Информационные процессы, Том 18, № 4, 2018, стр. 424–438

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ТЕХНИЧЕСКИХ СИСТЕМАХ

Балансная кластеризация с деревом над кластерами [1]

Марк Ш. Левин

Институт проблем передачи информации, Российская академия наук, Москва, Россия
email: mslevin@acm.org

Поступила в редколлегию 12.12.2018

Аннотация—В статье рассматривается балансная кластеризация с покрывающим деревом над кластерами. Такая задача может быть полезна в сетевых приложениях (проектирование, управление, маршрутизация). Рассматриваются элементы разного типа и соответствующие задачи балансной кластеризации с учетом состава элементов кластеров. Сформулированная оптимизационная задача основана на векторной целевой функции, включающей качество балансной кластеризации и качество покрывающего дерева. Предложены четыре базовых эвристических стратегии решения: балансирование-покрытие, покрытие-балансирование, прямое решение, построение много-уровневой структуры с сбалансированными кластерами. Приведено подробное описание стратегии покрытие-балансирование, включая четыре схемы решения и численный пример.

КЛЮЧЕВЫЕ СЛОВА: балансная кластеризация, комбинаторная оптимизация, покрывающее дерево, эвристики, сети

1. ВВЕДЕНИЕ

Балансная кластеризация часто является важной частью многих теоретических и практических задач. Общее описание задач балансной кластеризации приведено в [48, 49]. В некоторых случаях балансный подход позволяет обеспечит эффективную кластеризацию множеств данных большой размерности [70]. В данной статье рассматривается специальная задача балансной кластеризации с дополнительным требованием в виде покрывающего дерева над полученными кластерами. Такой тип кластеризации может быть полезен в сетевых приложениях (проектирование топологий, управление, маршрутизация). Общее качество решения данной задачи рассматривается как качество балансной кластеризации и качество покрывающего дерева (т.е., вектор из двух компонент). В результате, задача направлена на поиск Парето-эффективных решений. Задачи балансной кластеризации основаны на следующих требованиях баланса [48, 49]: (i) баланс по размеру кластеров (т.е., по числу элементов в кластерах), (ii) баланс по общему весу элементов в кластерах, (iii) баланс по составу типов элементов в кластерах (здесь рассматриваются несколько типов элементов). В статье, в основном, исследуется третий вариант балансного требования. При этом, качество решений основано на использовании мультимножеств [45, 48, 49]. В качестве базовых типов покрывающих деревьев обычно используются деревья, сбалансированные по степеням вершин (т.е., по числу "сыновей" каждой вершины) и деревья, сбалансированные по высоте. Качество покрывающего дерева можно оценивать как близость (похожесть) полученного покрывающего дерева к дереву заданного типа (т.е., заданному типу балансной структуры).

В данной работе предложены четыре базовых быстрых эвристических стратегии решения:

(1) балансирование-покрытие: балансная кластеризация элементов и последующее построение покрывающего дерева для полученных кластеров;

[1] Исследование выполнено в ИППИ РАН за счет гранта Российского научного фонда (проект 14-50-00150).

(2) покрытие-балансирование: покрытие исходных элементов минимальным деревом и разбиение полученного дерева на сбалансированные кластеры;

(3) прямое решение задачи: формирование сбалансированных кластеров с одновременным построением покрывающего дерева;

(4) построение много-уровневой структуры, на каждом уровне имеется набор сбалансированных кластеров и над этими кластерами сформировано покрывающее дерево.

Для второй стратегии представлены четыре схемы решения и иллюстративные числовой пример. Статья базируется на предварительном материале [50].

2. ОПИСАНИЕ ЗАДАЧИ

Иллюстрация рассматриваемой задачи представлена на Рис. 1.

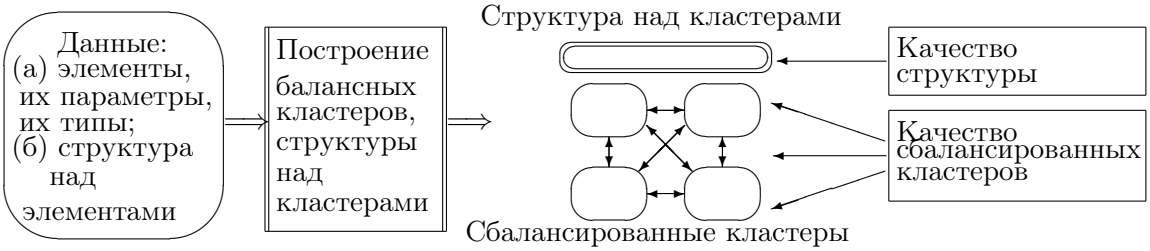

Рис. 1. Баласная кластеризация с структурой над кластерами

Следующие обозначения используются [45–49]:

(1) множество исходных элементов $A = \{a_1, ..., a_i, ..., a_n\}$;

(2) решение кластеризации $\widetilde{X} = \{X_1, ..., X_j, ...X_k\}$, где $X_j \subseteq A$ (для упрощения - без пересечения: $|X_{j_1} \cap X_{j_1}| = 0 \ \ \forall\ 1 \le j_1 < j_2 \le k$);

(3) типы элементов: тип 1,..., тип ξ, ...,тип l;

(4) описание кластера X_j на основе входящих в него элементов имеет вид (вектор, оценка в виде мультимножества) [45, 48, 49]): $e(X_j) = (\eta_1(X_j), ..., \eta_\xi(X_j), ..., \eta_l(X_j))$, где $\eta_\xi(X_j)$ - число элементов типа ξ которые содержатся в кластере X_j, $\sum_{\xi=\overline{1,l}} \eta_\xi(X_j) = |X_j|$.

Четыре типа элементов и несколько примеров кластеров приведены на Рис. 2. Описание (оценки) примеров кластеров имеют следующий вид (в виде мультимножеств):

$e(X_1) = (0, 0, 0, 4)$, $e(X_2) = (0, 0, 0, 3)$, $e(X_3) = (0, 0, 1, 3)$, $e(X_4) = (0, 0, 1, 2)$,

$e(X_5) = (0, 1, 2, 1)$, $e(X_6) = (0, 1, 1, 1)$, $e(X_7) = (1, 1, 1, 1)$, $e(X_8) = (0, 1, 2, 0)$.

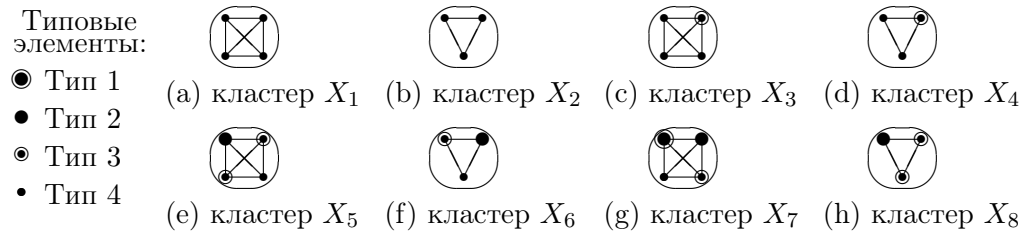

Рис. 2. Иллюстративный пример: типовые элементы и кластеры

Далее на рисунках приведены иллюстративные примеры решения рассматриваемой задачи:

(a) сбалансированные кластеры (по размеру, т.е., по числу элементов, число элементов примерно равно 4) и покрывающее дерево над кластерами (Рис. 3);

(b) сбалансированные кластеры (баланс по типу элементов: три элемента типа 2, один элемент типа 2) и и покрывающее дерево над кластерами (Рис. 4);

(c) кластеры разбиты на уровни, кластеры сбалансированы на каждом уровне (по типа элементов, по размеру кластера), покрывающая кластеры древовидная структура (Рис. 5); структуры кластеров на уровнях: нижний уровень: 3 элемента типа 4, 1 элемент типа 3, средний уровень: 3 элемента типа 3, 1 элемент типа 2, верхний уровень: 3 элемента типа 2, 1 элемент типа 1.

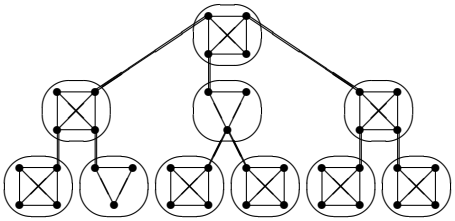

Рис. 3. Баланс по размеру кластера

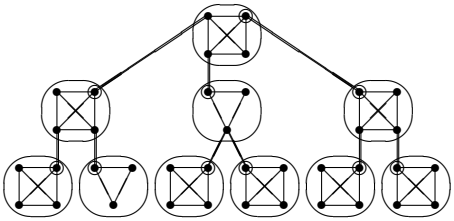

Рис. 4. Баланс по типам элементов

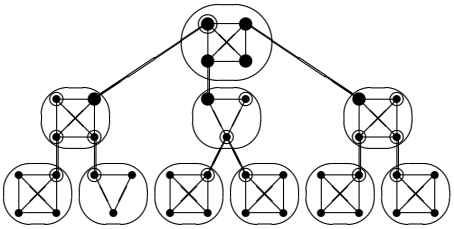

Рис. 5. Трех-уровневая структура

Качество получаемого решения включает две части:

(i) качество решения балансной кластеризации в виде близости (похожести) каждого сбалансированного кластера к заданному требованию (например: по числу элементов, по общему весу кластера, по типам элементов, входящих в кластер) или интегральной близости [48, 49];

(ii) качество выполнения требования к покрывающей структуре в виде близости (похожести) полученной покрывающей структуры к заданной типовой структуре (например: близость к дереву, балансному дереву, цепочке, иерархии) [45].

Рассмотрим оценку балансного решения кластеризациии. Пусть оценка $e^0 = (\eta_1^0, ..., \eta_\xi^0, ..., \eta_l^0)$ будет соответствовать требуемой структуре кластера (по типам элементов). Тогда введем близость для кластера X_j как различие по компонентам: $\delta(e(X_j), e^0)$. Для упрощения, следующая оценка для кластера используется: $\delta(e(X_j), e^0) = \sum_{\xi=\overline{1,l}} |\eta_\xi(X_j) - \eta_\xi^0|$. Для интегральной оценки решения балансной кластеризации $\widetilde{X}$ можно рассматривать следующий подход: $Q^{cb} = \max_{j=\overline{1,k}} \ \delta(e(X_j), e^0)$.

В качестве основы требования к покрывающей структуре модно использовать следующие типовые структуры [2,4,15,20,29,42,45,66]: (i) минимальные покрывающие деревья T^m) [18,20, 28, 29, 42], (ii) сбалансированные по ступени вершин деревья (T^d) (Fig. 6) [12, 15, 63] (включая k-арные деревья [66], деревья с ограниченной степенью [43]), (iii) сбалансированные по высоте деревья (T^h) [1,2,19,20,29,40,42] (Fig. 7), (iv) сбалансированные по весу деревья (T^w) [10,35,58], (v) специальные много-уровневые структуры [45].

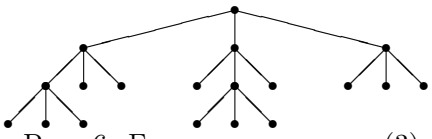

Рис. 6. Баланс по степени (3)

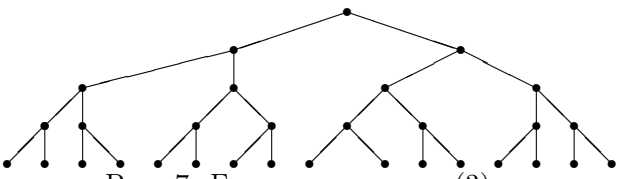

Рис. 7. Баланс по высоте (3)

Пусть T^r будет требуемая покрывающая структура (заданный тип дерева), $T(\widetilde{X})$ будет полученная покрывающая структура над решении балансной кластеризацие $\widetilde{X}$. Следует заметить, что подходы к оценке близости структур кратко рассмотрены в [45]. Оценку качества покрывающей структуры обозначим так: $Q^s = \Delta(T(\widetilde{X}), T^r)$. В результате, общая оценка качества решения кластеризации $\widetilde{X}$ с соответствующей покрывающей структурой $T(\widetilde{X})$ представляет собой вектор: $(Q^{cb}, Q^s) = (\max_{j=\overline{1,k}} \delta(e(X_j), e^0), \Delta(T(\widetilde{X}), T^r)$. Задача балансной кластеризации с требуемой покрывающей структурой имеет вид:

Найти решений балансной кластеризации $\widetilde{X}$ и соответствующую покрывающую структуру $T(\widetilde{X})$ такую, что $(Q^{cb}, Q^s) \Longrightarrow \min$.

Очевидно, Парето-эффективные решений рассматриваются для данной задачи.

В случае много-уровневой структуры, следующая общая схема может использоваться: (1) разбиение исходного множества элементов на подмножества, соответствующие каждому уровню, (2) решение задачи на каждом уровне, (3) установление связей между уровнями. В некоторых случаях, фазы разбиения элементов и покрытия структурой могут интегрироваться, например в задаче построения покрывающего дерева с максимальным числом висячих вершин.

Указанные задачи являются очень сложными и даже их подзадачи относятся к классу NP-трудных проблем. Только некоторые упрощенные версии позволяют применять являются полиномиальные алгоритмы (например, базовая задача построения минимального покрывающего дерева). Таким образом, целесообразно применение составных (композитных) схем решений на основе комбинаций подзадач, в частности, балансная кластеризация и задача покрытия. Очевидно, для каждой части таких составных схем решения могут использоваться различные методы: (i) эвристики (в частности, жадные алгоритмы, приближенные алгоритмы), (ii) переборные методы, (iii) точные полиномиальные алгоритмы для простейших случаев.

3. БАЗОВЫЕ СТРАТЕГИИ РЕШЕНИЯ

Четыре базовые стратегии решений указаны в Таблице 1. Далее, в основном, рассматривается баланс кластеров по типам содержащихся элементов.

Таблица 1. Базовые стратегии решения

Ном.	Стратегия	Фазы
1.	Стратегия балансировка-покрытие:	(1) балансная кластеризация, (2) покрывающая структура (например, покрывающее дерево) над кластерами
2.	Стратеги покрытие-балансировка:	(1) покрывающая структура (например, дерево) над элементами (2) балансное разбиение покрывающей структуры (например, разбиение дерева)
3.	Стратеги прямого решения:	кластеризация элементов с учетом требований баланса к кластерам и покрывающией структуре
4.	Построение сбалансированной много-уровневой структуры:	(1) разбиение множества элементов на подмножества соответствующие уровням (2) балансная кластеризация элементов на каждом уровне с учетом требования к покрывающей структуре над кластерами (3) построение связей между уровнями

3.1. Стратегия балансировка-покрытие

Данная стратегия (стратегия 1) включает две фазы:

Фаза 1. Балансная кластеризация исходного множества элементов [48, 49].

Фаза 2. Построение покрывающей структуры над множеством полученных кластеров (например, построение покрывающего дерева над полученными кластерами, базовые задачи покрытия деревом указаны в Таблице 2).

Эта стратегия является, по сути, базовой и применяется в многих сетевых приложениях (проектирование, управление, маршрутизация).

Таблица 2. Задачи/методы покрытия

Ном.	Исследование	Источник
1.	Задачи минимальных покрывающих деревьев:	
1.1.	Базовые задачи	[18, 20, 28, 29, 60]
1.2.	Задача покрывающее дерево минимального диаметра	[31]
1.3.	Задачи минимального покрывающего леса	[29, 61]
1.4.	Задачи покрытия минимальных мульти-деревьев	[30, 36, 68]
1.5.	Задача минимального покрывающего дерева с ограничением по висячим вершинам	[64]
2.	Задачи многокритериального покрывающего дерева:	
2.1.	Задача покрывающего дерева с многими целевыми функциями	[34]
2.2.	Многокритериальная задача минимального покрывающего дерева	[7, 16]
2.3.	Комбинирование линейных и нелинейных целевых функций в задачах покрывающего дерева	[23]
3.	Задачи покрывающих деревьев с максимизацией числа висячих узлов:	
3.1.	Задачи покрывающих деревьев с максимизацией числа висячих узлов	[26, 27, 29, 53, 54, 65]
3.2.	Задачи покрывающих деревьев с многими висячими узлами	[41, 65]
3.3.	Задачи покрывающих ориентированных деревьев с максимизацией числа висячих узлов	[3]
3.5.	Задачи связанного доминирующего множества (в смысле точного алгоритма) задача эквивалентна задаче покрывающего дерева с максимизацией числа висячих узлов	[11, 13, 17, 29, 69]
4.	Задачи покрытия сбалансированным деревом:	
4.1.	Базовые задачи покрытия сбалансированным деревом	[2, 15, 20, 29, 42]
4.2.	Задачи покрытия деревом сбалансированным по высоте	[1, 2, 20, 29, 42]
4.3.	Задачи покрытия деревом сбалансированным по степени	[12, 15, 63]

3.2. Стратегия покрытие-балансировка

Данная стратегия (стратегия 2) включает две фазы:

Фаза 1. Построение покрывающей структуры для исходного множества элементов (т.е., решение задачи типа минимального покрытия деревом для исходного множества элементов) (Таблица 2).

Фаза 2. Формирование сбалансированных кластеров (например, объедиение/интеграци соседних элементв) с учетом построенной общей структуры. Здесь можно рассматривать задачу типа разбиения структуры (например, разбиение дерева) и соответствующие методы решения (Таблица 3).

Далее, четыре схемы решения описаны (на основе подхода как в “жадных” алгоритмах). Эти стратегии базируются на выделении интегрируемых точек (узлы или ребра исследуемого дерева). Соседний узел, соответствующие интегрируемой точке, объединяется с указанной точкой для получения результирующего сбалансированного кластера или его части (с учетом требований к кластеру).

Следующие четыре типа интегрируемых точек рассматриваются:

(1) ребро минимального веса (как в иерархической кластеризации) (Рис. 8);

(2) висячие узлы: ребро минимального веса между висячим узлом к его соседу (процедура снизу-вверх) (Рис. 9);

(3) корневой узел, т.е., ребро минимального веса между корнем и его соседним узлом (процедура сверху-вниз) (Рис. 10);

(4) интегрирование указанных точек посредством использования специальной процедура размещения (Рис. 11).

Таблица 3. Некоторые подходы к задаче разбиения деревьев/иерархий

Ном.	Исследование	Источник
1.	Полиномиальные алгоритмы для разбиения дерева	[6]
2.	Алгоритмы на основе сдвига для разбиения деревьев	[9]
3.	Задача разбиения дерева с критерием max-min	[59]
4.	Балансное разбиение деревьев	[25]
5.	Задача разбиения дерева	[56]
6.	Разбиение иерархически кластеризованной сложной сети	[57]
7.	Оптимальная иерархическая декомпозиция графа (с минимизацией загрузок в сети)	[62]
8.	Кластеризация на основе минимального покрывающего дерева	[33, 46]
9.	Кластеризация на основе минимального и максимального покрывающего дерева	[8]

Рассматриваются следующие схемы решения:

Схема 2.1. Выделение ребра (или ребер) минимального веса (Рис. 8):

Стадия 1. Анализ исходного дерева. Выделение ребра минимального веса (a', a''), желательно, a' и a'' являются узлами разных типов.

Стадия 2. Построение интегрированного узла: $J_{a',a''} = a'\&a''$

Стадия 3. Присоединение соседнего узла к интегрированному узлу J для получения требуемого (по элементной структуре) кластера (или квази-кластера).

Стадия 4. Коррекция исходного дерева посредством исключения полученного кластера.

Стадия 5. Анализ нового дерева (или деревьев). Переход к Стадии 1 для исследования полученного дерево, если оно существует; иначе переход к Стадии 6.

Стадия 6. Анализ результирующего решения кластеризации. При наличии отдельных узлов, присоединение этих узлов к ближайшему кластеру (или общая задача присоединения всех отдельных узлов к кластерам). Оценивание решения.

Стадия 7. Стоп.

Схема 2.2. Выделение ребра (или ребер) минимального веса для висячего узла (или висячих узлов) (Рис. 9):

Стадия 1. Анализ исходного дерева. Выделение висячего узла b с минимальным весом ребра к его соседнему узлу a; обычно предпочтительнее, когда b и a имеют разные типы.

Стадия 2. Построение интегрированного узла $J_{b,a} = b\&a$

Стадия 3. Присоединение соседнего узла к интегрированному узлу J для получения требуемого (по элементной структуре) кластера (или квази-кластера).

Стадия 4. Коррекция исходного дерева посредством исключения полученного кластера.

Стадия 5. Анализ нового дерева (или деревьев). Переход к Стадии 1 для исследования полученного дерево, если оно существует; иначе переход к Стадии 6.

Стадия 6. Анализ результирующего решения кластеризации. При наличии отдельных узлов, присоединение этих узлов к ближайшему кластеру (или общая задача присоединения всех отдельных узлов к кластерам). Оценивание решения.

Стадия 7. Стоп.

Схема 2.3. Выделение ребра (или ребер) минимального веса для корневого узла (или узлов) (Рис. 10):

Стадия 1. Анализ исходного дерева. Выделение соседнего узла a для корня r с минимальным весом ребра между ними; обычно предпочтительнее, когда r и a имеют разные типы.

Стадия 2. Построение интегрированного узла: $J_{r,a} = r\&a$.

Стадия 3. Присоединение соседнего узла к интегрированному узлу J для получения требуемого (по элементной структуре) кластера (или квази-кластера).

Стадия 4. Коррекция исходного дерева посредством исключения полученного кластера.

Стадия 5. Анализ нового дерева (или деревьев). Переход к Стадии 1 для исследования полученного дерево, если оно существует; иначе переход к Стадии 6.

Стадия 6. Анализ результирующего решения кластеризации. При наличии отдельных узлов, присоединение этих узлов к ближайшему кластеру (или общая задача присоединения всех отдельных узлов к кластерам). Оценивание решения.

Стадия 7. Стоп.

Схема 2.4. Процедура на основе центров кластеров (это похоже на кластеризацию на основе -средних) (Рис. 11):

Стадия 1. Анализ исходного дерева. Выделение (отбор) специальных узлов (например, узлов типа 1) в качестве центров кластеров $c^1, ..., c^\gamma, ..., c^\lambda$.

Стадия 2. Выделение центра кластера c^γ который имеет соседний узел a (другой тип узла является обычно предпочтительным) с минимальным весом ребра.

Стадия 3. Построение интегрированного узла: $J_{c^\gamma,a} = c^\gamma\&a$

Стадия 4. Присоединение соседнего узла у интегрированному узлу J для получения требуемого (по элементной структуре) кластера (или квази-кластера).

Стадия 5. Коррекция исходного дерева посредством исключения полученного кластера.

Стадия 6. Анализ нового дерева (или деревьев). Переход к Стадии 1 для исследования полученного дерева, если оно существует; иначе переход к Стадии 7.

Стадия 7. Анализ результирующего решения кластеризации. При наличии отдельных узлов, присоединение этих узлов к ближайшему кластеру (или общая задача присоединения всех отдельных узлов к кластерам). Оценивание решения.

Стадия 8. Стоп.

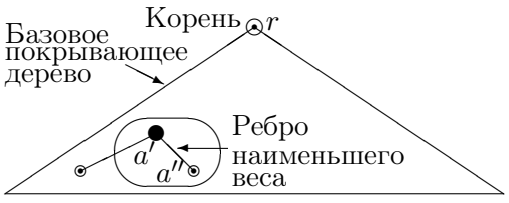

Рис. 8. Ребро наименьшего веса

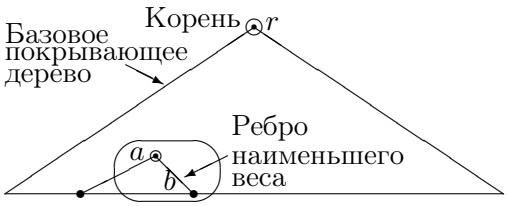

Рис. 9. Ребро из висячего узла

Представляется важным указать следующие замечания:

Замечание 1. Приведенные схемы решения являются полиномиальными (их сложность равна $O(n^2)$ или меньше).

Замечание 2. Очевидно, описанные схемы могут трансформироваться в параллельные процедуры (т.е., параллельное исследование точек интеграции и т.д.).

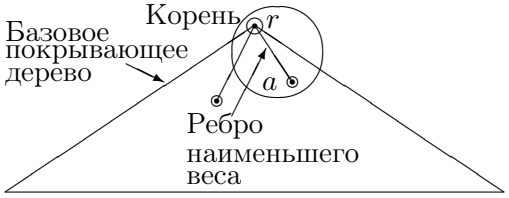

Рис. 10. Ребро из корня

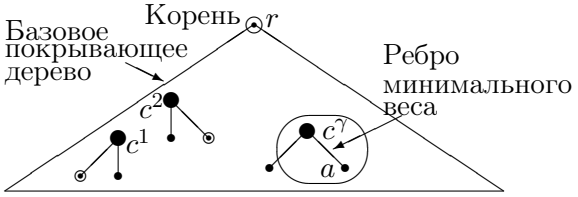

Рис. 11. Ребро из одного из заданных центров

Далее представлен иллюстративный пример:

(a) исходное множество элементов и их параметры содержатся в Таблице 4:

$A = \{a_1, ..., a_i, ..., a_{21}\}$;

(b) веса ребер между парой узлов a_{i_1} и a_{i_2} $(1 \le i_1 <\le i_2 \le 18)$ содержатся в Таблице 5;

(c) базовая покрывающая структура (минимальной по весам ребер дерево, получается на первой фазе) изображено на Рис. 12.

Предполагается, что требуемая структура кластера (по типам элементов) имеет следующий вид: узел типа 1, узел типа 2, узел типа 3. Соответствующая оценка такого кластера (мультимножество) равна $e^0 = (1, 1, 1)$. Получается следующее базовое множество висячих узлов: $L = \{a_8, a_9, a_{10}, a_{11}, a_{12}, a_{15}, a_{16}, a_{17}, a_{18}, a_{19}\}$ (Таблица 6).

Таблица 4. Элементы и их параметры

Ном.	Элемент a_i	Тип элемента ξ	Номер кластера X_j (в решении $\widetilde{X}$)
1.	a_1	1	6
2.	a_2	2	6
3.	a_3	1	6
4.	a_4	1	2
5.	a_5	1	5
6.	a_6	1	3
7.	a_7	2	2
8.	a_8	3	2
9.	a_9	2	5
10.	a_{10}	3	5
11.	a_{11}	2	3
12.	a_{12}	1	3
13.	a_{13}	1	4
14.	a_{14}	1	1
15.	a_{15}	2	4
16.	a_{16}	3	4
17.	a_{17}	2	1
18.	a_{18}	3	1
19.	a_{19}	3	1

Таблица 5. Веса ребер между элементами a_{i_1} и a_{i_2}

i_1	i_2 :	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1		2.5	2.8	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
2			⋆	3.0	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
3				⋆	3.5	4.1	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
4					⋆	⋆	1.0	0.6	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
5						⋆	⋆	⋆	1.3	1.2	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
6							⋆	⋆	⋆	⋆	1.1	1.0	⋆	⋆	⋆	⋆	⋆	⋆	⋆
7								⋆	⋆	⋆	⋆	⋆	4.2	⋆	⋆	⋆	⋆	⋆	⋆
8									⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
9										⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
10											⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
11												⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
12													⋆	⋆	⋆	⋆	⋆	⋆	⋆
13														⋆	1.1	1.3	⋆	⋆	⋆
14															⋆	⋆	1.0	0.5	2.0
15																⋆	⋆	⋆	⋆
16																	⋆	⋆	⋆
17																		⋆	⋆
18																			⋆

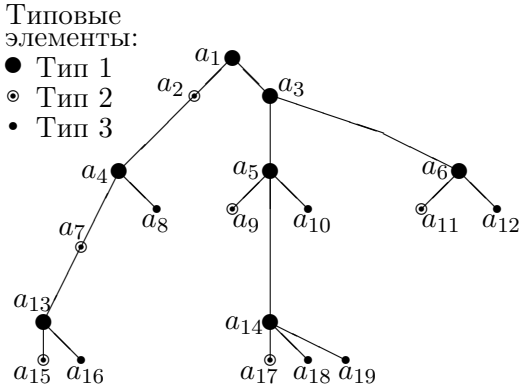

Рис. 12. Базовое покрывающее дерево

Таблица 6. Висячие узлы и соответствующие веса ребер

Висячий узел a_i	Ребро (a_i, b)	Вес ребра $w(a_i, b)$
I. Базовые висячие узлы:		
a_8	(a_8, a_4)	0.6
a_9	(a_9, a_5)	1.3
a_{10}	(a_{10}, a_5)	1.2
a_{11}	(a_{11}, a_6)	1.3
a_{12}	(a_{12}, a_6)	1.0
a_{15}	(a_{15}, a_{13})	1.1
a_{16}	(a_{16}, a_{13})	1.3
a_{17}	(a_{17}, a_{14})	1.0
a_{18}	(a_{18}, a_{14})	0.5
a_{19}	(a_{19}, a_{14})	2.0
II. Дополнительные висячие узлы:		
a_2	(a_2, a_1)	2.5
a_3	(a_3, a_1)	2.8

Здесь используется Схема 2.2:

1-й шаг: Выделение ребра минимального веса: (a_{18}, a_{14}) (вес $w(a_i, b) = 0.5$). Интеграция узлов: $J_{a_{18},a_{14}} = a_{18}\&a_{14}$ (исходная часть кластера). Очевидное присоединение узла a_{17} к интегрированному узлу для получения сбалансированного (по элементной структуры) кластера $X_1 = \{a_{14}, a_{17}, a_{18}\}$, оценка структуры кластера равна $e(X_1) = (1, 1, 1)$.

2-й шаг: Выбор ребра минимального веса: (a_8, a_4) (вес $w(a_i, b) = 0.6$). Интеграция узлов: $J_{a_8,a_4} = a_8\&a_4$ (исходная часть кластера). Очевидное присоединение узла a_7 к интегрированному узлу для получения сбалансированного (по элементной структуре) кластера $X_2 = \{a_4, a_7, a_8\}$, оценка структуры кластера равна $e(X_2) = (1, 1, 1)$.

Получается дополнительный висячий узел a_2 (дополнение к Таблице 24).

3-й шаг: Выделение ребра минимального веса: (a_{12}, a_6) (вес $w(a_i, b) = 1.0$). Интеграция узлов: $J_{a_{12},a_6} = a_{12}\&a_6$ (исходная часть кластера). Очевидное присоединение узла a_{11} к интегрированному узлу для получения сбалансированного (по элементной структуре) кластера $X_3 = \{a_6, a_{11}, a_{12}\}$, оценка структуры кластера равна $e(X_3) = (1, 1, 1)$.

4-й шаг: Выделение ребра минимального веса: (a_{15}, a_{13}) (вес $w(a_i, b) = 1.1$). Интеграция узлов: $J_{a_{15},a_{13}} = a_{15}\&a_{13}$ (исходная часть кластера). Очевидное присоединение узла a_{16} к интегированному узлу для получения сбалансированного (по элементной структуре) кластера $X_4 = \{a_{13}, a_{15}, a_{16}\}$, оценка структуры кластера равна $e(X_2) = (1, 1, 1)$.

5-й шаг: Выделение ребра минимального веса: (a_{10}, a_5) (вес $w(a_i, b) = 1.2$). Интеграция узлов: $J_{a_{10},a_5} = a_{10}\&a_5$ (исходная часть кластера). Очевидное присоединение узла a_9 к интегированному узлу для получения сбалансированного (по элементной структуре) кластера $X_5 = \{a_5, a_9, a_{10}\}$, оценка структуры кластера равна $e(X_5) = (1, 1, 1)$.

Получается дополнительный висячий узел (дополнение в Таблице 24). Полученное дерево представлено на Рис. 13.

6-й шаг: Выделение ребра минимального веса: (a_2, a_1) (вес $w(a_i, b) = 2.5$). Интеграция узлов: $J_{a_2,a_1} = a_2\&a_1$ (исходная часть кластера). Очевидное присоединение узла a_3 к интегированному узлу для получения сбалансированного (по элементной структуре) кластера $X_6 = \{a_1, a_1, a_3\}$, оценка структуры кластера равна $e(X_5) = (2, 1, 0)$.

Здесь построенный кластер не соответствует требуемой элементной структуре.

7-й шаг: Расширение кластера X_1 за счет присоединения отдельного узла a_{19}: $\overline{X}_1 = X_1\&a_{19}$, $e(\overline{X}_1) = (1, 1, 2)$.

Таким образом, получается следующее решение кластеризации:
$\widetilde{X} = \{\overline{X}_1, X_2, X_3, X_4, X_5, X_6\}$ (Рис. 14, Таблица 4).

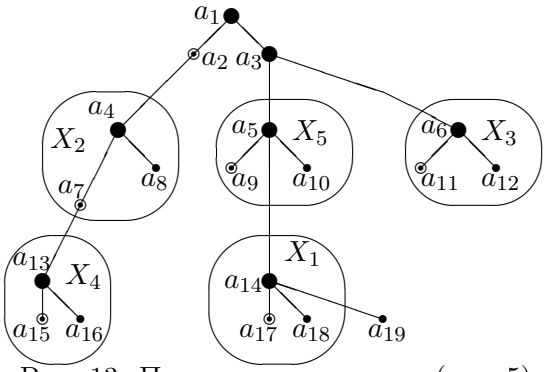

Рис. 13. Покрывающее дерево (шаг 5)

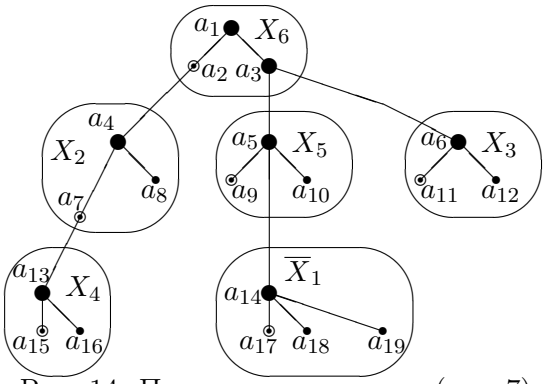

Рис. 14. Покрывающее дерево (шаг 7)

Качество решения балансной кластеризации можно вычислить следующим способом [48,49] (предполагается: $X_1 = \overline{X}_1$):

$$Q^{cb}(\widetilde{X}) = \max_{j=\overline{1,6}} \ \delta(e(X_j), e^0) = \max[\delta(e(X_1), e^0)), \delta(e(X_6), e^0)] = \max[1, 2] = 2.$$

3.3. Прямая стратегия решения

Данная стратеги может быть рассмотрена как иерархическая агломеративная кластеризация (процедура снизу-вверх) с учетом специальных балансных требований (т.е., к размеру кластера, к элементной структуре кластера). Некоторые основные подходы к иерархической кластеризации приведены в Таблице 7.

Таблица 7. Некоторые основные подходы к иерархической кластеризации (обзоры, методы)

Ном.	Исследование	Источник
1.	Общие обзоры по кластеризации	[37]
2.	Обзоры по комбинаторной кластеризации	[46, 47]
3.	Приближенная иерархическая кластеризация	[14]
4.	Иерархическая кластеризация с гарантированной эффективностью	[22]
5.	Иерархическая кластеризация на основе функции стоимости для близости	[21]
6.	Общий подход к иерархической кластеризации и поэтапной аппроксимация	[52]
7.	Иерархическая кластеризация на основе порядковых оценок	[44, 46]
8.	Иерархическая кластеризация на основе Венгерского метода	[32]
9.	Иерархическая концептуальная кластеризация на основе графа	[39]
10.	Иерархическая декомпозиция графа (в сетях)	[62]
11.	Иерархическая кластеризация (в беспроводных сенсорных сетях)	[55]
12.	Иерархическая кластеризация для категорных данных (использование вероятностных rough set моделей)	[51]
13.	Иерархическая кластеризация на основе объединенных расстояний (between-within distances)	[67]
14.	Интервальная иерархическая агломеративная кластеризация	[38]
15.	Многомерная иерархическая кластеризация	[24]

Процедура и числовой пример иерархической балансной кластеризации (с учетом размера кластера) приведены в [46]. При балансном требовании к элементной структуре кластера процедура похожа. С другой стороны, данная стратегия похожа на стратегию 2.2 из предыдущего раздела.

3.4. Построение много-уровневых структур с балансировкой

Здесь очевидная стратегия может быть использована:

Фаза 1. Разбиение исходного множества элементов для получения нескольких подмножеств, каждое из которых соответствует своему уровню. Далее различные процедуры могут быть использованы: (а) выбор элементов, (б) специальные задачи покрытия (например, задача древовидного покрытия с максимизацией числа висячих вершин).

Фаза 2. Решение специальных задач балансной кластеризации на каждом уровне.

Фаза 3. Построение связей между элементами различных уровней.

Иллюстративный пример трехуровневой структуры с балансировкой приведен на Рис. 15 (использованы 4 типа элементов). Несколько схем решения для для построения уровневых структур описаны в [45].

Следует отметить, что кластеры на верхних уровнях сетей (связь, транспорт) часто являются составными центрами (hubs).

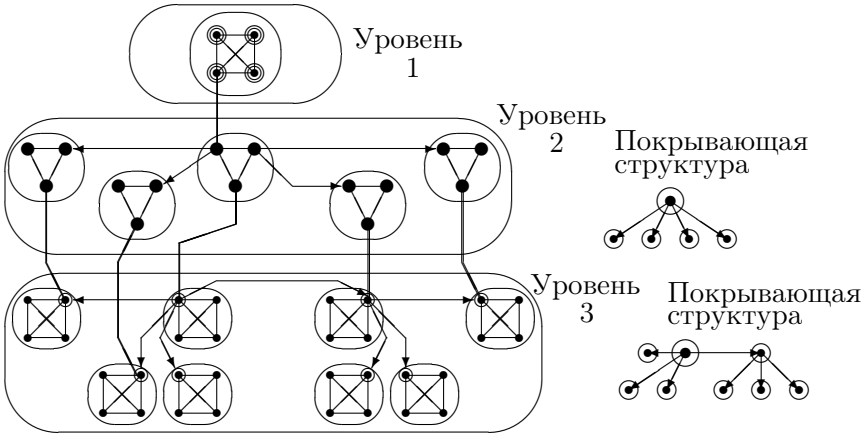

Рис. 15. Пример 3-уровневой структуры с балансировкой

4. ЗАКЛЮЧЕНИЕ

В данной статье представлена задача балансной кластеризации с учетом покрывающего дерева на построенные кластеры. Предложены четыре эвристическое стратегии, основанные на использовании жадного подхода. Примеры иллюстрируют задачи с схемы решения. Можно указать ряд направления для дальнейших исследований; (1) исследование многокритериальных моделей и моделей с неопределенностью. (2) рассмотрение других типов покрывающих структур, например: цепочки, леса, иерархии; (3) использование других подходов к решения задач (например, вероятностных алгоримов); (4) проектирование специального программного комплекса для моделировании и решения рассмотренных задач; (5) оценивание различных методов решения рассмотренных задач с использованием вычислительных экспериментов; (6) исследование приложений в области проектирования и управления различными сетями.

СПИСОК ЛИТЕРАТУРЫ

1. Adelson-Velsky, E.M. Landis E.M., An algorithm for the organization of information. Soviet Math. Doklady, 1962, vol. 3, pp. 1259–1263.
2. Aho A.V., Hopcroft J.E., Ullman J.D., Data Structures and Algorithms. Addison-Wesley, Readings, MA, 1983.
3. Alon N., Fomin F., Gutin G., Krivelevich M., Saurabh S., Spanning directed trees with many leaves. SIAM J. on Discr. Math., 2009, vol. 23, no. 1, pp. 466–476.
4. Andersen A., General balanced trees. J. of Algorithms, 1999, vol. 30, pp. 1–18.
5. Andreev K., Racke H., Balanced graph partitioning. Theory of Comput. Syst., 2006, vol. 39, no. 6, pp. 929–939.
6. Apollonio N., Lari I., Puerto J., Ricca F., Simeone B., Polynomial algorithms for partitioning a tree into single-center subtrees to minimize flat service costs. Networks, 2008, vol. 51, pp. 78–89.
7. Arroyo J.E.C., Vieira P.S., Vianna D.S., A GRAPS algorithm for the multi-criteria minimum spanning tree problem. Ann. of Oper. Res., 2008, vol. 159, no. 1, pp. 125–133.
8. Asano T., Bhattacharya B., Keil M., Yao F., Clustering algorithms based on minimum and maximum spanning trees. In: Symp. on Computational Geometry, pp. 252–257, 1988.
9. Becker R.I., Perl Y., The shifting algorithm techniques for the partitioning of trees. Disc. Appl. Math., 1995, vol. 62, pp. 15–34.
10. Blum N., Mehlhorn K., On the average number of rebalancing operations in weight-balanced trees. Theor. Comp. Sci., 1980, vol. 11, pp. 303–320.

11. Blum J., Ding M., Thaeler A., Cheng X., Connected dominating set in sensor networks and MANETs. In: Du D.-Z., Pardalos P. (eds), Handbook of Combinatorial Optimization, Springer, pp. 329–369, 2005.

12. Camerini P.M., Galbiati G., Maffioli F., On the complexity of finding multi-constrained spanning trees. Discr. Appl. Math., 1983, vol. 5, pp. 39–50.

13. Caro Y., West D.B., Yuster R., Connected domination and spanning trees with many leafs. SIAM J. on Discr. Math., 2000, vol. 13, no. 2, pp. 202–211.

14. Charikar M., Chatziafratis V., Approximate hierarchical clustering via sparsest cut and spreading metrics. In: SODA 2017, pp. 841–854, 2017.

15. Chen T.-S., Tseng Y.-C., Sheu J.-P., Balanced spanning trees in complete and incomplete star graphs. IEEE Trans. on Paral. and Distr. Syst., 1996, vol. 7, no. 7, pp. 717–723.

16. Chen G., Chen S., Guo W., Chen W., The multicriteria minimum spanning tree problem based genetic algorithm, Inform. Sci., 2007, vol. 177, no. 22, pp. 5050–5063.

17. Cheng X., Huang X., Li D., Wu W., Du D.-Z., A polynomial-time approximation scheme for the minimum-connected dominating set in ad hoc wireless networks. Networks, 2003, vol. 42, no. 4, pp. 202–208.

18. Cheriton D., Tarjan R.E., Finding minimum spanning trees. SIAM J. Comput., 1976, vol. 5, pp. 724–742.

19. Choudum S.A., Raman I., Embedding height balanced trees and Fibonacci trees in hypercubes. J. App. Math. Comput., 2009, vol. 30, pp. 39–52.

20. Cormen T.H., Leiserson C.E., Rivest R.L., Introduction to Algorithms. 3rd ed., MIT Press and McGraw-Hill, 2009.

21. Dasgupta S., A cost function for similarity-based hierarchical clustering. In: Proc. of the 48th Annual ACM SIGART Symp. on Theory of Computing STOC 2016, pp. 118–127, 2016.

22. Dasgupta S., Long P.M., Performance guarantees for hierarchical clustering. J. of Comp. and Syst. Sci., 2005, vol. 70, no. 4, pp. 555–569.

23. Dell'Amico M., Maffioli F., Combining linear and non-linear objectives in spanning tree problems. J. of Comb. Optim., 2000, vol. 4, no. 2, pp. 253–269.

24. Dugad R., Ahuja N., Unsupervised multidimensional hierarchical clustering. In: Proc. of the 1998 IEEE Int. Conf. on Acoustic, Speech and Signal Processing, vol. 5, pp. 2761–2764, 1998.

25. Feldmann A.E., Foschini L., Balanced partitions of trees and applications. Algorithmica, 2015, vol. 71, no. 2, pp. 354–376.

26. Fernau H., Kneis J., Kratsch D., Langer A., Liedloff M., Raible D., Rossmanith P., An exact algorithm for the Maximum Leaf Spanning Tree problem. Theor. Comp. Sci., 2011, vol. 412, no. 45, pp. 6290–6302.

27. Fujie T., An exact algorithm for the maximum leaf spanning tree problem. Comp. & Oper. Res., 2003, vol. 30, no. 13, pp. 1931–1944.

28. Gabow H.W., Galil Z., Spencer T., Tarjan R.E., Efficient algorithms for finding minimum spanning trees in undirected and directed graphs. Combinatorica, 1986, vol. 6, no. 2, pp. 109–122.

29. Garey M.R., Johnson D.S., Computers and Intractability. The Guide to the Theory of NP-Completeness. W.H. Freeman and Company, San Francisco, 1979.

30. Georgiadis L., Tarjan R.T., Dominator tree certification and independent spanning trees. Electr. prepr., 44 p., Oct. 31, 2012. http://arxiv.org/abs/1210.8303 [cs.DS]

31. Gfeller B., Faster swap edge computation in minimum diameter spanning trees. Algorithmica, 2012, vol. 62, no. 1–2, pp. 169–191.

32. Goldberger J., Tassa T., A hierarchical clustering algorithm based on the Hungarian method. Patt. Recogn. Lett., 2008, vol. 29, no. 11, pp 1632–1638.

33. Grygorash O., Zhon Y., Jorgensen Z., Minimum spanning tree based clustering algorithms. In: IEEE Int. Conf. on Tools with Artificial Intelligence, pp. 73–81, 2006.

34. Hamaher H.W., Ruhe G., On spanning tree problem with multiple objectives. Ann. of Oper. Res., 1995, vol. 52, no. 4, pp. 209–230.

35. Hirai Y., Yamamoto K., Balancing weight-balanced trees. J. of Funct. Progr., 2011, vol. 21, no. 3, pp. 287–307.

36. Itai A., Rodeh M., The multitree approach to reliability in distributed networks. Inform. and Comput., 1984, vol. 79, no. 1, pp. 43–59.

37. Jain A.K., Murty M.N., Flynn P.J., Data clustering: a review. ACM Comput. Surv., 1999, vol. 31, no. 3, pp. 264–323.

38. Jeng J.-T., Chuang C.-C., Tao C.W., Interval competitive agglomeration clustering algorithm. Exp. Syst. and Appl., 2010, vol. 37, no. 9, pp. 6567–6578.

39. Jonyer I., Cook D.J., Holder L.B., Graph-based hierarchical conceptual clustering. J. of Mach. Learn. Res., 2001, vol. 2, pp. 19–43.

40. Karlton P.L., Fuller S.H., Scroggs R.E., Kaehler E.B., Performance of height-balanced trees. Commun. of the ACM, 1976, vol. 19, no. 1, pp. 23–28.

41. Kleitman D., West D., Spanning trees with many leafs. SIAM J. on Discr. Math., 1991, vol. 4, no. 1, pp. 99–106.

42. Knuth D., The Art of Computing Programming. Vol. 3: Sorting and Searching, 3rd ed., Addison-Wesley, 1997.

43. Konemann J., Levin A., Sinha A., Approximating the degree-bounded minimum diameter spanning tree problem. Algorithmica, 2005, vol. 41, no. 2, pp. 117–129.

44. Levin M.Sh., Towards hierarchical clustering. In: Diekert V., Volkov M., Voronkov A. (eds), Proc. of Int. Conf. Comp. Sci. in Russia CSR-2007, LNCS 4649, Springer, pp. 205–215, 2007.

45. Levin M.Sh., Modular System Design and Evaluation. Sprigner, 2015.

46. Levin M.Sh., Towards combinatorial clustering: preliminary research survey. Electr. prepr., 102 p., May 28, 2015. http://arxiv.org/abs/1505.07872 [cs.AI]

47. Levin M.Sh., On combinatorial clustering: literature review, methods, examples. J. of Commun. Technol. and Electronics, 2015, vol. 60, no. 12, pp. 1403–1428.

48. Levin M.Sh., Towards balanced clustering - part 1 (preliminaries). Electr. prepr., 21 p., Jun. 9, 2017. http://arxiv.org/abs/1706.03065 [cs.DS]

49. Levin M.Sh., On balanced clustering (indices, models, examples). J. of Commun. Technol. and Electronics, 2017, vol. 62, no. 12, pp. 1506–1515.

50. Levin M.Sh., On balanced clustering with tree-like structures over clusters. Electr. prepr., 15 p., Dec. 9, 2018. http://arxiv.org/abs/1812.03535 [cs.DS]

51. Li M., Deng S., Wang L., Feng S., Fan J., Hierarchical clustering algorithm for categorical data using a probabilistic rough set model. Knowledge Based Syst., 204, vol. 65, pp. 60–71.

52. Lin G., Nagarajan C., Rajaraman R., Williamson D.P., A general approach for incremental approximation and hierarchical clustering. SIAM J. on Computing, 2010, vol. 39, no. 8, pp. 3633–3669.

53. Lu H., Ravi R., The power of local optimization: approximation algorithms for maximum leaf spanning tree. In: Proc. of the Annual Allerton Conf. on Commun., Contr. and Comput., USA, vol. 30, pp. 533–542, 1992.

54. Lu H., Ravi R., Approximating maximum leaf spanning trees in almost linear time. J. of Algorithms, 1998, vol. 29, no. 1, pp. 132–141.

55. Lung C.-H., Zhou C., Using hierarchical agglomerative clustering in wireless sensor networks: An energy-efficient and flexible approach. Ad Hoc Netw., 2010, vol. 8, pp. 328–344.

56. Mamada S., Uno T., Makino K., Fujishige S., A tree partitioning problem arising from an evacuation problem in tree dynamic networks. J. Oper. Res. Soc. Jpn., 2005, vol. 48, pp. 196–206.

57. Meyerhenke H., Sanders P., Schulz C., Partitioning (hierarchically clustered) complex networks via size-constrained graph clustering. J. of Heuristics, 2016, vol. 22, no. 5, pp. 759–782.

58. Nievergelt J., Reingold E.M., Binary search trees of bounded balance. SIAM J. on Comput., 1973, vol. 2, no. 1, pp. 33–43.

59. Perl Y., Schach S., Max-min tree partitioning. J. of the ACM, 1981, vol. 28, pp. 5–15.

60. Pettie S., Ramachandran V., An optimal minimum spanning tree algorithm. J. of the ACM, 2002, vol. 49, no. 1, pp. 16–34.

61. Pettie S., Ramachandran V., A randomized time-work optimal parallel algorithm for finding a minimum spanning forest. SIAM J. on Computing, 2002, vol. 31, no. 6, pp. 1876–1895.

62. Racke H., Optimal hierarchical graph decompositions for congestion minimization in network. In: Proc. of the fourteenth Ann. ACM Symp. on Theory of Comput. STOC'08, pp. 255–264, 2008.

63. Ran Y., Chen Z., Tang S., Zhang Z., Primal dual based algorithm for degree-balanced spanning tree problem. Appl. Math. and Comput., 2018, vol. 316, pp. 167–173.

64. Singh A., An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem. Appl. Soft Comput., 2009, vol. 9, no. 2, pp. 625–631.

65. Solis-Oba R., 2-approximation algorithm for finding a spanning tree with maximum number of leafs. In: Bilardi G., Italiano G.F., Pietracaprina A., Pucci G. (Eds.), Proc. of 6th Annual Eur. Symp. on Algorihtms - ESA'98, LNCS 1461, Springer, pp. 441–452, 1998.

66. Stores J.A., An Introduction to Data Structures and Algorithms. Birkhauser, Boston, 2001.

67. Szekely G.J., Rizzo M.L., Hierarchical clustering via joint between-within distances: Extending Ward's minimum variance method. J. of Classification, 2005, vol. 22, no. 2, pp. 151–183.

68. Tarian R.E., Edge-disjoint spanning trees and depth-first search. Acta Informatica, 1976, vol. 6, no. 2, pp. 171–185.

69. Thai M., Wang F., Liu D., Zhu S., Du D., Connected dominating sets in wireless networks with different transmission ranges. IEEE Trans. Mob. Comput., 2007, vol. 6, no. 7, pp. 721–730.

70. Zhang T., Ramakrishnan R., Livny M., BIRCH: A new data clustering algorithm and it applications. Data Mining and Knowledge Discovery, 1997, vol. 1, no. 2, pp. 141–182.

Balanced clustering with tree over clusters

Levin M.Sh.

The article addresses balanced clustering with spanning tree over clusters. This problem can be useful in network applications (design, management, routing). Elements of different type are considered and balanced clustering problems are targeted to clusters while taking into account the cluster element structure. The examined optimization model is based on vector objective function (quality of balanced clustering and quality of spanning tree). Four basic heuristic solving strategies are suggested: balancing-spanning, spanning-balancing, direct solving strategy, design of multi-layer structure with balanced clusters. Four solving scheme of spanning-balancing strategy are described with special numerical illustrative example.

KEYWORDS: balanced clustering, combinatorial optimization, spanning tree, heuristics, networks