

DOI: 10.15514/ISPRAS-2026-38(4)-1



К организации системы запросов в распределённой сети с кластерами диаметра не более 2

¹ И.Б. Бурдонов, ORCID: 0000-0001-9539-7853 <igor@ispras.ru>

^{1,2} Н.В. Евтушенко, ORCID: 0000-0002-4006-1161 <evtushenko@ispras.ru>

¹ А.С. Косачев, ORCID: 0000-0001-5316-3813 <kos@ispras.ru>

¹ В.Н. Пономаренко, ORCID: 0009-0002-2387-2760 <vera@ispras.ru>

¹ Институт системного программирования РАН им. В.П. Иванникова,
109004, Россия, г. Москва, ул. А. Солженицына, д. 25.

² Национальный исследовательский университет «Высшая школа экономики»,
101000, Россия, г. Москва, ул. Мясницкая, д. 20.

Аннотация. В статье усовершенствуется предложенная в предыдущей статье авторов модель кластеризованной распределённой сети с опросом, соответствующая сервис-ориентированной архитектуре. Целью опроса является нахождение ближайшего узла, готового оказать требуемую услугу (сервис) как можно раньше. Предлагаемые алгоритмы обеспечивают достижимость узлов, отсутствие дублирования сообщений, и обмен сообщениями с найденным узлом по кратчайшему пути. Кластеризованность сети означает, что в графе сети выделены подграфы – кластеры, покрывающие все узлы графа. Отсутствие дублирования гарантируется, если двудольный граф кластеров и общих (принадлежащих нескольким кластерам) узлов является деревом. Для поиска ближайшего узла с нужными свойствами опрос выполняется последовательно по раундам: на r -м раунде опрашиваются узлы на концах путей, начинающихся в начальном узле, запрашивающем услугу, и проходящих через $r - 1$ промежуточных общих узлов. В предыдущей статье авторов рассматривалась кластеризация сети, при которой каждый кластер был кликой, т.е. графом диаметра 1, однако число таких кластеров может получиться достаточно большим. В настоящей статье это требование ослабляется: кластер может быть графом диаметра не более 2. Это существенное ослабление, поскольку, во-первых, как известно, почти все графы имеют диаметр 2, во-вторых, число кластеров уменьшается примерно в 2 раза и также примерно в 2 раза уменьшается число требуемых раундов. Кроме того, показывается, что для кластеров диаметра больше 2 избежать дублирования не всегда возможно.

Ключевые слова: распределенная кластеризованная сеть; опрос сети; достижимость узлов; дублирование сообщений; кратчайший путь; подграфы диаметра не более 2.

Для цитирования: Бурдонов И.Б., Евтушенко Н.В., Косачев А.С., Пономаренко В.Н. К организации системы запросов в распределенной сети с кластерами диаметра не более 2. Труды ИСП РАН, том 38, вып. 4, часть 1, 2026 г., стр. 7–24. DOI: 10.15514/ISPRAS-2026-38(4)-1.

On the query system organization of a distributed network with clusters of diameter at most two

¹ I.B. Burdonov, ORCID: 0000-0001-9539-7853 <igor@ispras.ru>

^{1,2} N.V. Yevtushenko, ORCID: 0000-0002-4006-1161 <evtushenko@ispras.ru>

¹ A.S. Kossatchev, ORCID: 0000-0001-5316-3813 <kos@ispras.ru>

¹ V. N. Ponomarenko, ORCID: 0009-0002-2387-2760 <vera@ispras.ru>

¹ Institute for System Programming of the Russian Academy of Sciences,
25, Alexander Solzhenitsyn st., Moscow, 109004, Russia.

² National Research University Higher School of Economics,
20, Myasnitskaya st., Moscow, 101000, Russia.

Abstract. In this paper, an improved model of a service-oriented architecture as a clustered distributed network is considered. The improvements are related to a query system proposed in the previous paper of the authors. The polling purpose is location determination of the closest node that can provide a required service. Proposed techniques provide the node reachability and duplication absence; moreover, a shortest path is utilized for providing the communication. In a clustered network, clusters as subgraphs of the network are selected. The duplication absence is guaranteed if a bipartite graph of clusters and common nodes of several clusters is a tree. For finding the nearest node with necessary features, a bypass is sequentially performed: at the r th step the terminal nodes of the bypass are examined which starts at the initial node and passes via $r-1$ intermediate nodes. In the previous paper, every cluster had to be a clique that is a graph of diameter one and a number of such clusters can be large. In this paper, this requirement is relaxed: a cluster can be a subgraph of diameter at most two. This is essential, since almost all graphs have a diameter at most two and secondly the cluster number approximately decreases by half and in the same way, the number of polling steps is decreased. We also show that for clusters of diameter bigger than two, it is not always possible to implement the duplication absence.

Keywords: distributed clustered network; service-oriented architecture; node reachability; message duplication; shortest path; graphs of diameter 2.

For citation: Burdonov I.B., Yevtushenko N.V., Kossatchev A.S., Ponomarenko V.N. On the query system organization of a distributed network with clusters of diameter no more than 2. Trudy ISP RAN/Proc. ISP RAS, vol. 38, issue 4, part 1, 2026, pp. 7-24 (in Russian). DOI: 10.15514/ISPRAS-2026-38(4)-1.

1. Введение

Статья является продолжением исследования [1], в котором предложена модель кластеризованной распределённой сети с опросом, соответствующая сервис-ориентированной архитектуре (SOA – service oriented architecture). В такой сети узлы являются вычислительными единицами, и каждый узел может запросить ту или иную услугу (сервис), которая может быть оказана каким-то другим, но заранее неизвестным, узлом сети. Для этого выполняется опрос сети, аналогичный поиску в ширину для того, чтобы найти ближайший узел с необходимыми свойствами.

Предлагаемые алгоритмы обеспечивают достижимость узлов, отсутствие дублирования сообщений, и выполнение обмена сообщениями с найденным узлом по кратчайшему пути. Кластеризованность сети означает, что в графе сети выделены подграфы – кластеры. При этом требуется, чтобы каждый узел сети принадлежал хотя бы одному кластеру, а двудольный граф кластеров и общих (принадлежащих нескольким кластерам) узлов, в котором ребро соединяет общий узел с кластером, которому узел принадлежит, был деревом. Для того чтобы найти ближайший узел с необходимыми свойствами, опрос сети выполняется последовательно по раундам, когда на каждом r -м раунде опрашиваются узлы, для которых кратчайший путь, соединяющий их с начальным узлом, запрашивающим услугу, проходит через $r - 1$ промежуточных общих узлов, и переход на следующий раунд происходит только в том случае, когда на текущем раунде нужный узел не найден.

В предыдущей статье авторов [1] рассматривался случай, когда каждый кластер является *кликкой*, то есть графом диаметра 1, что минимизирует обмен сообщениями внутри кластера. Это требование может оказаться слишком сильным, а число требуемых кластеров слишком большим. С другой стороны, многие встречающиеся на практике сети удовлетворяют ослабленному требованию: в них кластеры – это «почти»-клики. Графы связей, удовлетворяющие этому ослабленному требованию, близки графам «мир тесен» [2][3]. Название происходит от гипотезы о «шести рукопожатиях» [4][5]: любые два человека на Земле разделены не более чем пятью уровнями общих знакомых и, соответственно, шестью уровнями связей. На математическом языке это означает, что диаметр графа знакомств не превышает 6. В графе «мир тесен» два произвольных узла с большой вероятностью не являются смежными, однако соединены достаточно коротким путём. Графы «мир тесен», во-первых, имеют тенденцию содержать в себе кластеры, которые являются «почти»-кликами, а, во-вторых, в графе «мир тесен» расстояние от одного узла до другого в среднем относительно мало. Свойства графа «мир тесен» были обнаружены не только в социальных сетях, но также во многих объектах реального мира, включая карты дорог, пищевые цепочки, электрические сети, сети протекания метаболизма, нейронные сети, сеть телефонных звонков и т.п.

В настоящей статье в качестве «почти»-клики предлагается рассматривать граф диаметра 2. Это существенное ослабление требования «кластер – это клика», поскольку, во-первых, как известно, *почти все* графы имеют диаметр 2, во-вторых, число кластеров в сети уменьшается примерно в 2 раза и также примерно в 2 раза уменьшается число требуемых раундов.

Структура статьи следующая. В разделе 2 формулируются требования к графу связей и его кластеризации и рассматриваются правила обмена сообщениями. В разделе 3 показывается выполнение этих правил для случая, когда кластер является кликой как в работе [1]. В разделе 4 для кластеров диаметра не более 2 предлагаются две процедуры кластеризации: процедура объединения блоков (компонент двусвязности) в блоковом графе (графе, в котором все блоки – это клики), и процедура разъединения максимальных подграфов диаметра не более 2. В разделе 5 рассматривается коммутация сообщений опроса внутри кластера и показывается, что для кластеров диаметра не более 2 не возникает дублирования сообщений, тогда как для кластеров диаметра больше 2 такое дублирование может возникнуть. В заключении подводятся итоги и намечаются направления дальнейших исследований.

2. Граф связей, кластеризация и обмен сообщениями

Сеть моделируется связным неориентированным графом без кратных рёбер и петель. Как принято в распределённых сетях, вершины графа будем называть *узлами*. Узлы соответствуют вычислительным единицам, которые обмениваются между собой сообщениями, посылаемыми по рёбрам графа. Этот граф называется *графом связей*. В сети с опросом каждый узел может запросить ту или иную услугу, которая может быть оказана каким-то другим узлом сети. Каждый узел знает, какие услуги он может оказывать.

Кластером будем называть связный подграф графа (подмножество вершин графа и всех рёбер, оба конца которых принадлежат этому подмножеству). Будем говорить, что задана *кластеризация графа*, если задан некоторый набор его кластеров. *Графом кластеров и общих узлов* будем называть двудольный граф, в котором узел – это кластер графа или общий узел, т.е. узел, принадлежащий более чем одному кластеру, а ребро соединяет кластер и общий узел тогда и только тогда, когда этот узел принадлежит этому кластеру.

Набор кластеров графа будем называть *правильным*, если выполняются следующие требования:

- 1) каждый узел графа входит хотя бы в один кластер,
- 2) граф кластеров и общих узлов является деревом.

Граф, в котором можно выделить правильный набор кластеров, будем называть *правильным* графом.

Сообщения будут двигаться только по рёбрам кластеров. Требования 1) и 2) позволяют гарантировать достижимость узлов: из каждого узла сети по рёбрам кластеров можно попасть в каждый другой узел. Требование 2) позволяет гарантировать отсутствие дублирования сообщений, когда узел получает несколько одинаковых сообщений от одного и того же узла, при условии, что дублирование отсутствует внутри кластера. Это условие будет рассмотрено в разделах 3 и 5.

Терминальные узлы (узлы степени 1) графа кластеров и общих узлов являются кластерами, поскольку общий узел связан рёбрами, по крайней мере, с двумя кластерами, которым он принадлежит. Тем самым, для правильного набора кластеров все листья дерева кластеров и общих узлов являются кластерами.

Из 1) и 2) вытекает, что два кластера имеют не более одного общего узла.

Отсюда следует, что каждое ребро графа связей принадлежит не более чем одному кластеру, общих рёбер, принадлежащих нескольким кластерам, нет. Но мы не требуем, чтобы каждое ребро графа принадлежало какому-нибудь кластеру. В графе связей могут быть «лишние» рёбра, которые не принадлежат никаким кластерам; по таким рёбрам сообщения не передаются.

Утверждение 1: Любой связный граф является правильным, и можно гарантировать отсутствие дублирования сообщений в каждом кластере.

Доказательство. Выберем произвольный остов графа, а в качестве кластеров выберем все рёбра этого остова. Выполнение требований 1) и 2) очевидно. Отсутствие дублирования сообщений в кластере следует из того, что в кластере только одно ребро.

□

Это утверждение даёт теоретическую возможность реализации описываемой ниже схемы обмена сообщениями и, главным образом, опроса сети при любом связном графе связей.

Основные сообщения в сети бывают двух видов: запрос – **Request** и ответ (на запрос) – **Result**. Сообщение **Request** генерируется узлом, который будем называть *начальным узлом*, движется по некоторому пути в графе и принимается на обработку узлом, который будем называть *конечным узлом*. Сообщение **Result** движется в обратном направлении от конечного узла к начальному узлу. Запрос можно понимать как требование выполнить некую услугу (некие вычислительные действия) с параметрами, содержащимися в сообщении **Request**. Такое сообщение может быть предназначено не какому-то заранее известному узлу, а «кому-нибудь», кто может такой запрос обработать.

Обмен сообщениями **Request – Result** аналогичен удалённому вызову процедуры с той существенной разницей, что процесс в начальном узле, выдавший запрос, не приостанавливается до получения ответа.

В сообщении **Request** указывается уникальное имя запрашиваемой услуги и набор параметров, а каждый узел сети «знает», какие услуги он может оказывать. Ответное сообщение **Result** содержит ответные параметры и посылается уже известному начальному узлу.

Для поиска нужного конечного узла, которому будет направлено сообщение **Request**, используются служебные сообщения опроса: сообщение **Question**, содержащее имя запрашиваемой услуги, и ответные сообщения (ответы) трёх видов: **Yes**, **No** и **Wait**, а также сообщение отмены **Cancel**. Схема обмена этими сообщениями между начальным и конечным узлами показана на **Ошибка! Источник ссылки не найден.**

При поиске нужного конечного узла опрашивается, как правило, не вся сеть. Для того чтобы находить ближайший конечный узел и минимизировать время опроса, опрос происходит по раундам: на r -м раунде опрашиваются узлы на концах путей, начинающихся в начальном

узле, запрашивающем услугу, и проходящих через $r - 1$ промежуточных общих узлов. И только в том случае, когда на r -м раунде не найден нужный конечный узел, выполняется опрос на $r + 1$ -м раунде.

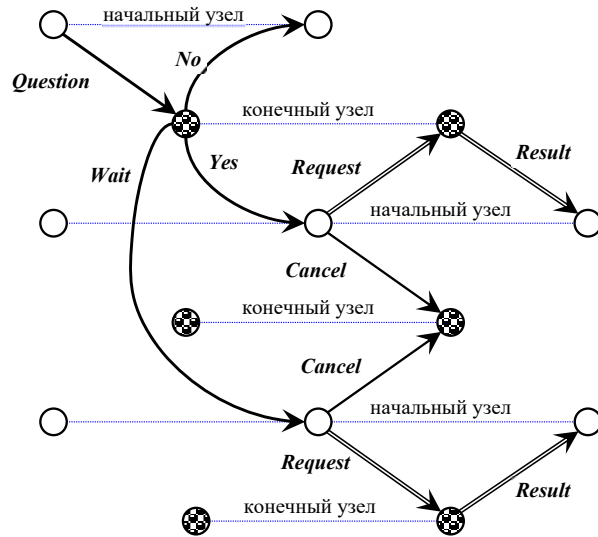


Рис. 1. Короткие (–) служебные и длинные (=) основные сообщения.
Fig. 1. Short (–) service and long (=) main messages.

Сообщение опроса **Question** содержит как параметр имя запрашиваемой услуги. Ответное сообщение **No** посылается узлом, если он не может оказать запрашиваемую услугу. Ответ **Yes** означает, что узел может оказать запрашиваемую услугу и готов сделать это немедленно. Если же узел может оказать запрашиваемую услугу, но только позже, поскольку сейчас «занят», он посылает ответ **Wait** с указанием предполагаемого времени ожидания.

Возможны три случая. 1) Если на раунде r все опрашиваемые узлы ответили сообщением **No**, начальный узел либо переходит к следующему раунду $r + 1$, если среди опрашиваемых узлов есть общий узел, либо завершает опрос с отрицательным результатом «в сети нет узла, который мог бы оказать запрашиваемую услугу». 2) Если хотя бы один узел ответил сообщением **Yes**, опрос завершается, и начальный узел посылает этому узлу сообщение **Request** и затем, после оказания услуги, получает от него сообщение **Result**. 3) Особый случай, когда в сети есть узлы, которые могли бы оказать запрашиваемую услугу, но все они сейчас «заняты». Тогда начальный узел выбирает узел, приславший сообщение **Wait** с минимальным временем ожидания, он становится получателем сообщения **Request** и, после оказания услуги, отправителем сообщения **Result**. В случаях 2 и 3 остальным узлам, приславшим ответ **Yes** (в случае 2) или ответ **Wait** (в случае 2 или 3), нужно отменить запрос, для этого начальный узел посылает им сообщение **Cancel**.

При опросе на раунде r между начальным узлом и опрашиваемым узлом находятся $r - 1$ промежуточных общих узлов, которые только пересылают сообщения **Question** и ответные сообщения **No**, **Yes** и **Wait**. Начальный узел на раунде $r = 1$ или последний промежуточный общий узел на раунде $r > 1$ (число промежуточных общих узлов между этим узлом и начальным узлом равно $r - 1$), получив сообщение **Question**, посылает его дальше всеобщей рассылкой (broadcast) всем узлам тех кластеров, в которые входит сам этот узел. Начальный

узел на раунде $r > 1$ или не последний промежуточный общий узел на раунде $r > 2$ (число промежуточных общих узлов между этим узлом и начальным узлом меньше $r - 1$), получив сообщение **Question**, тоже посылает сообщение **Question** дальше, но не всем узлам тех кластеров, в которые входит сам этот узел, а только общим узлам. Опрашиваемый узел (между ним и начальным узлом $r - 1$ промежуточных общих узлов) является конечным получателем сообщения **Question** на раунде r , сообщение **Question** дальше не посылается, а посылается ответное сообщение **No**, **Yes** или **Wait**.

Сообщение **Question**, проходя по пути графа от начального узла до конечного получателя, запоминает в проходимых им узлах путь до начального узла, по этому пути будет двигаться ответное сообщение **No**, **Yes** или **Wait**. Для запоминания используется динамически выделяемая память в узле, в которой создается, так называемое, описание сеанса. Когда ответное сообщение **Yes** или **Wait** движется к начальному узлу в описании сеанса в проходимых им узлах запоминается путь от начального узла до одного из узлов, пославших ответ **Yes**, а если таких нет, то до узла, приславшего ответ **Wait** с минимальным временем ожидания (если такой есть). После выбора начальным узлом конечного узла для обработки запроса (узла первым приславшего ответ **Yes** или, если таких ответов нет, узла приславшего ответ **Wait** с минимальным временем ожидания) по запомненному пути движется сообщение **Request**. После оказания услуги в обратном направлении движется сообщение **Result**.

В такой модели никакой предварительной настройки сети, связанной с распределением услуг по узлам сети, не требуется. Нужна только настройка сети для задания её топологии и, соответственно, перенастройка при изменении топологии.

В то же время дополнительные служебные сообщения опроса вводят дополнительные накладные расходы: увеличивается время выполнения запроса (на время поиска конечного узла) и увеличивается нагрузка на сеть. Предполагается, что это компенсируется тем, что служебные сообщения опроса – это *короткие* сообщения, тогда как основные сообщения (**Request** и **Result**) могут быть сколько угодно *длинными* за счёт того, что они содержат неограниченные по размеру параметры запроса услуги или ответные параметры, описывающие результат оказания услуги. В предлагаемой модели длинные основные сообщения движутся по кратчайшим путям и, кроме того, не происходит дублирования сообщений, когда узел получает одно и то же сообщение несколько раз по дублирующим путям.

При правильном наборе кластеров описанные правила обмена сообщениями гарантируют отсутствие дублирования сообщений во всех случаях, кроме сообщений **Question** внутри кластера. Ниже мы покажем, как можно гарантировать отсутствие дублирования внутри кластера для кластера-клик (раздел 3) и для кластера диаметра не более 2 (раздел 5). Более того, мы покажем, что в обоих случаях сообщение **Question** внутри кластера движется по кратчайшему пути, что гарантирует движение всех сообщений в сети по кратчайшим путям.

3. Кластеры-клики

Из утверждения 1 следует, что описанный в разделе 2 способ обмена сообщениями можно реализовать в любой сети со связным графом связей. Но понятно, что ожидаемый эффект высокой производительности может быть достигнут только тогда, когда граф относится к классу графов «мир тесен». Для этого кластер должен содержать всё-таки не 2, а как можно больше узлов, а число кластеров, покрывающих все узлы графа, достаточно мало.

В работе [1] рассматривался случай, когда кластер – это клика, т.е. подграф диаметра 1, в котором каждый узел соединён ребром с каждым другим узлом кластера. Клика может содержать любое число узлов. Для того чтобы гарантировать достижимость узлов и отсутствие дублирования (требования правильного набора кластеров), граф должен быть связным *блоковым графом* [6]-[7], то есть связным неориентированным графом, в котором каждый блок (компонента двусвязности) является кликой. Эти блоки-клики и объявляются

кластерами. Более точно, граф связей должен содержать суграф, являющийся блоковым графом, поскольку не все рёбра графа обязаны участвовать в передаче сообщений. Общий узел, т.е. узел, принадлежащий нескольким блокам, является *шарниром* в графе, и обратно: каждый шарнир является общим узлом нескольких блоков. Иными словами, общие узлы кластеров – шарниры. Дерево кластеров и общих узлов в этом случае является тем, что называется *деревом блоков и шарниров* (*block cutpoint graph*) [8]. В работе [1] есть утверждение 6: в любом связном графе есть суграф, являющийся блоковым графом. Доказательство этого утверждения аналогично доказательству утверждения 1 в настоящей статье.

Поскольку в клике любые два узла соединены ребром, легко гарантировать отсутствие дублирования сообщений внутри кластера. При всеобщей рассылке узел посылает сообщение **Question** либо всем соседним с ним узлам, либо всем соседним с ним общим узлам (шарнирам). Внутри кластера сообщение проходит путь длиной 1, дублирования нет. Более того, путь длиной 1 является, очевидно, кратчайшим путём.

Поскольку все кластеры являются кликами, и на раунде r между начальным узлом и опрашиваемым узлом находится $r - 1$ промежуточных общих узлов (шарниров), расстояние между этими узлами равно r .

На 0 приведён пример блокового графа с пятью блоками и пути опроса (красные стрелки) из начального состояния 0 на раундах 1-4. Чёрные кружки – это общие узлы (шарниры) кластеров-клик (блоков).

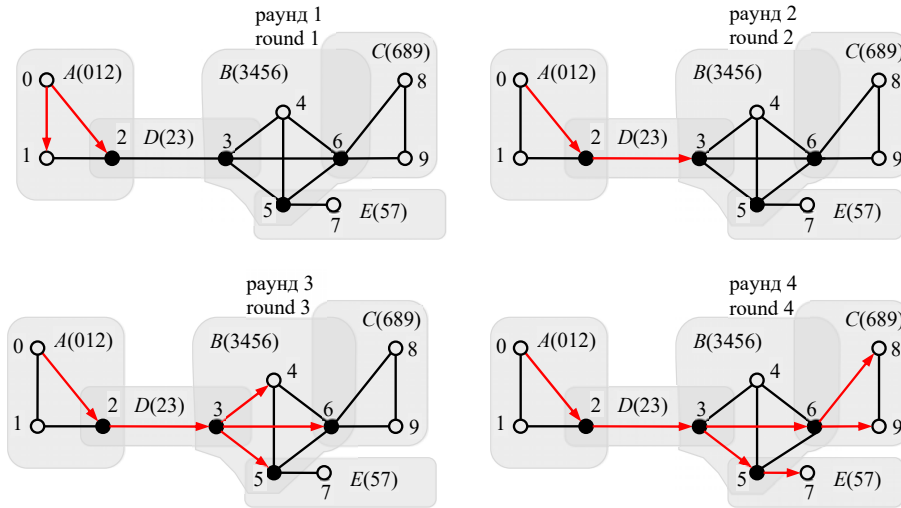


Рис. 2. Кластеры – клики. Пример блокового графа и опрос узлов в 4-х раундах.
Fig. 2. Clusters – cliques. An example of a block graph and four rounds of node polling.

4. Кластеры диаметра не более 2

Граф, ближайший к понятию «почти-клик», но не обязательно являющийся кликой, – это граф диаметра не более 2. Заметим, что переход от кластеров-клик к кластерам диаметром не больше 2 весьма существен. Говорят, что *почти все* графы обладают некоторым свойством, если отношение числа помеченных графов с n вершинами, имеющих это свойство, к числу всех помеченных графов с n вершинами стремится к 1 при $n \rightarrow \infty$. Число клик (полных графов) с n вершинами равно 1, поэтому почти все графы не являются кликами. В то же время

почти все графы имеют диаметр 2 [9]. В этом разделе мы исследуем способы построения правильного набора кластеров диаметра не более 2.

Поскольку граф блоков и шарниров для любого графа (не только блокового) является деревом, естественно, задаться вопросом: должен ли кластер диаметра не более 2 быть блоком, даже если блок не обязательно клика? Ответ отрицательный. На 0 приведён пример графа, который имеет диаметр 2 и может считаться одним кластером, но центральный узел является шарниром, т.е. этот кластер не блок. В этом примере блоки это рёбра.

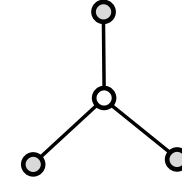


Рис. 3. Граф диаметра 2, в котором блоки это рёбра.
Fig. 3. A graph of diameter two for which blocks are edges.

В то же время в блоковом графе можно построить набор кластеров диаметра не более 2 с помощью следующей процедуры.

Процедура объединения блоков: Пусть граф является блоковым графом, т.е. каждый блок – это клика. Если узел a является общим узлом блоков $A_1, \dots, A_k$, то объединение этих блоков является подграфом диаметра не более 2. Будем объединять друг с другом блоки, которые ещё не входят ни в какое объединение, пока это возможно. После этого полученные объединения объявим новыми кластерами. Справедливо следующее утверждение.

Утверждение 2: Процедура объединения блоков строит правильный набор кластеров, каждый из которых имеет диаметр не более 2.

□

По утверждению 6 из работы [1] в любом связном графе есть суграф, являющийся блоковым графом. В то же время такой суграф выделяется в графе сети, вообще говоря, неоднозначно. На 0 приведён пример графа сети, который не является блоковым графом, и четыре способа выделения суграфа, являющегося блоковым графом.

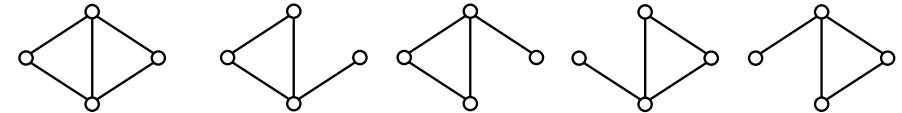


Рис. 4. Пример графа, который не является блоковым графом и разные способы выделения блокового суграфа.

Fig. 4. An example of a graph that is not a block graph and various ways for extracting a block subgraph.

Более того, процедура объединения блоков даже при выбранном суграфе, являющемся блоковым графом, определяет правильный набор кластеров диаметра не более 2, вообще говоря, неоднозначно. При различном порядке объединения блоков могут получиться разные правильные наборы кластеров диаметра не более 2. На 0 приведён пример блокового графа с тремя блоками A , B и C , и два способа объединения блоков: $A \cup B$ и $B \cup C$.

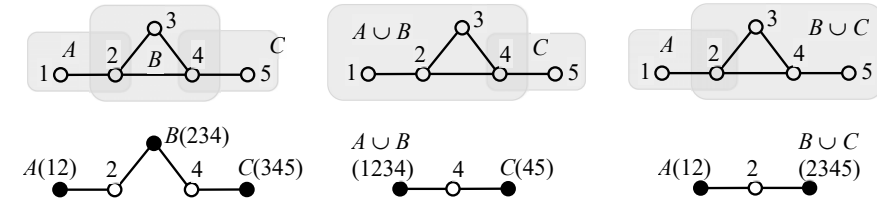


Рис. 5. Пример блочного графа с тремя блоками и два способа объединения блоков.
Fig. 5. An example of a block graph with three blocks and two ways of uniting blocks.

Теперь рассмотрим более общий способ построения правильного набора кластеров на основе множества максимальных подграфов степени 2.

Процедура разъединения максимальных подграфов диаметра не более 2: Выберем в графе связей G в качестве кластеров максимальные (по вложенности множества их узлов) подграфы диаметром не более 2. Рассмотрим граф кластеров и общих узлов. Если в нём есть простой цикл, выберем любое его ребро, соединяющее кластер A и узел a , и удалим узел a из кластера A , а из графа G удалим ребро, соединяющие узел a с другими узлами кластера A , то есть ребра вида ab , где узел b принадлежит кластеру A . Будем повторять эту процедуру, пока возможно, т.е. пока в графе кластеров и общих узлов есть цикл.

Справедливы следующие утверждения.

Утверждение 3: Процедура разъединения максимальных подграфов диаметра не более 2 строит правильный набор кластеров, каждый из которых имеет диаметр не более 2.

□

Утверждение 4: Дерево кластеров и общих узлов, получаемое процедурой разъединения максимальных подграфов диаметра не более 2 является остовом графа кластеров (как максимальных подграфов диаметра не более 2) и общих узлов исходного графа связей.

□

Как известно [10], число остовов графа определяется с помощью матрицы Кирхгофа, в которой на диагонали в клетке ii помещается степень узла i , в клетке ij , $i \neq j$, помещается «-1», если есть ребро ij , иначе – «0». Число остовов равно алгебраическому дополнению любого элемента матрицы Кирхгофа.

На рис 0 (вверху слева) приведён пример графа сети, две серые области A и B которого показывают его максимальные подграфы диаметром 2. Вверху в центре изображён соответствующий граф кластеров и общих узлов, когда кластеры – это максимальные подграфы диаметра не более 2. В этом графе имеются циклы. Для того чтобы удалить циклы применяется процедура разъединения, описанная выше. В конечном итоге выделяется набор кластеров, для которого граф кластеров и общих узлов является деревом. Это дерево является одним из остовов исходного графа кластеров (как максимальных подграфов диаметра не более 2) и общих узлов. Для данного примера матрица Кирхгофа приведена на 0 справа, число остовов равно 12. Все эти 12 остовов изображены на 0. Пунктиром изображены рёбра графа кластеров и общих узлов, удаляемые из этого графа для получения остова. Если удаляется ребро Xy , где X кластер, а y общий узел, то это означает, что узел y удаляется из кластера X , а в графе сети все рёбра, соединяющие узел y с другими узлами кластера X , не будут принадлежать кластерам.

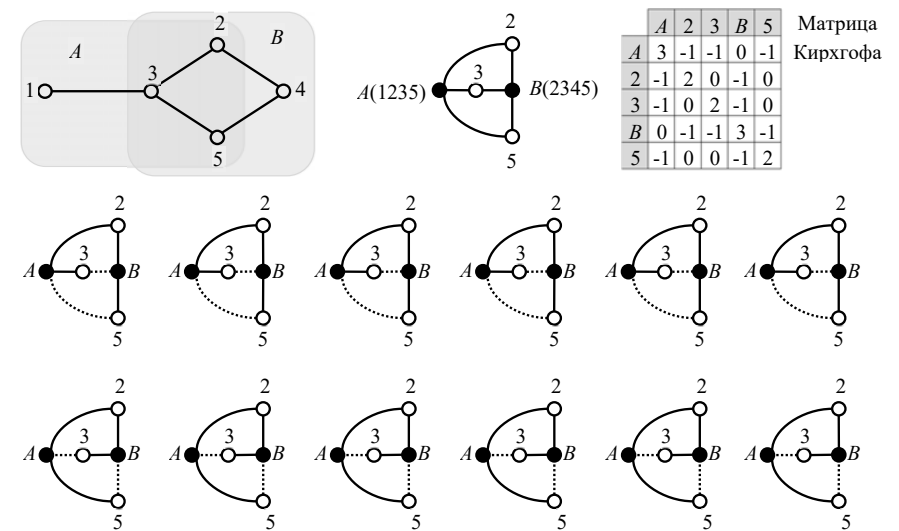


Рис. 6. Пример графа с двумя максимальными подграфами диаметра 2, соответствующий граф кластеров и общих узлов, и 12 способов выделения его остова.
Fig. 6. An example of a graph with two maximal subgraphs of diameter two, the corresponding graph of clusters and common nodes, and 12 ways for extracting its spanning tree.

Следует отметить, что правильный набор кластеров диаметра не более 2 может содержать число кластеров меньше числа максимальных подграфов диаметра не более 2. Это происходит в том случае, когда все узлы некоторого кластера являются общими узлами, т.е. принадлежат другим кластерам. На 0 (вверху слева) приведён пример графа сети, в котором выделены три максимальных подграфа диаметра 2: A , B и C . Вверху справа изображён граф кластеров и общих узлов, когда в качестве кластеров выбраны подграфы A , B и C . Этот граф циклический и имеет 105 различных остовов. Для одного из этих остовов правильный набор кластеров и соответствующий граф кластеров и общих узлов изображены в середине 0. Заметим, что все узлы подграфа B являются общими узлами: узел 2 принадлежит подграфу A , узел 4 принадлежит подграфу C , узлы 2 и 3 принадлежат обоим подграфам A и C . В результате в правильном наборе кластеров можно оставить только кластеры, полученные из подграфов A и B процедурой разъединения. На 0 (внизу) показан как раз такой случай, правильный набор кластеров содержит не три, а два кластера. Этот случай не входит в 105 случаев, основанных на выделении остова. В общем случае, этот и аналогичные случаи можно получить, допуская удаление тех кластеров, все узлы которых являются общими.

5. Коммутация сообщений опроса внутри кластера

Как сказано в разделе 2, коммутация всех сообщений основана на типе сообщения и для всех сообщений, кроме сообщения опроса **Question**, на описании пути, распределённом по узлам сети в динамически создаваемых описаниях сеанса. Только сообщение **Question** внутри кластера выполняется как всерная рассылка и нуждается в специальной коммутации.

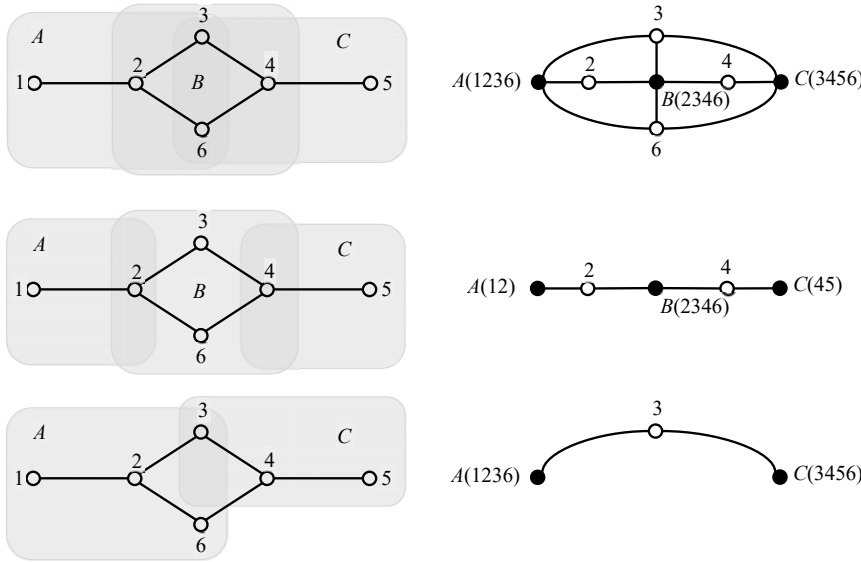


Рис. 7. Сверху граф с максимальным подграфом диаметра 2, состоящим из общих узлов.

В центре пример разъединения подграфов с помощью остова.

Внизу удаление подграфа, состоящего из общих узлов, и разделение оставшихся подграфов.

Fig. 7. Top: A graph in which a maximal subgraph of diameter 2 consists of common nodes.

Center: An example of subgraph separation using a spanning tree.

Bottom: Deletion of a subgraph consisting of common nodes and separation of the remaining subgraphs.

Такая коммутация определяется правилами коммутации сообщений, которые хранятся в каждом узле сети и устанавливаются при настройке сети. Правило коммутации в узле должно быть локальным, т.е. зависеть только от локального (ближайшего) окружения узла (а не от всей сети), типа сообщения и, возможно, некоторых его параметров с ограниченным множеством возможных значений. Это нужно для того, чтобы набор правил и их число было ограниченным и не зависело от глобальной сети, т.е. не требовало перенастройки при изменении топологии сети, не затрагивающей ближайшее окружение узла. В частности, в правило настройки не должны входить такие параметры, как идентификатор начального узла-отправителя или идентификатор конечного узла-получателя, поскольку число значений такого идентификатора не меньше числа узлов во всей сети. В целом правило коммутации в узле b имеет вид

$$(Tun_сообщения, значения_некоторых_параметров: a, b, c) \quad (1)$$

и означает, что когда узел b получает от соседнего узла a сообщение указанного типа с указанными значениями некоторых параметров, он должен переслать это сообщение соседнему узлу c .

Заметим, что обычно узел соседний с узлом b идентифицируется номером порта узла b , через который узел b получает сообщения от соседнего узла или отправляет ему сообщение. На графе связей это означает, что в узле b локально перенумерованы инцидентные ему ребра, причём ребро ab имеет, вообще говоря, разные номера в узле a и в узле b . При получении узлом b сообщения от соседнего узла a узлу b сообщается номер ребра ab в узле b , а при посылке сообщения из узла b в соседний узел c узел b указывает номер ребра bc в узле b . В

частности, так устроены правила коммутации в программно-конфигурируемых сетях (SDN – Software-Defined Networking).

Для кластеров-клик внутри кластера коммутация сообщений веерной рассылки, фактически, отсутствует, поскольку внутри кластера сообщение **Question** проходит только одно ребро, т.е. путь длины 1, на котором нет промежуточных узлов, в которых требовалась бы коммутация.

Для кластеров диаметра 2 или больше это уже не так: если внутри кластера сообщение **Question** проходит по ребру ab и должно далее направиться по ребру bc внутри кластера, то это определяется правилом коммутации (**Question**: a, b, c), которое означает: когда узел b получает сообщение **Question** от узла a , он должен дальше направить это сообщение узлу c . При веерной рассылке это сообщение, вообще говоря, должно клонироваться и направляться из узла b нескольким узлам $c_1, c_2, \dots, c_k$. Это означает наличие в узле b нескольких правил коммутации (**Question**: a, b, c_1), (**Question**: a, b, c_2), ..., (**Question**: a, b, c_k). Проблема в том, что в общем случае такая коммутация может порождать «лишние» пути перемещения сообщений. Это определяется правилом замыкания по дугам (arc closure) [11]: если есть пути $abcd$ и $a'bcd'$, то они порождают дополнительно пути $abcd'$ и $a'bcd$. Это показано на 0 (вверху), «лишние» пути показаны красным цветом, они возникают при прохождении двух путей $abcd$ и $a'bcd'$ через общую дугу bc .

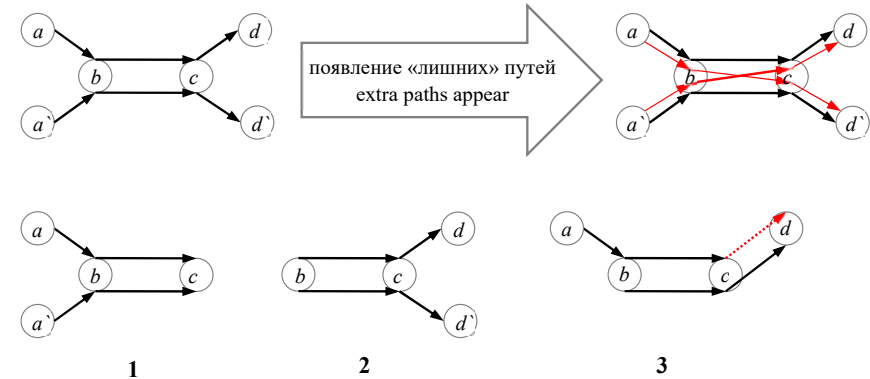


Рис. 8. Вверху: Замыкание по дугам множества путей длины ≥ 3 .

Внизу: при замыкании по дугам пути длины ≤ 2 не порождают новых путей длины ≤ 2 , но могут породить путь длины ≥ 3 (случай 3).

Fig. 8. Top: Arc closure of a set of paths of length ≥ 3 .

Bottom: in the arc closure procedure, paths of length ≤ 2 do not generate extra paths of length ≤ 2 , but can generate a path of length ≥ 3 (case 3).

Тем не менее, для кластеров диаметра не более 2 проблема «лишних» путей может быть решена. На 0 внизу показаны все три возможных случая 1, 2 и 3, когда два пути проходят по одной дуге bc . В случаях 1 и 2 лишних путей не возникает. В случае 3 порождается лишний путь длиной 3: $abcd$. Этот путь явно лишний, поскольку в кластерах диаметра не более 2 для достижимости всех его узлов из одного (любого) узла достаточно путей длины 1 и 2. Для решения проблемы предлагается различать первую и вторую дугу пути. Для этого в сообщение **Question** добавляется ещё один параметр: *номер_дуги*. Сначала сообщение **Question** посылается с параметром *номер_дуги* = 2. Узел, получивший сообщение **Question** с параметром *номер_дуги* = 2, не только обрабатывает его согласно описанным алгоритмам, но также согласно своим правилам коммутации посылает его дальше, но уже с параметром *номер_дуги* = 1. При получении сообщения **Question** с параметром *номер_дуги* = 1, оно

далее не посылается (правила коммутации не срабатывают). Это означает, что правило коммутации имеет вид

$$(Question, номер_дуги = 2: a, b, c). \quad (2)$$

Заметим, что если сообщение **Question** с параметром $номер_дуги = 2$ проходит по дуге ab , а в узле b нет правила коммутации вида (2), то это сообщение также далее не посылается из узла b .

Для того чтобы сгенерировать такие правила коммутации в узлах кластера диаметром не более 2, можно применить следующую процедуру. Для каждого узла a кластера выберем дерево T_a кратчайших путей, начинающихся в a и заканчивающихся в узлах кластера. Это дерево, очевидно, будет остовом кластера с высотой не более 2. Определим правила коммутации по дереву T_a , т.е. если в дереве есть путь abc , то в узле b определяем правило коммутации вида (2). Заметим, что правила коммутации определяются неоднозначным способом, поскольку зависят от выбора остова T_a для узла a для каждого узла кластера

Утверждение 5: Для кластера диаметром не больше 2 с описанными выше правилами коммутации сообщение **Question**, генерируемое (любым) узлом a кластера, проходит в кластере пути по дереву T_a , и попадает в каждый узел кластера ровно один раз.

Доказательство: Рассмотрим пары путей длины 2, проходящих по некоторой дуге bc . Пары путей $(abc, a'bc)$ и (abc, bcd) не порождают новых путей. Пара путей (abc, bcd) могла бы породить путь $abcd$ длиной 3, но этого не произойдет, поскольку после пути abc узел c получит сообщение **Question** с параметром $номер_дуги = 1$, поэтому не сработает правило коммутации $(Question, номер_дуги = 2: b, c, d)$, и сообщение не будет послано дальше в узел d .

Для кластеров диаметром больше 2 даже нумерация дуг пути, по которому проходит сообщение, не помогает: лишние пути могут возникать. А это приводит к дублированию сообщений, когда узел может получить несколько дублирующих друг друга сообщений **Question**. Это показано примером на 0 для графа диаметра 3. Синими стрелками нарисовано дерево кратчайших путей с корнем в узле a , а зелёными стрелками нарисовано дерево кратчайших путей с корнем в узле a' . Заметим, что оба дерева определяются однозначно. Пара путей $abcd'$ и $a'bcd$ порождают лишний путь $abcd$, дублирующий путь aed . Заметим также, что в каждом из путей $abcd'$ и $a'bcd$ общая дуга bc является второй дугой пути (нумерация дуг пути не помогает предотвратить дублирование путей).

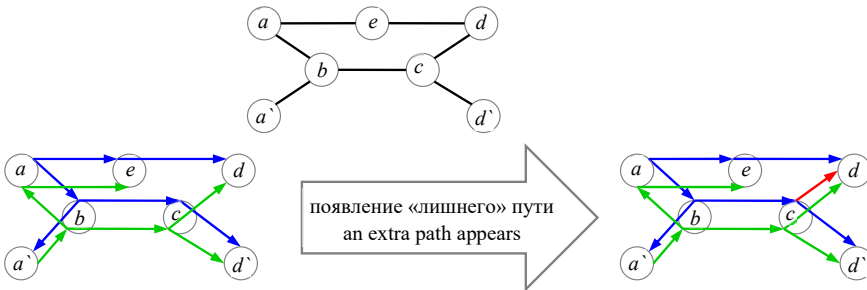


Рис. 9. Порождение лишнего пути в графе диаметра 3.
Fig. 9. Generating an extra path in a graph of diameter three.

Начальный узел и общий узел кластеров работают в двух режимах всеорной рассылки сообщения **Question**: либо всем узлам их кластера, либо только общим узлам этих кластеров. Поэтому на самом деле в узле сети должно быть два набора правил коммутации. Их можно

различить, введя два типа сообщений: **Question1** для всех узлов-получателей и **Question2** для общих узлов-получателей. Правила коммутации для сообщения, приходящего в узел b по дуге ab , имеют вид $(Question1, номер_дуги = 2: a, b, c_1), \dots, (Question1, номер_дуги = 2: a, b, c_k)$ и $(Question2, номер_дуги = 2: a, b, c'_1), \dots, (Question2, номер_дуги = 2: a, b, c'_k)$, где множества $\{c_1, c_2, \dots, c_k\}$ и $\{c'_1, c'_2, \dots, c'_k\}$, вообще говоря, разные.

0 иллюстрирует уменьшение числа кластеров и число раундов при переходе от кластеров-клик к кластерам диаметра не более 2 на одном и том же блоковом графе, идентичном графу на 0. 0 иллюстрирует случай, когда граф не является блоковым графом и, следовательно, кластеры не могут быть кликами. Этот граф получен из графа на 0 удалением трёх рёбер (показаны пунктиром), в результате из пяти клик на 0 три клики A , B и C перестают быть кликами (становятся подграфами диаметра 2). В обоих случаях на 0 и 0 по сравнению с 0 число кластеров уменьшается с 5 до 3, число раундов опроса из начального узла 0 уменьшается с 4 до 3. На обоих рисунках сверху слева показаны правила коммутации внутри кластеров сообщений **Question1** (одинарные синие стрелки) и **Question2** (двойные синие стрелки). Красными стрелками показано движение сообщений **Question1** (одинарные красные стрелки) и **Question2** (двойные красные стрелки). Рядом с красными стрелками показаны значения параметра $номер_дуги$ сообщений **Question1** и **Question2**.

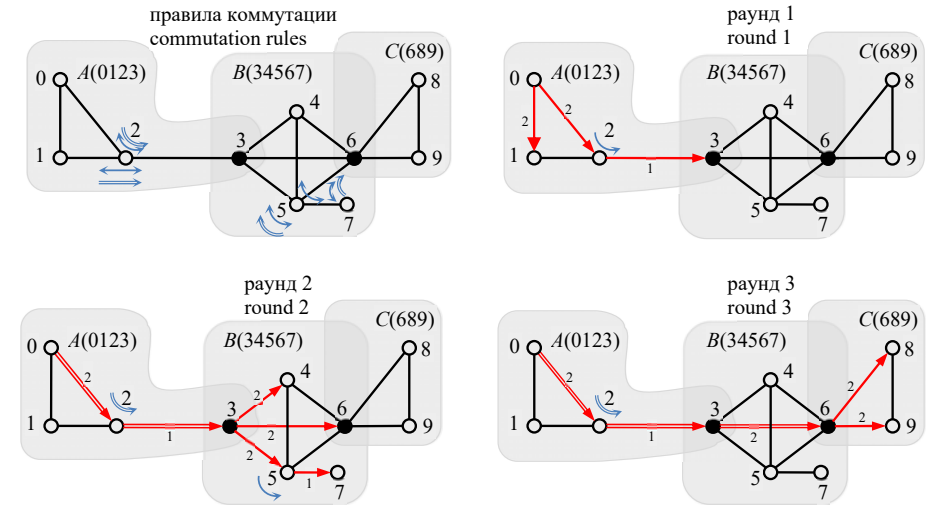


Рис. 10. Кластеры диаметра не более 2. Пример блокового графа и опрос узлов в 3-х раундах.
Fig. 10. Clusters of diameters at most two. An example of a block graph and three rounds of node polling.

6. Заключение

В статье предложено усовершенствование модели кластеризованной распределённой сети с опросом. Кластеризованность распределённой сети означает, что сеть состоит из множества кластеров, а каждый кластер это подмножество узлов сети и соединяющих их рёбер, причём два кластера имеют не более одного общего узла. Рёбра графа, которые не принадлежат ни одному кластеру, не используются для передачи сообщений, а переход из кластера в кластер осуществляется через общие узлы кластеров. Для того чтобы не было дублирования

сообщений, требуется, чтобы граф кластеров и их общих узлов, в котором ребро соединяет общий узел с кластером, которому этот узел принадлежит, был деревом.

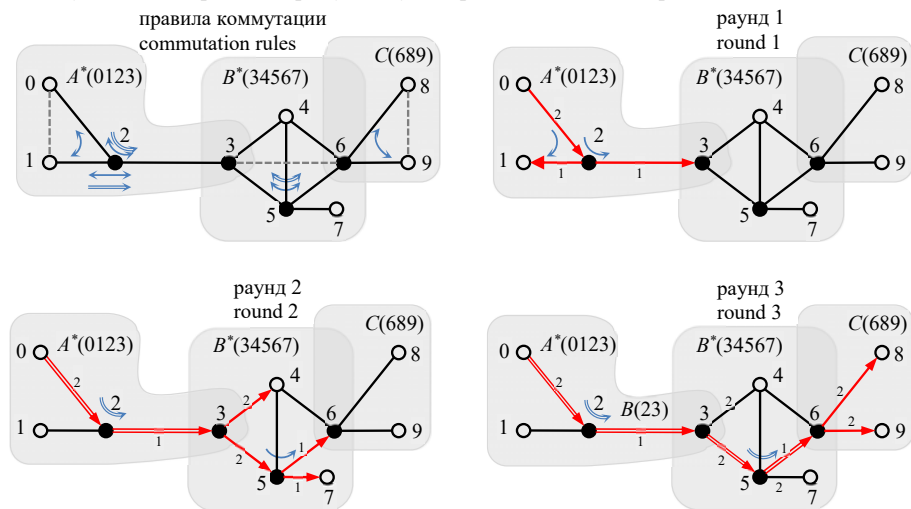


Рис. 11. Кластеры диаметра не более 2. Пример не блокового графа и опрос узлов в 3-х раундах.

Fig. 11. Clusters of diameters at most two.

An example of a non-block graph and three rounds of node polling.

Статья продолжает направление исследований, начатое авторами в работе [1], в которой кластер должен быть кликой, а в настоящей статье это требование существенно ослабляется: кластер может быть почти-кликкой, а именно, подграфом диаметра не больше 2.

Переход от кластеров-клик к кластерам диаметром не больше 2 весьма существенен, поскольку, во-первых, почти все графы имеют диаметр 2, во-вторых, число кластеров уменьшается примерно в 2 раза и также примерно в 2 раза уменьшается число требуемых раундов при пересылке сообщений, и, соответственно, данная работа имеет не только теоретическое, но и практическое значение.

В данной работе предложены два способа определения набора кластеров диаметром не больше 2, удовлетворяющих указанным требованиям: объединение блоков и разъединение максимальных подграфов диаметра не более 2. В дальнейшем было бы интересно провести исследование алгоритмов определения набора кластеров, учитывающего те или иные дополнительные (нефункциональные) параметры сети, т.е. для взвешенного графа связей. Кроме того, было бы интересно исследовать возможность использования тех рёбер графа связей, которые оказываются «лишними» при определении кластеров, но могли бы быть полезны для оптимизации передачи сообщений без дублирования сообщений.

Список литературы / References

- [1] Бурдонов И.Б., Евтушенко Н.В., Косачев А.С., Пономаренко В.Н. К организации системы запросов в распределенной сети с кластерами-кликками. Труды ИСП РАН, 2026, т. 38, вып. 3 (часть 1), стр. 7-32. DOI: 10.15514/ISPRAS-2026-38(3)-1. / Burdonov I.B., Yevtushenko N.V., Kossatchev A.S., Ponomarenko V.N. On the query system organization of a clustered distributed network. Trudy ISP RAN/Proc. ISP RAS, vol. 38, issue 3 (part 1), 2026, pp. 7-32 (in Russian). DOI: 10.15514/ISPRAS-2026-38(3)-1.

- [2] Евин И.А. Введение в теорию сложных сетей. Компьютерные исследования и моделирование, 2010, т. 2, вып. 2, стр. 121-141, Доступно по ссылке: <https://crm-en.ics.org.ru/uploads/crmissues/crm2010-2-2/crm10201.pdf>, обращение 01.04.2026.
- [3] Мир тесен (граф). Доступно по ссылке: [https://ru.wikipedia.org/wiki/Мир_тесен_\(граф\)](https://ru.wikipedia.org/wiki/Мир_тесен_(граф)), обращение 01.04.2026.
- [4] Milgram S. The Small World Problem. Psychology Today, 1967. Available at: <https://snap.stanford.edu/class/cs224w-readings/milgram67smallworld.pdf>, accessed 01.04.2026.
- [5] Мир тесен. Доступно по ссылке: https://ru.wikipedia.org/wiki/Мир_тесен, обращение 01.04.2026.
- [6] Kristina Vušković. Even-hole-free graphs: A survey. Applicable Analysis and Discrete Mathematics, 2010, vol. 4, no. 2, pp. 219-240. DOI: 10.2298/AADM100812027V, <https://doiserbia.nb.rs/img/doi/1452-8630/2010/1452-86301000027V.pdf>, обращение 01.04.2026.
- [7] Блоковый граф. Доступно по ссылке: https://ru.wikipedia.org/wiki/Блоковый_граф, обращение 01.04.2026.
- [8] Карпов Д.В. Теория графов. МЦНМО, 2022. ISBN 978-5-4439-1690-3. Доступно по ссылке: <https://www.klex.ru/2ad3>, обращение 01.04.2026.
- [9] Алексеев В.Е., Захарова Д.В. Теория графов. Учебное пособие. Нижний Новгород: Нижегородский госуниверситет, 2017, 119 с. Доступно по ссылке: http://www.unn.ru/books/met_files/Theory_graph.pdf, обращение 01.04.2026.
- [10] Асанов М.О., Баранский В.А., Расин В.В. Дискретная математика: графы матроиды, алгоритмы. Ижевск: НИЦ "РХД", 2001, 288 стр. Доступно по ссылке: <https://creewick.github.io/study/courses/graphs/book.pdf>, обращение 01.04.2026.
- [11] Бурдонов И.Б., Евтушенко Н.В., Косачев А.С. Безопасная реализация виртуальной сети на плоскости данных SDN. Труды ИСП РАН, том 33, вып. 1, 2021 г., стр. 123-136. DOI: 10.15514/ISPRAS-2021-33(1)-9, Доступно по ссылке: https://ispranproceedings.elpub.ru/jour/article/view/1378?sphrase_id=7259394, обращение 01.04.2026.

Информация об авторах / Information about authors

Игорь Борисович БУРДОНОВ – доктор физико-математических наук, главный научный сотрудник ИСП РАН. Научные интересы: формальные спецификации, генерация тестов, технология компиляции, системы реального времени, операционные системы, объектно-ориентированное программирование, сетевые протоколы, процессы разработки программного обеспечения.

Igor Borisovich BURDONOV – Dr. Sci. (Phys.-Math.), a Leading Researcher of ISP RAS. Research interests: formal specifications, test generation, compilation technology, real-time systems, operating systems, object-oriented programming, network protocols, software development processes.

Нина Владимировна ЕВТУШЕНКО, доктор технических наук, профессор, главный научный сотрудник ИСП РАН, до 1991 года работала научным сотрудником в Сибирском физико-техническом институте. С 1991 г. работала в ТГУ профессором, зав. кафедрой, зав. лабораторией по компьютерным наукам. Её исследовательские интересы включают формальные методы, теорию автоматов, распределённые системы, протоколы и тестирование программного обеспечения.

Nina Vladimirovna YEVTUSHENKO, Dr. Sci. (Tech.), Professor, a Leading Researcher of ISP RAS, worked at the Siberian Scientific Institute of Physics and Technology as a researcher up to 1991. In 1991, she joined Tomsk State University as a professor and then worked as the chair head and the head of Computer Science laboratory. Her research interests include formal methods, automata theory, distributed systems, protocol and software testing.

Александр Сергеевич КОСАЧЕВ – кандидат физико-математических наук, ведущий научный сотрудник ИСП РАН. Научные интересы: формальные спецификации, генерация тестов, технология компиляции, системы реального времени, операционные системы, объектно-

ориентированное программирование, сетевые протоколы, процессы разработки программного обеспечения.

Alexander Sergeevitch KOSSATCHEV – Cand. Sci. (Phys.-Math.), a Leading Researcher of ISP RAS. Research interests: formal specifications, test generation, compilation technology, real-time systems, operating systems, object-oriented programming, network protocols, software development processes.

Вера Николаевна ПОНОМАРЕНКО – кандидат физико-математических наук, старший научный сотрудник ИСП РАН. Научные интересы: формальные спецификации, генерация тестов, системы реального времени, операционные системы, объектно-ориентированное программирование, сетевые протоколы, процессы разработки программного обеспечения.

Vera Nikolaevna PONOMARENKO – Cand. Sci. (Phys.-Math.), a Senior Researcher of ISP RAS. Research interests: formal specifications, test generation, real-time systems, operating systems, object-oriented programming, network protocols, software development processes.