. V. 15. № 8. P. 854–856. <https://doi.org/10.1109/LCOMM.2011.061611.102271>
  30. *Senthooor K., Sasidharan B., Kumar P.V.* Improved Layered Regenerating Codes Characterizing the Exact-Repair Storage-Repair Bandwidth Tradeoff for Certain Parameter Sets // Proc. 2015 IEEE Information Theory Workshop (ITW'2015). Jerusalem, Israel. Apr. 26–May 1, 2015. P. 224–228. <https://doi.org/10.1109/ITW.2015.7133121>
  31. *Sasidharan B., Senthooor K., Kumar P.V.* An Improved Outer Bound on the Storage-Repair Bandwidth Tradeoff of Exact-Repair Regenerating Codes // Proc. 2014 IEEE Int. Symp. on Information Theory (ISIT'2014). Honolulu, HI, USA. June 29–July 4, 2014. P. 2430–2434. <https://doi.org/10.1109/ISIT.2014.6875270>
  32. *Sasidharan B., Kumar P.V.* On the Interior Points of the Storage-Repair Bandwidth Tradeoff of Regenerating Codes // Proc. 51st Annu. Allerton Conf. on Communication, Control, and Computing. Monticello, IL, USA. Oct. 2–4, 2013. P. 788–795. <https://doi.org/10.1109/Allerton.2013.6736605>
  33. *Duursma I.M.* Outer Bounds for Exact Repair Codes, <https://arXiv.org/abs/1406.4852> [cs.IT], 2014.

34. *Ahmad I., Wang C.C.* When and by How Much Can Helper Node Selection Improve Regenerating Codes? // Proc. 52nd Annu. Allerton Conf. on Communication, Control, and Computing. Monticello, IL, USA. Sept. 30–Oct. 3, 2014. P. 459–466. <https://doi.org/10.1109/ALLERTON.2014.7028491>
35. *Ahlsvede R., Cai N., Li S.-Y.R., Yeung R.W.* Network Information Flow // IEEE Trans. Inform. Theory. 2000. V. 46. № 4. P. 1204–1216. <https://doi.org/10.1109/18.850663>
36. *Elias P., Feinstein A., Shannon C.* A Note on the Maximum Flow Through a Network // IEEE Trans. Inform. Theory. 1956. V. 2. № 4. P. 117–119. <https://doi.org/10.1109/TIT.1956.1056816>
37. *Ford L.R., Jr., Fulkerson D.R.* Maximal Flow through a Network // Canad. J. Math. 1956. V. 8. P. 399–404. <https://doi.org/10.4153/CJM-1956-045-5>
38. *lp\_solve* (mathematical optimization software). A Mixed Integer Linear Programming (MILP) Solver. Version 5.5.2.0, 2011. Available at <http://lpsolve.sourceforge.net/5.5/>.

*Бенерджи Кришна Гопал*  
*Гупта Маниш Кумар*<sup>✉</sup>  
 Институт информационных технологий и техники связи  
 им. Дхирубхая Амбани, Гандинагар, штат Гуджарат, Индия  
<sup>✉</sup>[mankg@computer.org](mailto:mankg@computer.org)

Поступила в редакцию  
 13.08.2019  
 После доработки  
 02.10.2020  
 Принята к публикации  
 30.12.2020