УДК 519.175.1
DOI: 10.18799/29495407/2023/2/26

КОМПАКТНЫЕ РАЗБИЕНИЯ НА ТОПОЛОГИЧЕСКИХ ГРАФАХ БОЛЬШОЙ РАЗМЕРНОСТИ

Погребной Александр Владимирович[1],
pogrebnoy@tpu.ru

Погребной Андрей Владимирович[2],
avpogrebnoy@gmail.com

[1] Национальный исследовательский Томский политехнический университет,
Россия, 634050, г. Томск, пр. Ленина, 30.

[2] Общество с ограниченной ответственностью «Дром»,
Россия, 634061, г. Томск, ул. Лебедева, 16.

Актуальность. *Распределенные системы, содержащие сотни и тысячи объектов, как правило, строятся в виде иерархических структур. В этих структурах объекты нижнего уровня объединяются в подмножества для подключения к соответствующим центрам. Существующие алгоритмы не способны успешно решать задачи структуризации на множествах такой размерности. Поэтому необходимы новые алгоритмы, пригодные для решения задач структуризации на множествах, содержащих тысячи объектов.* ***Цель:*** *разработка алгоритма формирования компактного разбиения на множествах большой размерности, содержащих до тысячи объектов, расположенных на заданной территории.* ***Методы:*** *прикладная теория графов, методы линейного программирования, построения и анализа эффективности алгоритмов, теория компактных разбиений, компактных множеств объектов и их скоплений.* ***Результаты.*** *Территориальное расположение множества объектов распределенной системы предлагается представлять в виде топологического графа. Для повышения эффективности работы алгоритма формирования компактных множеств и выделения скоплений вводится понятие зоны активного поиска ближайших вершин. Это дает возможность матрицу расстояний между вершинами графа заменить списком инциденторов вершин, сформированных на основе зоны активного поиска. Разработан алгоритм приближенного решения задачи компактного разбиения множества объектов топологического графа, представленного списком инциденторов вершин, на заданное число подмножеств. Алгоритм для каждого объекта рекурентным образом наращивает мощность компактных множеств, анализирует образовавшиеся скопления и при определенных условиях переходит к формированию компактного разбиения. Задача формирования подмножеств компактного разбиения на основе скоплений формируется как задача линейного программирования транспортного типа. Изложение алгоритма сопровождается примером.*

Ключевые слова: *Компактное разбиение, компактное множество, скопление объектов, плотность скопления, топологический граф.*

Введение

Современные территориально-распределенные компьютерные системы могут объединять большие множества объектов различной природы (оборудование объектов управления с датчиками и приемниками информации [1], узлы компьютерных сетей [2], интеллектуальные терминалы систем мониторинга [3], моты сенсорных сетей [4] и т. п.). Необходимость объединения большого множества объектов предполагает построение системы в виде иерархической структуры. В этой структуре объекты нижнего уровня объединяются в подмножества для подключения к соответствующим центрам. В подмножествах центры размещаются так, чтобы минимизировать суммарные расстояния от центра до объектов. Такие оценки отражают компактность расположения объектов подмножества на территории, охватываемой системой. Задача разбиения множества объектов системы на подмножества по критерию минимальной суммы оценок компактности подмножеств получила название компактного разбиения [5].

В работах [5, 6] предложены эффективные алгоритмы приближенного решения данной задачи для различных условий и требований к компактным разбиениям. В качестве модели топологии размещения множества объектов на территории, названной топологическим полем, принят топологический граф. Вершины графа на топологическом поле представлены координатами соответствующих объектов, а веса ребер равны расстояниям между объектами. Алгоритмы реализуют два подхода к поиску компактного разбиения. Первый из них основан на применении метода последовательного улучшения компактности разбиения [5]. Метод включает многократное решение задачи линейного программирования транспортного типа. Второй подход основан на выделении компактных скоплений объектов, которые используются для получения компактного разбиения, удовлетворяющего заданным ограничениям [6].

Экспериментальные исследования данных алгоритмов показали, что они успешно работают на топологических графах, содержащих до ста вершин. Повысить эффективность алгоритмов можно за счет улучшения алгоритмизации и программной реализации. Но для решения данной задачи на множествах в сотни и тысячи вершин необходимы специальные алгоритмы, ориентированные на такие размерности. При разработке таких алгоритмов выделяются два направления исследований. Первое основывается на разработке новых правил выделения скоплений объектов и преобразования карты скоплений в компактное разбиение. Второе направление связано с применением правил понижения размерности топологического графа до приемлемой величины, например, путем наложения на топологическое поле координатной сетки. В этом случае объекты, попавшие в ячейки сетки, объединяются в одну вершину. Данная статья посвящена исследованиям по первому направлению.

Постановка задачи

Топологический граф представим в виде $TG=(E,R)$, где $E=\{e_i\}, i=1,2,\ldots,n$ – множество вершин (объектов на топологическом поле), $R=\|r_{ij}\|$ матрица расстояний r_{ij} между вершинами e_i и e_j. Задача заключается в разбиении множества E на совокупность подмножеств $\{E_k\}$, $k=1,2,\ldots,K$ по критерию:

$$\sum_{k=1}^{K} \sum_{e_i, e_j \in E_k} r_{ij} \Rightarrow \min; \qquad (1)$$

при соблюдении условий:

$$\bigcup E_k = E; \; E_{k_1} \bigcap E_{k_2} = \varnothing; \; \alpha \le |E_k| \le \beta. \qquad (2)$$

Интервал $[\alpha,\beta]$ определяет допустимую мощность подмножеств E_k.

Представление графа TG матрицей связности R для применения непосредственно при решении задачи (1), (2) является избыточным и неудобным. Избыточность обусловлена тем, что согласно критерию (1), вершины e_i и e_j, удаленные друг от друга на большое расстояние, не попадут в одно и то же подмножество E_k. В этом случае при принятии решения о включении e_i или e_j в E_k соответствующий элемент r_{ij} окажется невостребованным. Таким образом, в каждой i-й строке матрицы R элементы с относительно большим значением r_{ij} могут быть удалены. Граничная величина такого значения, обозначим ее Δr, для которой все элементы $r_{ij}>\Delta r$ удаляются, зависит от размеров топологического поля и числа K подмножеств E_k, формируемых на множестве E. Величину Δr относительно каждой e_i следует рассматривать как наибольшее удаление от вершины e_i при поиске вершин e_j, которые наряду с e_i могут претендовать на включение в E_k.

Для определения величины Δr топологическое поле условно разбивается на K областей. Внутри области зона активного поиска вершин e_j, ближайших к вершине e_i, не должна быть меньше величины Δr. Исходя из этого нетрудно подсчитать значение величины Δr:

$$\Delta r = \varphi \bullet L_x / (K \bullet L_x / L_y)^{1/2}. \qquad (3)$$

Здесь φ – коэффициент расширения зоны активного поиска ближайших вершин; L_x, L_y – размеры топологического поля по координатам x и y. Значение K определяется по числу вершин в множестве E и среднему значению мощности E_k, заданному интервалом $[\alpha,\beta]$, $K=2\cdot n/(\alpha+\beta)$. Согласно (3), при $L_x=680$, $L_y=440$, $K=12$, $\varphi=1{,}2$ величина $\Delta r=190$. С увеличением K до 20 Δr уменьшается до 146.

После определения величины Δr все элементы $r_{ij}>\Delta r$ исключаются из матрицы R. В этом случае граф TG перестает быть полным, а каждая вершина e_i связывается ребром с другой вершиной e_j, если $r_{ij}\le\Delta r$. Новый граф будем обозначать TG^*, а его матрицу – R^*. Для удобства работы с графом TG^* представим его в виде списка инциденторов $F(e_i)$ вершин e_i. Множество вершин инцидентора $F(e_i)$ содержит вершины e_j инцидентные ребрам r_{ij}. Инцидентор $F(e_i)$ представляет все вершины e_j, попадающие в зону поиска для вершины e_i, определяемую величиной Δr. Приведенные выше преобразования существенно упрощают представление графа TG и создают условия для разработки эффективных процедур поиска компактных разбиений.

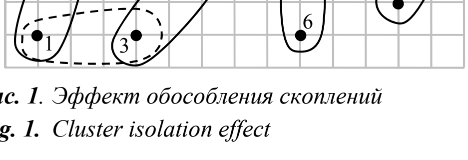
Формирование компактных множеств и скоплений объектов

При разработке алгоритма решения задачи (1), (2), отличающейся большой размерностью, рассчитывать на применение методов оптимизации не приходится. В данном случае даже применение задачи линейного программирования транспортного типа, с успехом используемой в методе [5], оказывается неприемлемым. Поэтому предлагаемый алгоритм основан на использовании естественной неравномерности расположения объектов на топологическом поле. Заметим, что при абсолютной равномерности расположения объектов решение задачи компактного разбиения должно строится на другой основе.

Наличие неравномерности предполагает существование скоплений подмножеств объектов. Важно научиться выделять эти скопления и использовать их для формирования компактных разбиений. Для выделения скоплений предлагается алгоритм последовательного наращивания мощности компактных множеств (KM) и анализа полученного состава скоплений. При формировании KM скопления на топологическом поле (ТП) выделяются автоматически. Здесь работает эффект обособления скоплений, формирующихся из совокупностей KM. Состав и план расположения скоплений на ТП, выделенных на основе KM мощностью g, $g=2,3,\ldots$, будем именовать топологией скоплений уровня g и обозначать $T(g)$. Появление эффекта обособления скоплений для TG, содержащего 8 объектов, показано на рис. 1. Формирование KM здесь осуществлялось для $g=2$ (рис. 1, *а*) и $g=3$ (рис. 1, *б*). Из рис. 1 следует, что с помощью KM объекты TG оказались разбиты на два скопления.

Пример на рис. 1 будем также использовать при изложении алгоритма формирования KM и анализа $T(g)$. Напомним, что граф TG представляется списком инциденторов $F(e_i)$ вершин e_i. В $F(e_i)$ вершины e_j включаются согласно Δr и упорядочиваются по возрастанию значений r_{ij}. В частности, для примера на рис. 1 при $L_x=14$, $L_y=5$, $K=2$, $\varphi=1{,}2$, величина Δr вычисленная по выражению (3), примерно равна 7. Это означает, что $F(e_1)=\{e_3,e_2,e_4\}$, а, например, $F(e_5)=\{e_6,e_4,e_7,e_8,e_3\}$.

а) $g=2$



б) $g=3$

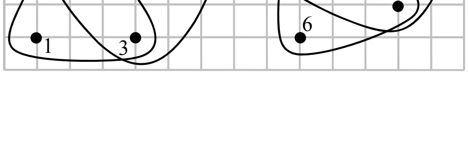


***Рис. 1**. Эффект обособления скоплений*
***Fig. 1.** Cluster isolation effect*

Алгоритм формирования *KM* начинает работу с минимального $g=2$. Для всех $e_i \in E$ в соответствующих $F(e_i)$ выбирается первая (ближайшая к e_i) вершина e_j. Множество вершин $\{e_i,e_j\}$ принимается в качестве *KM* и включается в список $KM(g)$, в данном случае $KM(2)$. Если в списке $KM(2)$ некоторые множества вершин совпадают, то сохраняется одно из них с пометкой d, равной числу совпадений. На рис. 1, *а* множества $\{e_3,e_4\}$ и $\{e_4,e_3\}$ совпадают, и в $KM(2)$ запоминается множество $2\{e_3,e_4\}$. Аналогично, в $KM(2)$ войдут $2\{e_5,e_6\}$ и $2\{e_7,e_8\}$. Множества $\{e_1,e_2\}$ и $\{e_1,e_3\}$ в списке встречаются один раз, и для них по умолчанию $d=1$. Сумма значений d для всех множеств списка $KM(2)$ равна $n(1)=8$.

Всего в список $KM(2)$ вошли пять *KM*. На рис. 1, *а* они обведены линиями. *KM* из списка $KM(2)$ расположились на ТП (рис. 1, *а*) так, что привели к обособлению трех скоплений. На рис. 1, *а* скопление, состоящее из трех *KM*, отличается от двух других тем, что содержит множество $\{e_1,e_3\}$, объекты которого уже принадлежат множествам $\{e_1,e_2\}$ и $\{e_3,e_4\}$. Такие *KM* именуются дублирующими и исключаются из списка $KM(2)$. Множество $\{e_1,e_3\}$ на рис. 1, *а* отмечено пунктирной линией. Если скопление содержит несколько дублирующих *KM*, то первым исключается *KM* с наиболее низкой оценкой компактности. Далее уточняется состав дублирующих *KM*, и исключение повторяется.

После анализа всех скоплений и дублирующих *KM* топология $T(g)$ подлежит анализу для принятия решения о необходимости наращивания мощности *KM* и формирования списка $KM(g+1)$. При анализе воспользуемся вполне очевидным положением о том, что компактность множества всегда больше суммарной компактности его подмножеств. Алгоритм, напротив, последовательно наращивает мощность *KM*. В качестве исходных *KM* выступают вершины графа *TG*, т. е. $g=1$. Наращивание мощности *KM* для $g=2,3,\ldots$ приводит к снижению компактности. Постепенно, увеличивая g на 1, алгоритм должен анализировать очередное $T(g)$ и уловить такое значение g^*, при котором наращивание мощности становится нецелесообразным.

При анализе $T(g)$ необходимо учитывать параметры скоплений, их состав и план расположения на топологическом поле, исходные требования к компактному разбиению. Многократное выполнение такого анализа является трудоемким делом. Проведение анализа в полном объеме оправдано на заключительном этапе алгоритма при формировании компактного разбиения. Поэтому при наращивании мощности *KM* анализ $T(g)$ ограничивается получением оценки $\delta(g)$, которая опосредованно отражает динамику процесса обособления скоплений.

Оценка $\delta(g)$ характеризует снижение числа *KM* в списке $KM(g)$ по отношению к списку $KM(g–1)$. Число *KM* в списке $KM(g)$ обозначим величиной $n(g)$.Оценку снижения $\delta(g)$ определим в виде отношения $(n(g–1)–n(g))/n(g–1)$. Для примера на рис. 1, *а* оценка $\delta(2)=(n(1)–n(2))/n(1)=(8–4)/8=1/2$. Чем больше оценка снижения $\delta(g)$, тем сильнее проявляется эффект обособления скоплений. Приближение оценки $\delta(g)$ к нулю является сигналом того, что процесс обособления стабилизировался и может ускориться лишь через несколько переходов от g к $g+1$. В общем случае переход к формированию списка $KM(g+1)$ на основе $KM(g)$ осуществляется при соблюдении следующих условий:

- число *KM* в списке $KM(g)$ больше рекомендованного в задании значения K, т. е. $n(g)>K$;
- значение g не достигло значения α в рекомендованном интервале $[\alpha, \beta]$, т. е. $g<\alpha$;
- при переходе от $g–1$ к g оценка снижения числа *KM* оказалась больше 0, т. е. $\delta(g)>0$.

При соблюдении всех перечисленных условий алгоритм продолжает наращивание мощности *KM* путем перехода от g к $g+1$. Работа алгоритма при несоблюдении одного или нескольких условий будет рассмотрена в следующем разделе статьи.

Формирование списка $KM(g+1)$ выполняется путем наращивания на одну вершину мощности *KM* из списка $KM(g)$. С этой целью для каждого t-го множества $KM_t=\{e_{i_1},e_{i_2},\ldots,e_{i_g}\}$ из списка $KM(g)$ определяется множество вершин $\{e_j\}_t$, которые одновременно связаны со всеми вершинами KM_t, т. е. $\{e_j\}_t=(F(e_{i_1})\cap F(e_{i_2})\cap\ldots\cap F(e_{i_g}))\setminus KM_t$. Среди вершин $\{e_j\}_t$ выбирается вершина e_j^*, для которой сумма расстояний $r(e_{i_1},e_j^*)+r(e_{i_2},e_j^*)+\ldots+r(e_{i_g},e_j^*)$ является наименьшей. Вершина e_j^* включатся в множество KM_t, которое заносится в список $KM(g+1)$. Если в списке $KM(g+1)$ встречаются совпадающие множества вершин, то сохраняемое множество помечается соответствующей величиной d.

В примере на рис. 1, *б* показаны 4 *KM* списка $KM(3)$, полученные на основе наращивания мощности *KM* из списка $KM(2)$. Множество $\{e_1, e_2\}$ подключило вершину e_3, а множество $\{e_3, e_4\}$ – вершину e_2. В итоге в списке $KM(3)$ множество $\{e_1, e_2, e_3\}$ будет помечено величиной $d=1$. При этом сумма значений d в списке $KM(3)$ будет равна $n(2)=4$. Оценка снижения $\delta(3)$ в этом случае составит $(n(2)–n(3))/n(2)=(4–4)/4=0$. Нетрудно убедиться, что при переходе к $g=4$ в списке $KM(4)$ будет две записи, а оценка $\delta(4)=1/2$. Что касается перехода к $g=5$, то здесь число *KM* в списке $KM(5)$ сохраниться на уровне $KM(4)$, а оценка $\delta(5)=0$. Это означает, что процесс наращивания мощности *KM* следует завершить для $g=4$.

Анализ скоплений и формирование компактных разбиений

Топология скоплений $T(g)$ представляется для анализа в случае невыполнения любого из названых выше условий. Каждое из них проявляется на $T(g)$ в виде определенной специфики, которая предопределяет особенности анализа и принятия решений по формированию компактных разбиений.

Первое условие соответствует ситуации, когда $n(g)\leq K$. Очевидно, что такая ситуация вполне возможна, т. к. процесс наращивания мощности *KM* в конечном итоге приводит к $n(g)=1$. Если число *KM* в списке $KM(g)$ равно K или близко к нему, а значение g находится в пределах интервала $[\alpha, \beta]$, то можно считать, что задача компактного разбиения решена. Остается лишь перераспределить объекты, которые одновременно принадлежат нескольким *KM*. Эта задача будет рассмотрена ниже на заключительном этапе алгоритма.

При анализе $T(g)$ в условиях $g=\alpha$ предполагается, что $n(g)$ значительно превышает значение K. Такая ситуация возникает при достаточно равномерном расположении объектов на *ТП* и, соответственно, слабом проявлении эффекта обособления скоплений. В этом случае алгоритм продолжает наращивать мощность *КМ*, в том числе и при $g>\beta$, до формирования списка $KM(g)$ с $n(g)=K$. После этого решается задача перераспределения объектов между *КМ*.

Каждый шаг наращивания мощности *КМ*, пока значение g остается в интервале $[\alpha, \beta]$, сопровождается выполнением анализа скоплений. Выделение скоплений производится на основе списка $KM(g)$. Очередное s-е скопление формируется из *КМ*, связанных между собой через наличие пересечений. Введем оценку плотности скопления и обозначим ее $P(g)_s$

$$P(g)_s = (d_s \cdot g - n_s) / n(g)_s \cdot g, \qquad (4)$$

где n_s – число объектов в s-м скоплении; d_s – сумма значений d, приписанных к *КМ*, составивших s-е скопление; $n(g)_s$ – число *КМ*, образующих s-е скопление.

Согласно (4) величина оценки плотности $P(g)_s \le 1$. Максимальная плотность $P(g)_s=1$ соответствует ситуации, когда скопление образовано одним *КМ*. Такое скопление принимается в качестве подмножества компактного разбиения, а его объекты не учитываются в последующей работе алгоритма. Скопления, для которых $P(g)_s<1$, а число n_s укладывается в интервал $[\alpha,\beta]$, также могут быть приняты в качестве подмножеств, т. к. при последующих переходах от g к $g+1$ они окажутся представлены одним *КМ*. Примеры таких скоплений представлены на рис. 1, *б*. Плотность каждого скопления составит $(2\cdot3–4)/2\cdot3=1/3$. Если для данного примера принять $K=2$, $[\alpha, \beta]=[3, 5]$, то, несмотря на невысокое значение плотности, оба скопления принимаются в качестве подмножеств.

Невыполнение третьего условия, когда оценка снижения числа *КМ* равна нулю, $\delta(g)=0$, а первые два условия соблюдаются, является сигналом того, что такие оценки могут сохраняться на ряде последующих шагов наращивания мощности *КМ*. Например, для $T(g)$ на рис. 1, *б* оценка $\delta(g)=0$ будет сохраняться при переходах к $g=5, 6, 7$. В этой ситуации решение о продолжении работы алгоритма принимается в зависимости от исходных требований к решению задачи компактного разбиения. Если значения K и интервала $[\alpha, \beta]$ носили рекомендательный характер, то они могут быть скорректированы. После этого алгоритм может продолжить работу по анализу первого и второго условия. Если изменение параметров K и $[\alpha,\beta]$ недопустимо, то алгоритм продолжает наращивать мощность *КМ*.

Рассматривая ситуации, возникающие при несоблюдении одного или нескольких условий, в конечном итоге принимаются решения о продолжении работы алгоритма по формированию числа *КМ*, близкого или равного K. При этом допускается превышение значения g над величиной β. Чем больше это превышение, тем больше число объектов принадлежит пересечениям *КМ*. Следовательно, большее число объектов потребуется перераспределить между *КМ* так, чтобы значения их мощности были в пределах интервала $[\alpha,\beta]$.

Очевидно, что ситуацию, когда $n(g)=K$, которая может сложиться в ходе наращивания мощности *КМ* и сокращения списка $KM(g)$, следует рассматривать как редкий и удачный случай. Как правило, работа алгоритма завершается при достижении наибольшего приближения к K. Если, например, при $K=20$ алгоритм достигнет снижения *КМ* до $n(g)=21$, то согласно первому условию, $21>K$, будет продолжен переход от g к $g+1$. При этом может оказаться, что $n(g+1)\le18$. В этом случае необходимо принять решение о формировании компактного разбиения на основе списка $KM(g)$ или $KM(g+1)$, т. к. алгоритм не имеет возможности перестроиться на формирование списка с $n(g)=20$. Возможность расформирования отдельных *КМ* так, чтобы в компактном разбиении получилось ровно K подмножеств, здесь не рассматривается.

Задача перераспределения объектов между *КМ*

На заключительном этапе алгоритма в каждом *КМ* из списка $KM(g)$ уточняется состав объектов, которые образуют соответствующее подмножество компактного разбиения. Уточнение касается тех объектов, которые входят в состав двух или более *КМ*. Необходимо решить, к каким *КМ* отнести данные объекты, чтобы получить компактное разбиение. При этом число объектов в каждом подмножестве должно находиться в заданном интервале $[\alpha,\beta]$. Перераспределение объектов между *КМ* выполняется так, чтобы максимально улучшить компактность разбиения.

Пример задачи перераспределения объектов представлен на рис. 2. Граф $T(G)$ содержит 33 объекта, которые необходимо разбить на $K=5$ подмножеств с мощностями из интервала [6, 8]. Список $KM(8)$ содержит 5 множеств с пометкой $d=1$. Принимая $L_x=80$, $L_y=45$, $\varphi=1{,}2$, величина Δr составит 32. Список $KM(8)$ образует одно скопление с плотностью $P(8)=(5\cdot8–33)/5\cdot8=7/40$. Низкая плотность объясняется небольшим объемом пересечений *КМ* и нулевой оценкой $\delta(8)$, т. к. у всех *КМ* пометка $d=1$. Если принять $d_s=7$, то плотность $P(8)$ возрастет до 23/40.

Из рис. 2 следует, что перераспределению подлежат объекты $\{e_9, e_{10}, e_{11}, e_{22}, e_{23}, e_{25}\}$. Решение данной задачи можно получить путем последовательного рассмотрения каждого из этих объектов и отнесения его к ближайшему из альтернативных *КМ*. Например, если удаление e_{23} от объектов в KM_5 меньше, чем от объектов в KM_2 или KM_3, то e_{23} следует отнести к KM_5. При этом необходимо следить, чтобы в KM_1, KM_2, KM_3 число объектов не оказалось меньше $\alpha=6$.

Данную задачу можно записать в оптимизационной постановке. Для этого введем переменную $x_{ij}=1$, если объект e_i, подлежащий перераспределению, отнесен к KM_j, и $x_{ij}=0$, если этого не случилось. Оценку удаления объекта e_i от объектов KM_j обозначим величиной c_{ij}. Тогда задача перераспределения объектов между *КМ* запишется в виде:

$$\sum_{i=1}^{n_q}\sum_{j=1}^{m_q} c_{ij}x_{ij} \Rightarrow \min; \qquad (5)$$

$$\sum_{j=1}^{m_q} x_{ij} = 1,\ i = 1,2,...n_q; \qquad (6)$$

$$\sum_{i=1}^{n_q} x_{ij} \ge s_j,\ j = 1,2,..,m_q. \qquad (7)$$

Здесь n_g – число объектов, подлежащих перераспределению; m_g – число *КМ*, участвующих в перераспределении; s_j – минимальное число объектов, которые необходимо отнести к KM_j, чтобы в нем оказалось не менее α объектов.

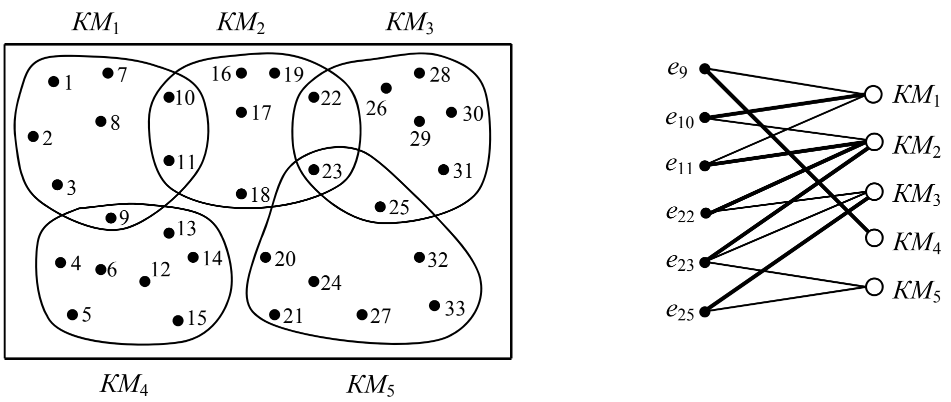


__Рис. 2__. Пример перераспределения объектов
__Fig. 2.__ Example of objects redistribution

Уравнение (6) обеспечивает отнесение каждого объекта e_i к одному из KM_j. Неравенство (7) требует сохранить в KM_j число объектов из интервала $[\alpha, \beta]$.

Заметим, что для решения задачи при условии $g>\beta$ требуется дополнительное ограничение:

$$\sum_{i=1}^{n_q} x_{ij} \le p_j,\ j = 1, 2, ..., m_q. \qquad (8)$$

Здесь p_j – максимальное число объектов, которые допускается отнести к KM_j, чтобы в нем оказалось не более β объектов.

Ограничение (8) предполагает, что при условии $g>\beta$ мощность любого объединения $|KM_{j1} \cup KM_{j2}| \le 2\beta$. Это обеспечивает совместность задачи (5)–(8). Соблюдение данного условия при работе алгоритма наращивания мощности *КМ* неочевидно. Поэтому вопрос о совместности задачи (5)–(8) требует дополнительного исследования и здесь не рассматривается. Можно предположить также, что в работу алгоритма при $g>\beta$ потребуется внести некоторые дополнения.

На рис. 2 справа приведена иллюстрация к задаче перераспределения объектов e_i по KM_j. Ребра (e_i, KM_j) соответствуют возможным вариантам отнесения e_i к KM_j и имеют веса c_{ij}. Величины s_j для данного примера следующие: $s_1=s_3=1$, $s_2=2$, $s_4=s_5=5$. Допустимый вариант отнесения e_i к KM_j выделен жирными ребрами. В итоге 33 объекта разбиваются на 5 подмножеств с мощностями равными 6, 7, 6, 8, 6, соответственно.

Очевидно, что вариант решения для данного примера даже в постановке (5)–(7) не является наилучшим. Это связано с тем, что величины c_{ij} вычисляются относительно исходных KM_j и не учитывают перераспределения объектов в результате решения задачи. Попытки последовательного уточнения центров в новых подмножествах и решения задачи распределения объектов относительно этих центров могут улучшить компактность подмножеств, но это возвращает нас к проблеме большой размерности такой задачи.

Что касается размерности данной задачи, то она определяется числом подмножеств и числом объектов, подлежащих перераспределению. Число подмножеств в практических приложениях даже для множеств в несколько тысяч объектов, как правило, ограничивается несколькими десятками. Число объектов для перераспределения зависит от превышения значения g над средним значением интервала $[\alpha, \beta]$. Число таких объектов ограничивается в первую очередь тем, что дублирующие *КМ* в ходе работы алгоритма исключаются. Важно также, что задача решается для каждого скопления отдельно. Некоторые скопления с плотностью $P(g)=1$ при определенных условиях включаются в состав подмножеств компактного разбиения.

Все приводит к снижению общей размерности задачи. Заметим также, что решение задачи (5)–(8) не является обязательным для алгоритма. Имеется возможность последовательно перераспределять объекты, анализируя предпочтение каждого из них на отнесение к одному из альтернативных *КМ*, соблюдая при этом ограничения задачи.

Заключение

В статье предложен алгоритм приближенного решения задачи разбиения множества объектов топологического графа большой размерности на заданное число подмножеств. В алгоритме используется эффект образования и обособления скоплений объектов повышенной плотности при формировании компактных множеств. Алгоритм, начиная с каждого объекта, последовательно наращивает мощность компактных множеств, анализирует образовавшиеся скопления и при определенных условиях переходит к формированию компактного разбиения.

Для снижения объемов вычислений предлагается ввести понятие и вычислить размер активной зоны поиска ближайших объектов. Это позволило преобразовать топологический граф и представить его списком инциденторов, в каждом из которых содержатся объекты из соответствующей зоны поиска.

Эффект обособления скоплений, отражающих топологию расположения объектов на территории, оказался полезным не только при решении задачи компактного разбиения, но и в других приложениях по анализу территориально распределенных систем. Способность алгоритма выделять и анализировать скопления может быть использована для принятия решений по компактному разбиению в условиях, когда число подмножеств заведомо неизвестно. В этом случае состав и расположение скоплений дают основание для рекомендаций по структуризации на множестве объектов системы.

В представленном виде статья содержит изложение основных положений алгоритма. В последующем предстоит программирование, проведение экспериментальных исследований, совершенствование алгоритма и настройка на условия решения задач структуризации больших множеств объектов.

СПИСОК ЛИТЕРАТУРЫ

1. Андреев А.М., Можаров Г.П., Сюзев В.В. Многопроцессорные вычислительные системы: теоретический анализ, математические модели и применение. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2011. – 332 с.
2. Олифор В., Олифер Н. Компьютерные сети. Принципы, технологии, протоколы. – СПб: Питер, 2013. – 944 с.
3. Навигационно-телекоммуникационные системы мониторинга и управления труднодоступными объектами и мобильными группами / М.А. Сонькин, В.З. Ямпольский, В.К. Погребной, Е.И. Печерская, А.А. Шамин, С.В. Семыкин, Д.М. Сонькин. – Томск: Изд-во НТЛ, 2013. – 220 с.
4. Hu Fei, Xiaojun Cao. Wireless sensor networks: principles and practice. – NY: Auerbach Publications, 2010. – 531 p. URL: https://archive.org/details/wirelesssensorne0000hufe/page/n7/mode/2up (дата обращения: 21.06.2023).
5. Погребной Ал.В., Погребной Ан.В. Алгоритм решения задачи компактного разбиения множества объектов территориально распределенной системы // Известия Томского политехнического университета. – 2010. – Т. 317. – № 5. – С. 22–28.
6. Погребной Ал.В., Погребной В.К. Задача разбиения множества объектов территориально распределенной системы на подмножества неравной мощности // Известия Томского политехнического университета. – 2012. – Т. 321. – № 5. – С. 61–65.

Поступила: 20.08.2023
Принята: 03.09.2023

Информация об авторах

Погребной А.В., кандидат технических наук, доцент отделения информационных технологий Инженерной школы информационных технологий и робототехники Национального исследовательского Томского политехнического университета.

Погребной А.В., программист общества с ограниченной ответственностью «Дром».

UDC 519.175.1
DOI: 10.18799/29495407/2023/2/26

COMPACT PARTITIONS ON LARGE DIMENSION TOPOLOGICAL GRAPHS

Alexander V. Pogrebnoy[1],
pogrebnoy@tpu.ru

Andrey V. Pogrebnoy[2],
avpogrebnoy@gmail.com

[1] National Research Tomsk Polytechnic University,
30, Lenin avenue, Tomsk, 634050, Russia.

[2] Limited Liability Company "Drom",
16, Lebedev street, Tomsk, 634061, Russia.

Relevance. *Distributed systems containing hundreds and thousands of objects are usually built in the form of hierarchical structures. In these structures, lower-level objects are combined into subsets for connection to the corresponding centers. The existing algorithms are not capable of successfully solving structuring problems on sets of this dimension. Therefore, new algorithms are needed that are suitable for solving structuring problems on sets containing thousands of objects.* ***Aim****. To develop an algorithm for generating a compact partition on high-dimensional sets containing up to a thousand objects located in a given territory.* ***Methods****. Applied graph theory, linear programming methods, construction and analysis of the effectiveness of algorithms, the theory of compact partitions, compact sets of objects and their clusters.* ***Results.*** *It is proposed to represent the territorial location of many objects of a distributed system in the form of a topological graph. The paper introduces the concept of an active search zone for the nearest vertices to increase the efficiency of the algorithm for generating compact sets and identifying clusters. This makes it possible to replace the matrix of distances between graph vertices with a list of vertex incidentors generated based on the active search zone. The authors developed the algorithm for approximate solution to the problem of compact partitioning a set of objects of a topological graph, represented by a list of vertex incidentors, into a given number of subsets. For each object, the algorithm recursively increases the power of compact sets, analyzes the resulting clusters and, under certain conditions, proceeds to the formation of a compact partition. The problem of forming subsets of a compact partition based on clusters is formed as a linear programming problem of transport type. The algorithm presentation is accompanied by an example.*

Keywords: *Compact partition, compact set, cluster of objects, cluster density, topological graph.*

REFERENCES

1. Andreev A.M., Mozharov G.P., Syuzev V.V. *Mnogoprotsessornye vychislitelnye sistemy: teoreticheskiy analiz, matematicheskie modeli i primenenie* [Multiprocessor computing systems: theoretical analysis, mathematical models and applications]. Moscow, MSTU im. N.E. Bauman Publ. house, 2011. 332 p.
2. Olifor V., Olifer N. *Kompyuternye seti. Printsipy, tekhnologii, protokoly* [Computer networks. Principles, technologies, protocols]. St Petersburg, Piter Publ., 2013. 944 p.
3. Sonkin M.A., Yampolsky V.Z., Pogrebnoy V.K., Pecherskaya E.I., Shamin A.A., Semykin S.V., Sonkin D.M. *Navigatsionno-telekommunikatsionnye sistemy monitoringa i upravleniya trudnodostupnymi obyektami i mobilnymi gruppami* [Navigation and telecommunication systems for monitoring and controlling hard-to-reach objects and mobile groups]. Tomsk, STL Publ. House, 2013. 220 p.
4. Hu Fei, Xiaojun Cao. *Wireless sensor networks: principles and practice*. NY, Auerbach Publications, 2010. 531 p. Available at: https://archive.org/details/wirelesssensorne0000hufe/page/n7/mode/2up (accessed: 21 June 2023).
5. Pogrebnoy Al.V., Pogrebnoy An.V. Algorithm for solving the problem of compact partitioning of a set of objects of a geographically distributed system. *Bulletin of the Tomsk Polytechnic University*, 2010, vol. 317, no. 5, pp. 22–28. In Rus.
6. Pogrebnoy A.V., Pogrebnoy V.K. The problem of partitioning a set of objects of a territorially distributed system into subsets of unequal power. *Bulletin of the Tomsk Polytechnic University*, 2012, vol. 321, no. 5, pp. 61–65. In Rus.

Received: 20.08.2023
Reviewed: 03.09.2023

Information about the authors

Alexander V. Pogrebnoy, Cand. Sc., associate professor, National Research Tomsk Polytechnic University.
Andrey V. Pogrebnoy, programmer, Limited Liability Company "Drom".