数据集的粒化树及其建模应用

闫 林 宋金朋
(河南师范大学计算机与信息工程学院 新乡 453007)

摘 要 通过对数据集的不同划分,得到了基于数据集的粒化树。结合关联元素的信息,建立了基于不同数据集粒化树之间的关联关系,确定了两种粒化树中的两条关联链,促成了它们经关联元素的相互联系。由于每一关联链中的粒从粗到细逐步变化,使得关联元素与粒度的逐步细化密切相关,这是粒计算数据处理模式的体现。相关的结论为人才供求问题的算法描述提供了数学模型,并通过实例予以展示。

关键词 分层算法,粒化树,关联元素,关联关系,关联链
中图法分类号 TP18 **文献标识码** A

Granular Trees Based on Different Data Sets and their Modeling Applications

YAN Lin SONG Jin-peng
(College of Computer and Information Engineering, Henan Normal University, Xinxiang 453007, China)

Abstract By classifying a data set into different partitions, a granular tree based on the data set was obtained. By using the information of associated elements, a relation called an associated relation was introduced, and connections between two granular trees based on different data sets were established, so that two associated chains were identified, which are connected with each other by an associated element. Because the granules in each associated chain gradually change from a coarse state to a fine state, the associated element model closely links with the changes of the granules, which is consistent with the data processing mode of granular computing. Consequently, as the basis of an algorithm, a mathematical model was created. It can be used to describe the relationship between talent supply and demand. Also, an example shows the process of using the model to carry out data processing.

Keywords Hierarchical algorithm, Granular tree, Associated element, Associated relation, Associated chain

1 引言

随着信息科学的发展,新的课题不断出现,粒计算[1,2]的研究就是新课题之一,它追寻的数据处理模式深得研究者的认可,并很快成为信息科学研究的热点。粒计算的探究包括两个方面:粒化和基于粒的计算。直观地讲,粒化是对整体数据的分类,使得满足某种性质的数据以类的形式得以聚拢,并称为粒。所以粒化是从整体中分离出部分的过程,反映了数据整体被分类的特性。粒的计算是通过粒的组合、粒的运算、粒的关系等对数据的处理、描述、转换或约简,目的在于通过粒的计算,简化或明晰整体中的数据处理过程,或使问题通过粒的计算得到顺利解决。粒计算就是探究粒化和粒的计算的课题,被认为是数据处理的一种模式,它将整体与部分相互联系,以寻求从整体到部分,再由部分实施数据处理,达到转换、简化、明确或完成数据处理的目的。

粒计算的提出,源于计算机领域各类研究的支撑,是基于它们共性的课题。这种共性就是从整体中分离出部分或粒,利用粒之间的组合、运算或关系等对数据进行处理。比如:粗糙集理论[3,4]中,基于等价关系的划分确定了从整体中分离出粒的方法,并通过粒的组合对知识进行了近似的描述。模糊集理论[5]中,与隶属函数相关的信息粒,以及基于模糊等价关系截集的分层递阶粒空间[6]等是模糊理论用于数据粒化和数据处理的成果,展示了模糊知识与粒计算的联系。概念格中概念信息粒和概念知识粒[7]、邻域系统中邻域粒子[8,9]、商空间理论[10]中的粒度问题等都体现了整体与部分相互关联的数据处理方式。虽然这些研究来自不同的研究,但它们的共同特性就是大数据的粒化分解,以及基于粒的运算所实施的数据处理。

下面的讨论是从整体到部分的数据处理过程,一致于粒计算的数据处理方式,目的在于描述实际中的分类问题,为数据处理提供算法构建的数学模型。在实际当中,许多问题的刻画体现了粒化的差异。如何通过数学方法描述粒的差异,建立粒之间的联系,由此形成数据处理的粒化模型是本文将要展开的研究。如果对一类数据进行考察,可以发现不同的考察方法与该类数据的不同分类有关,各种考察方法的组合可以使该类数据形成一种数学结构,这就是下面涉及的粒化树的概念。粒化树是本文构建的一种结构性数据,它建立在一类数据之上,同时粒化树与粒计算的数据处理方法相一致,

到稿日期:2013-05-24 返修日期:2013-08-12 本文受河南省自然科学基金(082300410340)资助。
闫 林(1957—),男,教授,主要研究方向为数理逻辑、非经典逻辑、粒计算和粗糙集,E-mail:hnsdyl@163.com;宋金朋(1988—),男,硕士生,主要研究方向为粗糙集、粒计算。

将展示从整体到部分以及进一步层层细化的过程。通过粒化树对一类数据进行处理，可达到描述数据之间相互联系的目的，这些将展示在如下的研究之中。

2 基本概念

为了讨论的需要，先对一些概念进行明确，为研究工作做必要的准备。

定义 1 设 U 是任意一集合，称为数据集，U 中的元素称为数据。$T=\{T_1,T_2,\cdots,T_r\}$ 是 U 的子集组成的集合，即 $T_i\subseteq U,i=1,2,\cdots,r$。如果满足下述条件：

①对于 $i=1,2,\cdots,r$，有 $T_i\neq\emptyset$；

②当 $i\neq j$ 时，$T_i\cap T_j=\emptyset,1\leqslant i,j\leqslant r$；

③$T_1\cup T_2\cup\cdots\cup T_r=U$。

则称 $T=\{T_1,T_2,\cdots,T_r\}$ 是集合 U 的划分，称 T_i 为该划分的划分块。

划分是对数据集 U 的粒化，下面也把划分块 T_i 称为粒，粒中的数据往往具有共同的性质。

定义 2 若 $S=\{S_1,S_2,\cdots,S_n\}$ 为数据集 U 的一种划分，$T=\{T_1,T_2,\cdots,T_k\}$ 为数据集 U 的另一种划分，并且对于任意的 $S_i\in S$，存在 $T_j\in T$，使得 $S_i\subseteq T_j$，则称 S 是 T 的细分。

由划分及细分的定义可知，细分具有传递性，即若 S 是 T 的细分，T 是 W 的细分，则 S 是 W 的细分。细分是对粒进一步粒化的过程，当划分 $S=\{S_1,S_2,\cdots,S_n\}$ 是划分 $T=\{T_1,T_2,\cdots,T_k\}$ 的细分时，如果 $S_i\in S$ 及 $T_j\in T$，且满足 $S_i\subseteq T_j$，则称粒 S_i 比粒 T_j 更精细。

上述定义 1 和定义 2 中的概念可在文献[11]中找到，将其列出是基于讨论整体性的考虑。由于下面的工作将把数据集 U 与 U 的划分联系在一起作为整体，因此有必要给出描述结构的定义。

定义 3 设 U 是任一有限数据集，R 是 U 上的二元关系，即 $R\subseteq U\times U$。若 $\langle a,b\rangle\in R$，则 a 称为 b 的前驱，b 称为 a 的后继。将 U 和 R 组成的整体记作 $T=(U,R)$，称为描述结构。

对于 U 上二元关系 R，当 R 满足一定条件时，描述结构 $T=(U,R)$ 将展示相应的知识概念，如下的树就是一种描述结构。

定义 4 设 $T=(U,R)$ 是一描述结构，如果满足以下条件，则称 $T=(U,R)$ 为一棵树：

①有且仅有一个数据 $a_0\in U$，a_0 没有前驱，即对任意的 $x\in U$，当 $x\neq a_0$ 时，有 $\langle x,a_0\rangle\notin R$，$a_0$ 称作根；

②除根 a_0 外，U 中的每个数据有且仅有一个前驱，即对任意的 $y\in U$，如果 $y\neq a_0$，则存在唯一的 $x\in U$，使得 $\langle x,y\rangle\in R$；

③U 中每个数据可以有 0 个后继，或 1 个以及 1 个以上的后继。

显然，树 $T=(U,R)$ 是一种描述结构，R 满足定义 4 中①、②和③中的性质。

树中的数据具有层次，同时树也有高度。

数据的层次：根的层次为 0，其余数据的层次为其前驱数据的层次加 1。

树的高度：树中结点数据的最大层次。

3 基于数据集的粒化树

树 $T=(U,R)$ 是计算机可接受处理的数据结构，并广泛应用于计算机科学之中。在程序设计中，树可描述程序的语法结构；在操作系统中，树可表示文件的组织结构；在数据库研究中，树形数据库展示了数据关联的分层结构等等。另一方面，树也常常关联于实际中的许多问题，如家族的家谱、机构人员的隶属、国土面积的省市县划分等等均呈现树形结构。因此各类问题的树形表示是问题数据处理的数学模型。以下讨论将基于一有限数据集，通过对该数据集进行划分及层层细分促成树的产生，并体现逐步粒化的思想。

设 K 是一有限数据集，有目的对 K 划分并逐步细分可得到划分的不同层次，各层次的组合可形成树，该树可由一算法产生。考虑如下算法：

分层算法：

①输入一个正整数 n；

②令 $U=\{K\}$，对 K 有目的划分，得到 K 的一个划分 $\{A_1,A_2,\cdots,A_n\}$，令 $U=U\cup\{A_1,A_2,\cdots,A_n\}$，置 $i=1$，记 $S_i=\{A_1,A_2,\cdots,A_n\}$，称 S_i 为 K 的第 i 层划分；

③对 S_i 有目的细分，得到 $\{B_1,B_2,\cdots,B_m\}$，该细分 $\{B_1,B_2,\cdots,B_m\}$ 仍是 K 的一个划分，令 $U=U\cup\{B_1,B_2,\cdots,B_m\}$，置 $i=i+1$，记 $S_i=\{B_1,B_2,\cdots,B_m\}$，此时 S_i 为 K 的第 i 层划分；

④若 $i=n$，则算法停止，U 确定。若 $i\neq n$，转③继续执行。

分层算法①中的正整数 n 表示对 K 实施 n 次划分或细分。对于自然数 $i(1\leqslant i\leqslant n)$，第 i 层划分 S_i 是对 K 有目的划分或细分，其中“有目的划分或细分”由 K 中元素的性质确定，比如 K 是某高校全体学生的集合，根据院系的划分就是一种有目的划分，根据学生籍贯的划分也是一种有目的划分，等等。根据细分的传递性知，当 S_j 为 S_i 的上层时，即 $j<i$ 时，S_i 为 S_j 的细分，该细分也具有一定目的，比如 S_j 是根据院系产生的划分，S_i 可以是通过院系中年级的划分所产生的细分，也可以是利用院系中学生的籍贯所产生的细分，等等。由分层算法得到的 U 是集合 $\{K\}$ 与各层划分的并集，比如 $n=5$ 时，通过分层算法可得到 U，此时 $U=\{K\}\cup S_1\cup S_2\cup S_3\cup S_4\cup S_5$，其中 $S_i(i=1,\cdots,5)$ 都是 K 的划分，当 $j<i$ 时，S_i 是 S_j 的细分。以下称 U 是基于 K 的分层粒化集，它由分层算法生成。

上述把划分中的每一划分块称为粒，基于 K 的分层粒化集 U 中的每一元素都是 K 的某一划分中的划分块，当然也是 K 的子集，即 $A\in U$ 时，有 $A\subseteq K$。若把 K 看作整体，则 A 是整体的部分，下面的定义仍把 A 称为粒，体现了从整体中分离出部分的粒化思想。同时，当 $A,B\in U$ 时，可能有 $A\subseteq B$ 或 $B\subseteq A$，由此可以在 U 上引入二元关系 P，使得 U 和 P 组成描述结构，形成进一步研究的基础。考虑如下定义：

定义 5 设 K 是一有限数据集，U 是由分层算法产生的基于 K 的分层粒化集。对于 $A\in U$，称 A 为 K 的粒。对于任意的粒 $A,B\in U$，A 和 B 之间的关系可确定 U 上偏序关系 P，定义如下：

$\langle A,B\rangle\in P$ 当且仅当 $B\subseteq A$。

称 P 为基于 K 的粒包含关系，将 U 和 P 构成的描述结构记作 $T(K)=(U,P)$，依据分层算法中的 n 值，称 $T(K)$ 为基于 K 的 n 层粒化树，简称为基于 K 的粒化树。

基于 K 的粒化树 $T(K)=(U,P)$ 与 K 联系紧密，$T(K)$ 中的 K 体现了 $T(K)$ 对 K 的依赖。之所以将该描述结构称

为粒化树，是因为 $T(K)=(U,P)$ 是一棵树，实际上，对于任意的 $A\in U$，有 $A\subseteq K$，由定义 5，$\langle K,A\rangle\in P$，所以 K 是 $T(K)$ 的根，且 $T(K)$ 满足树的其他条件（见定义 4），因此 $T(K)$ 是树。比如针对数据集 $K=\{1,2,3,4,5,6,7,8,9\}$，在分层算法中设定 $n=3$，执行分层算法得到：

$S_1=\{\{1,2,3,4,5\},\{6,7,8,9\}\}$

$S_2=\{\{1,2,3\},\{4,5\},\{6,7,8,9\}\}$

$S_3=\{\{1,2\},\{3\},\{4,5\},\{6,7\},\{8,9\}\}$

S_1，S_2 和 S_3 均为 K 的划分。此时，$U=\{K\}\cup S_1\cup S_2\cup S_3$。这里的各层划分 S_1，S_2 和 S_3 均具有一定目的，体现了分层算法中有目的划分的思想。$T(K)=(U,P)$ 表示为如图 1 所示的树形结构。

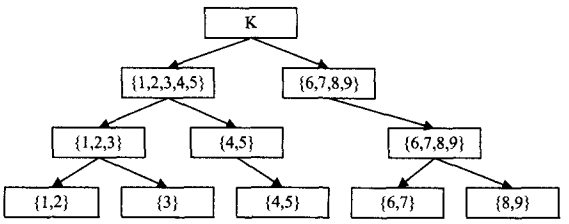

图 1　基于 K 的 3 层粒化树 $T(K)=(U,P)$

其中，K 是该树的根，且当粒 $A,B\in U$ 及 $\langle A,B\rangle\in P$（即 $B\subseteq A$）时，在图 1 中从 A 到 B 存在有向连线。该树的每一层 $S_i(i=1,2,3)$ 均为 K 的划分，当 S_j 为 S_i 的上层即 $j<i$ 时，S_i 为 S_j 的细分，比如 S_3 是 S_2 的细分，同样也是 S_1 的细分。观察 S_1，其中含有两个粒 $\{1,2,3,4,5\}$ 和 $\{6,7,8,9\}$，每一粒中的数据由相关的性质决定，比如粒 $\{1,2,3,4,5\}$ 中的数据 1，2，3，4，5 可认为具有共同的性质，粒 $\{6,7,8,9\}$ 中的数据 6，7，8，9 也具有共同的性质。有目的划分就是根据数据性质实施的划分，这是对分层算法中“有目的划分”的解释。在执行分层算法时，可通过交互确定元素性质，以实现有目的划分的意图。

4　关联元素与关联链

对于有限数据集 K，通过分层算法得到基于 K 的分层粒化集 U，再结合基于 K 的粒包含关系 P，便确定了基于 K 的粒化树 $T(K)=(U,P)$。对于任意的粒 $A\in U$，则有 $A=K$ 或者 $A\in S_i(1\leqslant i\leqslant n)$，同时 $A\subseteq K$。

定义 6　设 K 是一有限数据集，$T(K)=(U,P)$ 是基于 K 的粒化树，若 $L\subseteq U$ 且对任意的粒 $A,B\in L$，有 $\langle A,B\rangle\in P$ 或 $\langle B,A\rangle\in P$，则称 L 为 $T(K)=(U,P)$ 的一条链。

直观地讲，$T(K)=(U,P)$ 的一条链是从 K 的某粒出发沿有向连线行至另一粒的路径上所有粒的集合。从定义 6 可知，$T(K)=(U,P)$ 的链 L 不仅是 U 的子集，且与基于 K 的粒包含关系 P 相关，为了表明 L 依赖于 P，链 L 一般记作 $L(P)$。例如在图 1 中，$L(P)=\{\{1,2,3,4,5\},\{1,2,3\},\{1,2\}\}$ 是 $T(K)=(U,P)$ 的一条链，图 1 中还存在其它的链。此外，在图 1 基于 K 的粒化树 $T(K)=(U,P)$ 中，不同的层中有相同的粒，这对于划分或细分是允许的，图 1 中的画法显示了这样的事实：树中的每一层都是 K 的一种划分。

基于 K 的粒化树 $T(K)=(U,P)$ 由数据集 K 确定，当 K 改变时，$T(K)=(U,P)$ 也将随之改变，因此可以考虑不同的数据集。设 K_1 和 K_2 是两个不同的数据集，且满足 $K_1\cap K_2\neq\varnothing$，当 $a\in K_1\cap K_2$ 时，下述定义称 a 为关联元素。对于基于 K_1 的粒化树 $T(K_1)=(U_1,P_1)$ 和基于 K_2 的粒化树 $T(K_2)=(U_2,P_2)$，当粒 $A\in U_1$ 且粒 $B\in U_2$ 时，有 $A\subseteq K_1$ 和 $B\subseteq K_2$，由于 $K_1\cap K_2\neq\varnothing$，因此可能有 $A\cap B\neq\varnothing$。由此引出如下定义：

定义 7　设 K_1,K_2 是两个不同的数据集，$K_1\cap K_2\neq\varnothing$，当 $a\in K_1\cap K_2$ 时，称 a 为关联元素。对于基于 K_1 和 K_2 的粒化树 $T(K_1)=(U_1,P_1)$ 和 $T(K_2)=(U_2,P_2)$，定义从 U_1 到 U_2 的关系 H 如下：

对于粒 $A\in U_1$ 及粒 $B\in U_2$，$\langle A,B\rangle\in H$ 当且仅当 $A\cap B\neq\varnothing$。称关系 H 为 $T(K_1)=(U_1,P_1)$ 和 $T(K_2)=(U_2,P_2)$ 之间的关联关系。

从关联关系 H 的定义可知，如果 $\langle A,B\rangle\in H$，则粒 A 与粒 B 中存在公共元素，即存在关联元素 $a\in K_1\cap K_2$，使得 $a\in A\cap B$，粒 A 与粒 B 通过关联元素 a 产生了关联。同时通过关联关系 H，可以找到关联元素 a，使 $T(K_1)=(U_1,P_1)$ 的某条链和 $T(K_2)=(U_2,P_2)$ 的某条链之间通过关联元素 a 相互联系。

定理 1　设 H 为 $T(K_1)=(U_1,P_1)$ 和 $T(K_2)=(U_2,P_2)$ 之间的关联关系，对于粒 $A\in U_1$ 及粒 $B\in U_2$，如果 $\langle A,B\rangle\in H$，则必有关联元素 a，满足如下性质：

存在 $T(K_1)=(U_1,P_1)$ 的链 $L(P_1)$ 和 $T(K_2)=(U_2,P_2)$ 的链 $L(P_2)$，使得 $A\in L(P_1)$ 及 $B\in L(P_2)$，且对于任意的粒 $E\in L(P_1)$ 和粒 $F\in L(P_2)$，有 $a\in E\cap F$。

证明：因为 $\langle A,B\rangle\in H$，所以 $A\cap B\neq\varnothing$，令 $a\in A\cap B$，则 $a\in K_1\cap K_2$，所以 a 是关联元素。定义 U_1 的子集 $L(P_1)$ 和 U_2 的子集 $L(P_2)$ 如下：

$L(P_1)=\{E\mid E\in U_1$ 且 $E\subseteq A$ 及 $a\in E\}$

$L(P_2)=\{F\mid F\in U_2$ 且 $F\subseteq B$ 及 $a\in F\}$

由于 $A\subseteq A$ 及 $a\in A$，并且 $B\subseteq B$ 及 $a\in B$，由 $L(P_1)$ 和 $L(P_2)$ 的定义可知 $A\in L(P_1)$ 且 $B\in L(P_2)$。下证 $L(P_1)$ 和 $L(P_2)$ 分别是 $T(K_1)=(U_1,P_1)$ 和 $T(K_2)=(U_2,P_2)$ 的链：

对于 $L(P_1)$，显然 $L(P_1)\subseteq U_1$。其次对于任意的粒 E_1，$E_2\in L(P_1)$，如果 $E_1=E_2$，则 $E_1\subseteq E_2$，由定义 5 可知 $\langle E_2,E_1\rangle\in P_1$。如果 $E_1\neq E_2$，因为 $a\in E_1$ 且 $a\in E_2$，所以关联元素 a 是 E_1 和 E_2 的公共元素。由于粒化树 $T(K_1)$ 同一层次中的划分块（或粒）构成 K_1 的划分，其任意两个不同的划分块（或粒）无公共元素，因此由分层算法行进的结果，E_1 和 E_2 一定位于不同的层次。分层算法的步骤顺序表明，下层是上层的细分。由于 $E_1\cap E_2\neq\varnothing$ 且 E_1 和 E_2 位于不同的层次，因此必有 $E_2\subseteq E_1$ 或 $E_1\subseteq E_2$，即 $\langle E_1,E_2\rangle\in P_1$ 或 $\langle E_2,E_1\rangle\in P_1$，由定义 6 可知 $L(P_1)$ 是 $T(K_1)=(U_1,P_1)$ 的链。同理可证 $L(P_2)$ 也是 $T(K_2)=(U_2,P_2)$ 的链。

对于任意的粒 $E\in L(P_1)$ 和粒 $F\in L(P_2)$，由 $L(P_1)$ 和 $L(P_2)$ 的构造可知，有 $a\in E$ 且 $a\in F$，于是推得 $a\in E\cap F$。

当 $\langle A,B\rangle\in H$ 时，称 A 和 B 是关联粒，并把定理 1 证明中的链 $L(P_1)$ 和 $L(P_2)$ 称为两条关联链，那么定理 1 是从关联粒 A 和 B 出发，寻求关联元素 a，使得 a 出现于两条关联链中的每一个粒中，展示了从关联关系出发，经关联链，寻找关联元素的过程。另一方面，若 b 是关联元素，即 $b\in K_1\cap K_2$，从关联元素 b 出发，可以找到两条关联链，使得一条链中的任意的粒与另一条链中的任意的粒具有关联关系。

定理 2　设 H 为 $T(K_1)=(U_1,P_1)$ 和 $T(K_2)=(U_2,P_2)$

之间的关联关系，对于任意的关联元素 $b\in K_1\cap K_2$，存在 $T(K_1)=(U_1,P_1)$ 和 $T(K_2)=(U_2,P_2)$ 的链 $L(P_1)$ 和 $L(P_2)$，使得任意的 $A\in L(P_1)$ 和 $B\in L(P_2)$，有 $\langle A,B\rangle\in H$，此时 $b\in A\cap B$，于是 $L(P_1)$ 和 $L(P_2)$ 是两条关联链。

证明：定义 U_1 的子集 $L(P_1)$ 和 U_2 的子集 $L(P_2)$ 如下：

$L(P_1)=\{E|E\in U_1$ 且 $b\in E\}$

$L(P_2)=\{F|F\in U_2$ 且 $b\in F\}$。

显然 $L(P_1)\subseteq U_1$ 且 $L(P_2)\subseteq U_2$。

对于任意 $E_1,E_2\in L(P_1)$，由于 $b\in E_1$ 且 $b\in E_2$，同定理 1 的证明，有 $\langle E_1,E_2\rangle\in P_1$ 或 $\langle E_2,E_1\rangle\in P_1$，因此 $L(P_1)$ 是 $T(K_1)=(U_1,P_1)$ 的链。同理 $L(P_2)$ 是 $T(K_2)=(U_2,P_2)$ 的链。由链 $L(P_1)$ 和 $L(P_2)$ 的定义可知，对于任意的粒 $A\in L(P_1)$ 以及任意的粒 $B\in L(P_2)$，有 $b\in A\cap B$，所以 $\langle A,B\rangle\in H$，此时 $L(P_1)$ 和 $L(P_2)$ 是两条关联链。

该定理是从关联元素 b 出发，寻找两条关联链 $L(P_1)$ 和 $L(P_2)$，使得任意的粒 $A\in L(P_1)$ 和粒 $B\in L(P_2)$，均有 $\langle A,B\rangle\in H$，即 A 和 B 是关联粒，并且 $b\in A\cap B$。这展示了从关联元素出发，经关联链，寻求满足关联关系的关联粒的过程。

观察定理 1 和定理 2 可以看到，前者利用关联粒确定关联元素，后者利用关联元素确定关联粒，它们是互逆的过程，并且都涉及了两条关联链。这些讨论为实际问题的算法处理提供了数学支撑，是程序设计的顶层工作。

5 具体问题的数学模型

上述定理 1 和定理 2 的讨论中，均把基于 K_1 的粒化树 $T(K_1)=(U_1,P_1)$ 和基于 K_2 的粒化树 $T(K_2)=(U_2,P_2)$ 联系在一起作为结论产生的支撑。实际当中，某些数据集之间也相互联系，数据的性质常常反映出粒化树的特性，因此上述的讨论可以作为具体问题的数学模型，或为实际问题的数据处理提供理论方法。不妨考虑如下具体例子：

例 地方政府与专业人才之间联系渠道的数学模型问题：设 K_1 是某地区出生人员的集合，包括目前生活居住在该地区的人员，以及因为求学、就业等离开该地区的人员。设 K_2 是某地区一些高校近些年来在校或毕业的学生集合，某些高校可以是该地区的一所、多所或全部，近些年可以确定为 5 年、7 年或 10 年等等，可根据情况而定。因此 K_1 和 K_2 是两个数据集，涉及两类人员，一类与某地区有关，一类与相关高校有关，现假设 $K_1\cap K_2\neq\emptyset$，则可展开如下的讨论。

对于数据集 K_1，把 K_1 涉及的地区更具体地设定为某县，则 K_1 是目前生活居住在该县以及因为求学、就业等离开该县的人员集合。将分层算法应用于 K_1，设定 $n=5$，通过分层算法得到基于 K_1 的分层粒化集 $U_1=\{K_1\}\cup S_1\cup S_2\cup S_3\cup S_4\cup S_5$，其中 S_1 是根据人员隶属的乡镇对 K_1 的划分，根据乡镇的划分对应于分层算法中有目的划分的表述；S_2 是通过行政村对 S_1 的细分；S_3 是通过村民组对 S_2 的细分；S_4 是通过性别对 S_3 的细分；S_5 是通过年龄段对 S_4 的细分，年龄段可以按照少年、青年、中壮年或老年等进行分类。这些细分是有目的进行的，是分层算法中有目的细分的具体体现。这样按照定义 5，得到基于 K_1 的粒包含关系 P_1，从而产生基于 K_1 的粒化树 $T(K_1)=(U_1,P_1)$。

对于数据集 K_2，假设 K_2 是 5 所高校近 5 年来毕业和在校的学生集合，5 年内入校和毕业的学生共 10 届，当然可以更改 5 年的期限，使数据集 K_2 产生变化。将分层算法应用于 K_2，设定 $n=7$，由此得到基于 K_2 的分层粒化集 $U_2=\{K_2\}\cup V_1\cup V_2\cup V_3\cup V_4\cup V_5\cup V_6\cup V_7$，这里采用 V_i 是为了与上述 S_i 进行区分，其中 V_1 是通过学校对 K_2 的划分；V_2 是通过院系对 V_1 的细分；V_3 是通过年级对 V_2 的细分，年级按 08 级、09 级…等划分；V_4 是通过专业对 V_3 的细分；V_5 是通过班级对 V_4 的细分；V_6 是通过性别对 V_5 的细分；V_7 是通过同乡对 V_6 的细分。由定义 5，得到基于 K_2 的粒包含关系 P_2，并产生基于 K_2 的粒化树 $T(K_2)=(U_2,P_2)$。

当 $K_1\cap K_2\neq\emptyset$ 时，由定义 7，在 $T(K_1)=(U_1,P_1)$ 和 $T(K_2)=(U_2,P_2)$ 之间可以建立关联关系 H。如果存在粒 $A\in U_1$ 和粒 $B\in U_2$，满足 $\langle A,B\rangle\in H$，则依据定理 1，必有关联元素 a，以及 $T(K_1)=(U_1,P_1)$ 的链 $L(P_1)$ 和 $T(K_2)=(U_2,P_2)$ 的链 $L(P_2)$，使得 $A\in L(P_1)$ 及 $B\in L(P_2)$，且对于任意的粒 $E\in L(P_1)$ 和粒 $F\in L(P_2)$，有 $a\in E\cap F$，此时 $L(P_1)$ 和 $L(P_2)$ 是两条关联链。从定理 1 的证明过程可知，对任意的 $E\in L(P_1)$，有 $E\subseteq A$，所以 A 可以看作链 $L(P_1)$ 的最大元，同样 B 也可以看作链 $L(P_2)$ 的最大元。因此对于粒 $A\in U_1$ 和粒 $B\in U_2$，当 $\langle A,B\rangle\in H$ 时，可以分别找到它们下层的粒 $E(\subseteq A)$ 和粒 $F(\subseteq B)$，使得粒 E 和粒 F 共同含有关联元素 a。这具有实际的意义，因为随着分层算法中划分层数的增加，粒的相似性增强，元素之间的联系更加紧密，相似性更加明确。对于基于 K_1 和 K_2 的分层粒化集 $U_1=\{K_1\}\cup S_1\cup S_2\cup S_3\cup S_4\cup S_5$ 以及 $U_2=\{K_2\}\cup V_1\cup V_2\cup V_3\cup V_4\cup V_5\cup V_6\cup V_7$，假如 A 是 S_1 中的粒，即 A 是某乡镇人员的集合，B 是 V_1 中的粒，即 B 是某高校近 5 年来毕业或在校学生的集合。此时当 $\langle A,B\rangle\in H$ 时，定理 1 中的两条关联链 $L(P_1)$ 和 $L(P_2)$ 保证了存在比粒 A 更精细的粒 $E\subseteq A$ 和比粒 B 更精细的粒 $F\subseteq B$，使得 E 和 F 含有关联元素 a。该关联元素架起了以粒 E 为单位的人员群体和以粒 F 为单位的人员群体之间的桥梁，它可为地方政府的人才引进、经济发展、问题征询等提供有价值的信息，同时也可以给予高校专业人员在工作就业、家乡建设、个人爱好等方面的指导。

另一方面，对于一关联元素 $b\in K_1\cap K_2$，由定理 2，存在 $T(K_1)=(U_1,P_1)$ 的链 $L(P_1)$ 和 $T(K_2)=(U_2,P_2)$ 的链 $L(P_2)$，使得任意的 $A\in L(P_1)$ 和 $B\in L(P_2)$，有 $b\in A\cap B$。这表明对 K_1 划分并逐步细分后，b 位于基于 K_1 的粒化树的一条链的每一粒中，该链涉及的范围与 b 在家乡感情方面具有千丝万缕的联系，在政治、经济和文化等方面也都相互影响。同时 b 又位于基于 K_2 的粒化树的一条链上，该链给出了 b 就读或毕业的高校、院系、专业和性别等信息。因此，如果 b 是关联元素，则从 b 出发，可将基于 K_1 的粒化树的一条链与基于 K_2 的粒化树的一条链联系在一起，由于 b 关联于这两条链上的每一粒，因此关联元素 b 确定的两条关联链提供了行政划分从整体到局部与专业人员从宽泛到精专的关联信息，使得地方的政治、经济或文化与人才的就业、服务或咨询等相互联系在一起。

这些讨论的重要之处在于将地方政府与专业人才之间的联系通过数据集的粒化树以及关联关系予以了数学描述，为算法的构建提供了数学模型。当数据集十分庞大时，基于数

据集的粒化树是对数据集的层层粒化表示。如果一个粒中的数据看作近似相同，则粒化集的每一层是对数据集的近似表示，粒化树使之进一步结构化。这样的数据处理方式与粒计算的数据处理模式相一致，是粒计算方法的体现，更为数据处理的程序化提供了顶层支撑。据此构建算法并程序化后，可利用计算机寻找关联元素和关联链，并由计算机提供人才供需的相关信息，从而达到数据自动处理的目的。

结束语 基于数据集的粒化树是对数据集的逐步划分与细化，这促成了关联元素与关联链之间的相互依存，以及粒度变化与数据关联程度相互联系的数学描述，使得关联链之间的关联元素得以确定，以及通过关联元素寻求关联链的想法成为了可能。因此，本文的讨论提供了针对关联数据处理的一种方法，这种方法一致于粒计算的数据处理模式，是粒计算用于数据处理的尝试。同时，由于树是计算机可以接受处理的数据结构，并能通过算法予以模拟，因而上述的讨论是以粒计算的数据处理模式为依托，探究如何描述关联数据，以及研究如何产生算法的数学基础的过程。

在实际应用中，问题涉及的数据集可能超出两个，形成多个数据集的局面。是否可以利用本文的方法同时处理多个数据集，按照不同的分类规则分别生成粒化树，描述它们之间潜在的联系，建立多维度的数学模型，是下一步将要面对的问题。从粒计算的角度考虑，大数据的分解是今后数据处理的重要方面，是众多学者关注的方向，也可考虑利用粒化树的方法予以研究。目前可以肯定的是，本文的讨论为今后的工作做了铺垫，基于粒化树的数据处理方法与多个数据集相结合的探究可作为今后努力的目标。

参 考 文 献

[1] Lin Tong-yan. Granular computing on binary relations II: rough set representations and belief functions[C]// Rough Sets and Knowledge Discovery. 1998:122-140

[2] Lin Tong-yan. Granular computing II: infrastructures for AI-Engineering[C]// Proceedings of 2006 IEEE International Conference on Granular Computing. Atlanta, USA, 2006:2-7

[3] Yao Yi-yu. A partition model of granular computing[J]. LNCS Transactions on Rough Sets, 2004(1):232-253

[4] 苗夺谦，李道国. 粗糙集理论、算法与应用[M]. 北京：清华大学出版社，2008

[5] Zadeh L A. Fuzzy sets and information granulation [M]// Advances in Fuzzy Set Theory and Applications. Amsterdam: North-Holland Publishing, 1979

[6] Zhang Ling, Zhang Bo. Theory of fuzzy quotient space(methods of fuzzy granular computing)[J]. Journal of Software, 2003, 14(4):770-776

[7] Qiu Guo-fang, Chen Jin. Concept knowledge system and concept information granular lattice[J]. Chinese Journal of Engineering Mathematics, 2005, 22(6):963-969

[8] Lin Tong-yan. Neighborhood systems and relational database [C]// Proceedings of CSC, 88. New-York, 1988

[9] 胡清华，于达仁，谢宗霞. 基于邻域粒化和粗糙逼近的数值属性约简[J]. 软件学报，2008，19(3)：640-649

[10] 张燕平，张铃，夏莹. 商空间理论与粗糙集的比较[J]. 微机发展，2004，14(10)：21-24

[11] 闫林. 数理逻辑基础与粒计算[M]. 北京：科学出版社，2007：155-160

(上接第 252 页)

所有 NE 重复模式。由于算法中只使用了 QSA 数组，空间效率得到了提高。在分类属性数据样本集上进行的实验表明，算法对生物序列尤其是 DNA 序列以及维吾尔语 Web 文本中 NE 重复模式的识别都很有效。

如前所述，本文算法 RPT 只输出 NE 重复模式，但如果实际应用需要，则只需一个线性时间算法就可输出所有类型的重复模式。约束条件最小周期 p_{min} 和最大间距 g_{max} 的引入，使得使用算法时，可根据应用环境的要求和实际的需要，对某一具体应用所涉及的数据提出不同的约束性条件。

由于使用相似的处理步骤，本文算法可方便地扩展到基于重复模式的聚类算法研究、舆情热点发现、生物序列中公共模体发现、代码克隆检测、数据压缩等方面的研究中。

参 考 文 献

[1] Benson G. Tandem repeats finder: a program to analyze DNA sequences[J]. Nucleic Acids Research, 1999, 27(2):573-580

[2] Lander E S, Linton L M, Birren B, et al. Initial Sequencing and Analysis of the Human Genome[J]. Nature, 2001, 409(6822):860-921

[3] Price A L, Jones N C, Pevzner P A. De novo identification of repeat families in large genomes[J]. Bioinformatics, 2005, 21(Suppl.):351-358

[4] 霍红卫，王小武. DNA 序列中基于适应性后缀树的重复体识别算法[J]. 计算机学报，2010，33(4)：747-754

[5] 纪震，周家锐，姜来，等. DNA 序列数据压缩技术综述[J]. 电子学报，2010，38(5)：1113-1121

[6] Franek F, Smyth W F, Tang Yu-dong. Computing all repeats using suffix arrays[J]. Automata, Languages and Combinatorics, 2003, 8(4):579-591

[7] Narisawa K, Inenaga S, Bannai H, et al. Efficient computation of substring equivalence classes with suffix arrays[C]// Proc. 18th Annual Symp. Combinatorial Pattern Matching. 2007:340-351

[8] Puglisi S J, Smyth W F, Yusufu M. Fast optimal algorithms for computing all the repeats in a string[J]. Mathematics in Computer Science, 2010, 3(4):373-389

[9] 胡吉祥，许洪波，刘悦，等. 重复串特征提取算法及其在文本聚类中的应用[J]. 计算机工程，2007，33(2)：65-67

[10] Franek F, Holub J, Smyth W F, et al. Computing quasi suffix arrays[J]. J. Automata, Languages & Combinatorics, 2003, 8(4):593-606