论文

泛知识化三维 GIS 表达模型(UKRM)

张树清^{①②*}, 周成虎^{②③}, 张俊岩^①, 陈祥葱^{①②}

① 中国科学院东北地理与农业生态研究所, 长春 130102;

② 中国科学院大学, 北京 100049;

③ 中国科学院地理科学与资源研究所, 资源与环境信息系统国家重点实验室, 北京 100101

* E-mail: zhangshuqing@neigae.ac.cn

收稿日期: 2015-04-10; 接受日期: 2015-09-06; 网络版发表日期: 2016-01-26

国家自然科学基金项目(批准号: 41271196)和中国科学院重点部署项目(编号: KZZD-EW-07, KZZD-EW-07-02)资助

摘要 面对复杂、多样化的三维世界, 现有的三维 GIS 模型面临着兼容性差、表达类型不足、数据表达效率低等问题, 难以满足多用途(户), 尤其是海量对象, GIS 空间管理、查询与分析的需求. 本文根据人类认知思维特征, 整合并拓展了与人类认知方式一致的离散表达方式(包括离散面(DCS)、离散体(DB)等), 提出并集成了函数、映射两种新的结构类型, 构建灵活开放、可扩展性强的泛知识化表达模型(UKRM), 实现多种表示方式(包括边界表达 B-rep、构造实体几何 CSG、参数/函数等)的异质性归一化表达, 从而极大提高了 GIS 对海量、复杂、多样化三维世界的表达能力, 实现了 GIS 数据的重用, 并建立了静态 GIS 向动态 GIS 拓展的接口. 初步应用试验表明, 无论规则还是非规则几何体, UKRM 模型表达简洁、高效, 优于现有构模方式(如 IFC, CityGML).

关键词 离散面(DCS), 离散体(DB), 离散结构, 函数结构, 映射结构, “2D/3D”一体化表达, 3D GIS数据模型, UKRM模型

1 引言

二维GIS始于20世纪60年代的机助制图, 如今已深入到社会的各行各业中, 如土地管理、市政、水利、消防、交通以及城市规划等(肖乐斌等, 1998). 二维空间的GIS数据结构是将要描述的空间对象形状采用微分理论将其原型离散后对其数据进行存储和应用的, 其理论与应用已趋于成熟. 然而, 真实的世界是三维、甚至是四维的, 如不仅有房屋、建筑、地质体等

三维静态物体, 还存在诸如车辆、船舶、飞机、无人飞行器、卫星和台风、泥石流等人造及自然动态目标, 因而由二维GIS向三维GIS以及时态GIS拓展是GIS发展的必然趋势(袁林旺等, 2010). 当前静态的、符号化的2D GIS及据此拓展而来的3D GIS, 尽管在可视化等方面取得了进展, 但传统地图的GIS方法所描述的数字空间跟真实地理空间及人们认识空间之间的矛盾日益突出(朱庆, 2014), 随着空间数据获取、计算存储能力的提高, 以及应用的深入和需求的扩大, GIS

中文引用格式: 张树清, 周成虎, 张俊岩, 陈祥葱. 2016. 泛知识化三维 GIS 表达模型(UKRM). 中国科学: 地球科学, 46: 214-228, doi: 10.1360/N072014-00488

英文引用格式: Zhang S Q, Zhou C H, Zhang J Y, Chen X C. 2016. A ubiquitous knowledgeable data representation model (UKRM) for three-dimensional geographic information systems (3D GIS). Sci China Earth Sci, doi: 10.1007/s11430-015-5255-7

自身的局限性愈加明显,难以满足现代社会对三维世界大场景、高动态、多样化的应用需求。

三维空间数据模型及数据结构是以数字方式描述空间对象及其关系,它是3D GIS建立的基础(Shi等, 2003; Gong等, 2004; Zhou等, 2008)。作为数字虚拟世界及现实世界的桥梁, GIS空间数据模型也是地理信息科学研究的核心与基础(Reitsma, 2013)。空间概念模型涉及空间对象的几何、关系、拓扑、语义和属性等信息,但其核心则是几何数据的有效表达(Zlatanova和Tempfli, 2000)。过去20年间,三维GIS数据模型与数据结构理论及技术研究取得了长足的进步,推出了多种GIS模型,但与实际需求仍相距甚远。与二维世界相比,不同形式人类活动作用下的三维世界及其现象复杂、多样,如不同文化背景下的建筑物可能差异巨大(Goodchild, 2009)。三维GIS数据模型不仅仅是传统二维GIS空间数据模型上增加垂直Z坐标,不能通过从低维向高维的简单拓展实现,而是需要相应认知与思维方式的改变。

但当前三维GIS数据模型还存在以下严重不足。

首先,现有三维GIS数据模型中,一类是以满足某个领域应用需求或考虑GIS某些特定功能为目的而建立的(Zlatanova等, 2004),因而所构造的模型为面向特定应用的(Zhou等, 2008),如3D geo-database (Mallet, 2002)主要应用于地质体建模, Coors(2003)的UDM模型主要描述城市地物,朱庆等(2008)的“基于语义的多细节层次三维房产模型”主要用于三维地籍房产管理,以及Zlatanova (2000)用于网络GIS环境下的简化空间模型(SSM)等。但面向特定应用范畴的模型难以应用于不同领域中。另一类为相对通用模型,如形式化数据结构(3D FDS)(Molenaar, 1991)、四面体模型(TEN)(Pilouk, 1996)、地理信息空间图解(ISO19107 Geographic Information-Spatial Schema)(Herring, 2001)、CityGML(Gröger等, 2012)等。为提高几何描述的普适性,该类数据模型一般采用不同空间维度的最简(或较简)图元(如三角面片、四面体等),尽管从理论上讲,任何空间对象与空间现象均可用最简图元描述,但该表示方式将导致三维空间数据大量冗余,一个复杂对象往往需由众多图元构成(即大数据量),何况GIS描述海量对象;同时,由于其图元小于人类认知的基本几何单元,即存储与分析单元不一致,空间查询与分析等计算将伴随基本图元

的聚合操作,严重影响GIS数据管理、查询、分析及可视化等功能。因此,迫切需要研究兼容不同数据模型、表达高效、普适性强、可满足多用途(户)应用需求的新型GIS集成模型(李清泉和李德仁, 1998; 闫国年等, 2013)。

其次,表达类型不足,现实世界中许多人造地物如圆形广场、拱形建筑、抛物形冷却塔等都可用函数有效描述,而当前模型均为基于点、线、面等几何实体的离散模型,缺乏函数表达(Breunig和Zlatanova, 2011)。离散建模方式一方面可导致数据量的大幅度增加,如CityGML(Gröger等, 2012)数据文件长度远高于拥有函数、参数化表达方式的建筑物模型(BIM)文件,为后者的几十倍(Berlo和Laat, 2011);另一方面,理论函数的离散化(如将球面离散化为三角形表示)可产生精度误差(Kuttig, 2003)。由于缺少函数, GIS与其他软件(BIM, CAD等)间的信息无缝转换也将无法实施。

第三,数据的重用性问题。一个小区中多栋建筑楼房几何形态经常完全一致,仅有的差异是楼层数目和楼的方位朝向;同样各楼层、单元之间也完全一致。这些结构相同,只是坐标位置、朝向不同的建筑,设计人员只会绘制一遍,然后在平面图上标注它们建筑位置和建筑姿态,这就是后面提到的映射概念,将映射概念引入GIS,即仅存储地物的相对位移与姿态(朝向),可极大提高数据的重用性(reuse),以避免重复建模与重复存储的成本,提高三维数据生产效率(需要指出的是:数据制作、更新成本高是目前制约3D GIS发展的主要瓶颈之一)。

总之,面对复杂、多样化的三维世界以及新型计算环境,现有三维GIS数据模型建模与分析效率差、数据制作效率低,难以满足高动态、多要素(海量对象)管理及多用途需求,其重要原因是现有模型中基本图形单元小于人类认知分析的基本几何单元,语义层次单一,难以表达现实世界知识特征;三维GIS数据模型的创新不仅需采用人类所掌握的几何图形知识,还应广泛借鉴(称作泛知识化)CAD和3DMax等成熟三维软件的经验,并不断从数学、计算机科学和工程等其他学科中吸取营养,比如工程设计中一般不会重复设计相同建筑, GIS建模则也应纳入数据重用概念,以避免重复建模劳动与计算机存储成本。为此,本文在已有三维数据模型(包括边界表达

(B-rep)、构造实体几何(CSG)等), 结合人类认知粒度特征, 对其进行了系统的拓展, 即离散结构中可使用参数, 形成了新型高效“离散结构”, 并提出了函数及映射两类新型表达方式, 而每类结构又可进一步分为子类, 如离散结构可分为离散单元、集合操作等多个子类; 同时在统一的建模框架下, 对离散、函数及映射三大类结构类型进行了集成, 进而构建了UKRM(ubiquitous knowledgeable representation model)数据模型. 该模型具有良好的可拓展与可维护特征, 用户可以根据应用需求方便地定义或维护新结构, 以充分集成人类对客观世界认知表达的不同知识, 从而灵活、高效地表达复杂多样的客观世界. 所要指出的是: 3D模型是连接2D及动态GIS的桥梁, 从几何上讲, 2D是3D的二维投影, 而动态GIS则是“3D+目标瞬间姿态”. 本文在解析其逻辑关系的基础上, 通过二维“环”的定义及“映射结构”的引入, 实现2D/3D一体化建模, 建立3D及动态GIS的接口和关联. 本研究以阐述泛知识化表达模型UKRM为主, 尽管UKRM可满足多用途(户)GIS应用需求, 但受限于篇幅, 本文只给出建模应用, 通过与CityGML 2.0和IFC 2×4建模效率对比, 讨论了UKRM模型的优势.

2 基本定义与概念

书写约定: 为表示清晰, 本文约定集合中的元素如下表示, 当集合中的元素为基本元素时, 用小写字母组成的单词表示(如 $trip$); 当集合中的元素为集合时, 用大写字母组成的单词表示(如, 空间对象六面体包围盒 BOX); 当集合中的元素为集合的集合时, 称为集族, 为了强调这种特点, 加注方括号表示(如, $[BOX]$ 表示集合 BOX 的集合, 即 BOX 集族).

(1) 空间2D及3D点:

$$P2D=\{x,y\}, \quad (1a)$$

$$P=\{x,y,z\}. \quad (1b)$$

(2) 运算符 op : 为集合操作符号并“ $\cup$ ”、交“ $\cap$ ”、差“ $/$ ”及括号“(”和“)”等.

(3) 子面(sub-surface)、子平面(facet), 子平面是指三维空间中闭合有界的平面多边形(含三角形), 由若干个平面子面的集合或一个函数描述的有界曲面称为子面.

(4) 子对象(sub-object): 若一个空间对象(object)

由若干个小的对象构成, 则这些小对象称作子对象. 在建模过程中, 子对象可根据宏观几何或者语义差异加以划分.

(5) 复合对象或子对象(composite object/sub-object): 本文定义空间对象或子对象具有4类基本类型(shapeType): 点(point)、线(line)、面(surface)和体(body或solid), 而由两种或者两种以上构成的对象称为复合对象.

(6) 辅助对象或子对象(auxiliary object/sub-object): 现实世界中不存在, 但为了GIS表达、分析方便, 人为生成的对象或子对象. 如图1中, 球壳对象(Obj)可利用大球(SubObj₂)及内部空心球(SubObj₁)两个辅助子对象, 通过差集(“ $/$ ”)计算方便的构建.

(7) 环[RING]: 面、体或者含有面、体成分的对象或子对象, 在2D空间投影形成的闭合边界(该边界线可为由函数表示的曲线); 对于离散点构成的环(即并非为函数表示的曲线), 环的定义为 $[RING]=\{np, [P2D]\}$, 其中 np 为环上二维点数量, $[P2D]$ 为相应点二维坐标集族.

(8) 映射(Mapping)、姿态(ATTITUDE)及位移(OFFSET或DISPLACEMENT): 假定运动目标三维空间局部坐标系相对全局坐标系3个坐标轴夹角(α, β, γ), 由此构成的3×3姿态矩阵($\{a_{ij}\}; i,j \in [1,3]$), 称为姿态(图2), 而运动目标局部坐标原点在全局坐标系的坐标称为位移. 利用已有对象或子对象数据(称为映射源对象), 经过位移、姿态变换, 可产生新的映射对象. 任何类型的对象(点、线、面、体和复合体)均可作为映射源对象.

图1 对象(Obj)通过辅助子对象(SubObj₂和SubObj₁)差集计算构建

图2 映射二维示意图

黑色矩形对象A为映射源对象, 其他对象为映射对象, 坐标系 (oxy) 为全局坐标系(已变换至对象A), 映射对象A'的局部坐标系为 $(O'x'y')$

3 泛知识化表达模型

本文提出的泛知识化表达模型UKRM包含三种结构类型: 离散结构、函数结构及映射结构(详见本文3.2节), 离散结构是最重要的结构类型, 而该类型的核心是离散单元结构(离散数据基本单元), 因此有必要先着重讨论.

3.1 离散单元 *DU* (discrete unit)

根据数据的组织形式, 离散单元结构分为离散曲面 *DCS* (discrete curved surface) 和离散体 *DB* (discrete body), 离散曲面称之为显性步履, 而离散体称之为半隐性步履. 这里用步履类型 (*tripType*) 代表特定 *DCS* 或 *DB* 的描述方式.

3.1.1 离散曲面 (*DCS*)

离散曲面 *DCS* 是以离散数据描述曲面(平面可认为是曲面的特殊形式), *DCS* 直接给出点、子平面(或双直纹面等)图形及其关系. 按不同步履方式, 可定义如下类型的 *DCS*.

(1) 三角扇 (*Triangle Fan*), 即为现行通用的三角扇, 在录入的数据链中, 每相邻两个点均与数据链中第1个点形成一个三角形(图3a), 点的数量 (np) 和三角形个数 (ns) 的关系是:

$$np = ns + 2. \quad (2)$$

三角扇的数据集合定义为

$$[DCS] = \{tripType, np, [P]\}. \quad (3)$$

这里 $[DCS]$ 为 *DCS* 的集族, $tripType = \text{"Triangle Fan"}$,

图3 不同步履类型的 *DCS*

(a) 三角扇, (b) 三角带, (c) 四边形带, (d) 双直纹面带步履, (e) 多点连续步履, (f) 单点独立步履

np 为节点的个数, $[P]$ 是节点的集族.

(2) 三角带 (*Triangle Strip*), 即为现行通用的三角带. 在数据链中相邻的3点形成一个三角形(图3b), 点的数量和三角形个数的关系仍为式(2)、数据集合仍为式(3).

(3) 四边带 (*Quadrilateral Strip*), 每组读取2个数据, 与上组读取的数据组成一个三维空间内的平面四边形(图3c). 数据集合仍为式(3), 点的数量 (np) 和四边形个数 (ns) 的关系为

$$np = 2 \cdot ns + 2. \quad (4)$$

同样方法, 我们可以设计双直纹带面带步履 (*Double Ruled Surface Strip*) (图3d), 用于地形、地质断层等空间对象的表示.

(4) 多点连续步履 (*MCT*) (*Multi-point consecutive trip*). *MCT* 一次性描述拥有相同边界数、且语义相同的空间子平面, 其数据集合为

$$[DCS] = \{tripType, ns, nsp, [P]\}, \quad (5)$$

式中, $tripType = \text{"MCT"}$, ns 为子平面数, nsp 为每个子平面节点数量, $[P]$ 为全部节点 (P_{ij} , $(1 \leq i \leq ns, 1 \leq j \leq$

nsp))的集族, 即 $P=\{\{P_{1,1}, P_{1,2}, \dots, P_{1,nsp}\}, \dots, \{P_{i,1}, P_{i,2}, \dots, P_{i,j}, \dots, P_{i,nsp}\}, \dots, \{P_{ns,1}, P_{ns,2}, \dots, P_{ns,nsp}\}\}$, 如图3e所示, 该MCT包含6个正方形空间子平面, 这里子平面之间可以相互独立与分离的, 如楼房门或者窗户等。

(5) 多点独立步履(MIT) (*Multi-point independent trip*). MIT一次性描述若干个语义相同, 但每个子平面边界数可能不等的空间子平面, 其数据集为

$$[DCS]=\{tripType, \{ns, NSP\}, np, [P]\}, \quad (6)$$

式中, $tripType = "MIT"$, ns 及 np 分别为子平面数目和节点总数, 集合 NSP 的元素按顺序说明各子平面的边界数目. 图3f所显示了一个不规则多边形MIT, 其中 $ns=1$, $NSP=\{8\}$, $np=8$, $[P]=\{P_1, P_2, \dots, P_8\}$.

DCS不仅可以表示面对象或子对象, 也可以构建以面闭合包围的体对象或子对象(即B-rep).

3.1.2 离散体(DB)

离散体DB是由子面包围的三维有界几何体. 离散体步履结构隐性描述了一个特定离散体的表面, 同时也描画了体的形状. DB的定义点等离散数据, 也可采用参数(参数化描述效率较高, 故尽量采用). 各种立体结构具有结构紧凑、数据量小的特点. 立体离散结构步履形式人们早已进行了应用, 如, 对四面体的描述, 只给出4个顶点, 暗含每三个点组成一个子平面. 在立体步履中, 分为直立体步履结构和斜立体步履结构. 直立结构步履的特点是: (1) 结构中都给出一个二维环的节点集族([P2D]), 与其他参数集族([OTHERS])组合形成体; (2) 体中所有空间节点在平面 Oxy 上的投影, 都在二维环之内; 设环的节点数为 np , 集合的一般形式:

$$[DB]=\{tripType, np, [P2D], [OTHERS]\}. \quad (7)$$

$tripType$ 为步履类型, 典型的直立结构包括直棱柱、屋脊、楔体、正棱台等, 限于篇幅, 本文示范性给出直棱柱、屋脊和楔体的DB结构定义.

(1) 直棱柱(right prism, [PRISM]), 直棱柱的数据集为

$$[PRISM]=\{tripType, np, \{[P2D], z_0, h\}\}, \quad (8)$$

式中, $tripType = "right prism"$, $[P2D]$ 为柱体的二维环, 即二维坐标表示底地面节点的集合, np 为相应节点数量, $[P2D]$ 也可以三维坐标点集合 $[P]$ 代替; h 为柱高, z_0 为柱地面高程, 隐式给出底面子平面、顶面子平面和

侧面子平面.

(2) 屋脊(house ridge, [RIDGE]), 屋脊是一种常见的房屋顶层建筑, 是特殊形式的多面体(图4a), 其集合表达形式为

$[RIDGE]=\{"house ridge", np, [P2D], z_B, h, offset\}$, (9) 式中, $[P2D]$ 为屋脊地面二维坐标表示的节点集合, np 为相应节点的数目, z_B 为屋脊地面高程, h 为屋脊高度, $offset$ 为屋脊线端点与底面边界线的偏移量(图4b).

(3) 楔体(wedge, [WEDGE]), 楔体(图5)的集合表达式仍为式(8), 其中 $tripType = "wedge"$.

3.2 结构类型

本文定义的结构类型(structureType)包括离散结构、函数结构、映射结构三大类, 而每一类还可包含诸多子类, 现分述如下.

3.2.1 离散结构(discrete structure)

由若干个小对象组成的对象, 或由若干对象通过集合操作生成的对象, 这种结构称为离散结构. 离散结构子类有离散单元结构、集合操作结构、离

图4 屋脊三维图(a)及其二维投影图(b)

图5 楔体三维图

散对象结构, 下面进行分述.

(1) 离散单元结构([DUS]). 离散部分均以离散单元DU(discrete unit)形式出现, 该结构称为离散单元结构, 其模型为

$$[DUS]=\{ndu, [DU]\}, \quad (10a)$$

$$[DUS]=\{\{np, [P]\}, \{ndu, [DU]\}\}, \quad (10b)$$

式(10a)与(10b)分别为离散单元的非拓扑和拓扑结构; [DU]可为[DCS](离散面的集族)或[DB](离散体的集族), 对于拓扑结构[DU]仅存储点索引; ndu为DCS或DB的个数; np是点的数量, [P]是拓扑结构点P的集族.

(2) 集合操作结构(set operation structure, [SOP]). 通过对子对象进行集合操作生成的对象或子对象, 该结构称为集合操作结构, 其模型为

$$[SOP]=\{nSubObjNum, nop, SUBNUM, OP\}, \quad (11)$$

式中, nSubObjNum为引用子对象数量, 包括辅助对象; nop为操作符数量; SUBNUM为引用的子对象编号的集合; OP是操作符集合, 为后序操作符.

(3) 离散对象结构(discrete object structure, [DOS]), 由多个子对象形成的对象, 该结构称为离散对象结构. 模型为

$$[DOS]=\{nso, [BOX], [SO]\}, \quad (12a)$$

$$[DOS]=\{nso, [SO]\}, \quad (12b)$$

式(12a)包含包围盒集族[BOX], 适于由大量节点子对象来构建对象, 而式(12b)适于少量节点的子对象. 式(12)中, nso和[SO]分别为子对象的数量和子对象集族. 对于集合操作, 组成对象的子对象包括集合操作生成的和参与集合操作的子对象. 对象仅由一个子对象组成, 但使用的子对象多于1个子对象, 也是离散对象结构. 如图1中Obj=SubObj₂/SubObj₁, SubObj₁, SubObj₂是辅助子对象, 只有Obj才是对象的子对象, 此时nso=3. 不失一般性, 式(12)中的nso是结构数据中子对象的数量. 缺省情况下, 子对象之间的操作是并集. 此外, 由子对象构建子对象时, 也可采用离散对象结构.

3.2.2 函数结构(function structure, [FS])

函数结构采用一个函数构建对象或子对象, 其模型为

$$[FS]=\{functionName, PARAMETER\}, \quad (13)$$

式中, functionName为函数名称, PARAMETER为参数集, 变量也包含在参数集中.

多种具有规则几何形态的空间对象(如: 球体、椎体和双曲面等)可由函数定义, 每个函数的定义需要给出相应的变量、参数及操作符. 这里的变量和参数不仅仅数学中常用的定义范畴, 如球体的球心与半径等, 离散单元(DCS或DB)也可作为参数定义新函数. 本文给出包含变量、参数及操作符全方位球缺函数(spherical segment), 以及采用DCS作为参数的门窗循环函数的定义, 加以说明.

(1) 球缺函数(spherical segment, [SSEGMENT])

参数PARAMETER={ {P₀, R, α, β, S}, {≥, ≤} },

[SSEGMENT]={ "spherical segment",

$$\{ \{P_0, R, \alpha, \beta, S\}, \{op_1, op_2\} \} \}. \quad (14)$$

设球缺(产状)面方程为 $F = x \cos \alpha \sin \beta - y \sin \alpha \cos \beta + z \cos \alpha$ (Zhang S Q和Zhang J Y, 2010), 其中倾角α, 即球缺平面的法向量与z坐标轴的夹角; β为走向角, 即球缺所在子平面的投影与x轴的夹角(图6, 其中设ABCD为球缺所在子平面); R为球的半径, S为球心到球缺平面的距离, P₀为球心, F为坐标原点到球缺平面的距离; 两个操作符分别是: 第一个符号表示“F op₁ S”, 第二个表示“r op₂ R”(这里 $r = \sqrt{x^2 + y^2 + z^2}$), op₁, op₂的值可为符号“>”、“<”、“≥”、“≤”、“=”、“≠”中的一个. 若球缺面法向量垂

图6 平面产状示意图

子平面ABCD与水平面平行, 倾角α为子平面法线与z轴夹角, 此时α=0; 子平面投影边界线与x轴夹角为β

直向上, 即球缺面与水平平行, 则有 $\alpha=0$, $F \geq S$, $r \leq R$. 以这种方式设计球缺函数的优点是可以表达任意大小、全方位的球缺. 限于篇幅, 此函数的完整定义过程本文未列出.

(2) 门窗循环函数(*window or door looping*, [WDL]).

$$[WDL]=\{\text{"window or door looping"}, SN, h_0, nf, h, [DCS]\}, \quad (15a)$$

$$[WDL]=\{\text{"window or door looping"}, SN, h_0, nf, h, \{MCT, ns, np, [P]\}\}, \quad (15b)$$

式中, SN 是 P 数据类型, 表示窗(或门)所在平面的法向量; h_0 为起始层窗(或门)的高程; nf 、 h 分别是楼层数和楼层高度; 式(15a)的 $[DCS]$ 为起始层窗(或门)的结构数据, 该结构数据可用多点连续步履(MCT)描述(式(15b)).

函数的参数可以是常用的变量、参数、操作符, 或者是离散单元(DCS 或 DB), 还可使用映射结构(见3.2.3节), 定义映射函数.

3.2.3 映射结构(mapping structure, [MPS])

以已有对象(或子对象)为基础, 通过赋予其新的姿态等操作, 形成新的映射对象(或子对象), 其模型为

$[MPS]=\{OFFSET, ATTITUDE, [VARIABLE]\}$, (16)
式中, $OFFSET$ 为坐标原点偏移量, 是 P 型结构; $ATTITUDE$ 是坐标姿态, 即为 $(\{a_{ij}\}; i, j \in [1, 3])$ 3×3 的姿态矩阵. 变量集族 $[VARIABLE]$ 若为索引 $INDEX$, 称之为对象索引映射; 若为函数 $[FS]$, 称之为函数映射. 此外, $[VARIABLE]$ 还可单独或与 $OFFSET$ 形成索引结构、位移结构等, 下面做具体定义.

(1) 索引结构(*index structure*). 不经坐标变换, 映射对象(或子对象)经直接引用已有对象(或子对象)形成:

$INDEX=\{dataOffset, dataCL, connectString\}$, (17)
式中, $dataOffset$ 为数据偏移量; $dataCL$ 索引内容长度; $connectString$ 文件路径全称. 索引内容是否包含环, 视索引内容、长度而定.

(2) 索引映射结构(*index mapping structure*, [IMS]). 已有对象(或子对象)经姿态与位移变换, 形成新的映射对象(或子对象), 其模型为

$$[IMS]=\{OFFSET, ATTITUDE, INDEX\}, \quad (18)$$

式中, $INDEX$ 由式(17)所示. 当然, 也可直接以该映

射结构存储, 也可经变换后存储实例化数据. 设其索引的点的集族为 $[P']$, 经过变换后映射对象(或子对象)的点的集族为 $[P]$, 则 $[P]$ 可通过以下公式计算:

$$[P]=OFFSET+ALTITUDE[P'], \quad (19a)$$

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} + \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \cdot \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix}, \quad (19b)$$

式中, $\{a_{ij}\}$ ($i, j \in [1, 3]$)为映射对象(或子对象)姿态矩阵, (x_0, y_0, z_0) 为其局部坐标系原点坐标, 即变量 $OFFSET$ (图2).

(3) 位移映射结构(*offset mapping structure*, [OMS]), 已有对象(或子对象)仅经坐标位移($OFFSET$)变换, 形成新的映射对象(或子对象), 模型为

$$[OMS]=\{OFFSET, INDEX\}. \quad (20)$$

(4) 函数映射结构(*function mapping structure*, [FMS]), 已有函数对象(或子对象)坐标经姿态与位移变换, 形成新的映射对象(或子对象), 模型为

$$[FMS]=\{OFFSET, ATTITUDE, [FS]\}, \quad (21)$$

式中, $[FS]$ 为函数结构定义的任一函数, 该结构相当于对函数对象或子对象的变换及引用.

3.3 对象与子对象 UKRM 模型

上面阐述了泛知识化表达模型UKRM的结构类型, 利用这些结构可构建对象(或子对象)模型. 其中, 空间对象([OBJ]的定义一般包括数据头([HEAD]) and 图形数据([D]), 而子对象([SO])则可直接用图形数据描述:

$$[HEAD]=\{objectNumber, shapeType, BOX\}, \quad (22a)$$

$$[D]=\{structureType, [SDATA], nr, [RING]\}, \quad (22b)$$

$$[OBJ]=\{[HEAD], [D]\}, \quad (22c)$$

$$[SO]=[D]=\{structureType, [SDATA], [SA], nr, [RING]\}. \quad (22d)$$

式(22a)中, $[HEAD]$ (data head)为数据头, 它包含了对象编号(*objectNumber*); *shapeType*为图形类型, 包括点、线、面、体和复合对象等枚举; BOX 为对象的数据包围盒. 式(22b)中, $[D]$ (data)为结构数据, 它包含结构类型(*structureType*), 如 DCS 、 DB 、*function structure*、*mapping structure*等, 也包括环的集族 $[RING]$ 以及相应环的数量(nr); $[SDATA]$ (structure data)为结构数据, 不包含环; $[SA]$ (*surface attributes*)为面属性集族(如: 面的形状、内外面等), 一般点、

线、体、复合对象(子)对象[SA]= $\varnothing$ 为空集,多用于面;
 nr 为环的数量,[RING]为环的集族,[RING]为可选项,
 若不设环,则该项为空.式(22c)中,[OBJ]为对象,而
 式(22d)中[SO]为子对象;数据头仅赋给对象,而非子
 对象;*structureType*放在结构数据[SDATA]之外,是考
 虑到映射结构的使用,否则,当映射结构数据引用
 时,找不到结构类型,无法读取数据.

图7为UKRM模型(式(22))的UML图。从式(22)及图7可以看出,任何类型的结构(离散、函数和映射)都可以用于对象或子对象的描述,即对象(或子对象)可以用任意结构类型表达;而一个对象可由任意个数的子对象构成,对象的建模可以由各种结构类型的组合,从而极大提高了空间对象表示的灵活性,当由多个子对象构成对象或子对象时,需要通过离散对象结构构造。当然,一个对象也可直接由任何一种类型的结构数据所描述,此时对象相当于仅由一个子对象构成,例如对于体对象(或子对象)的构建,可以采用DCS为基本的面闭合包围而成,也可以离散体DB离散结构构建,还可以函数或映射方式形成。集合操作仅应用于子对象上,而非对象上,同时规定

差集集合操作仅对函数描述的子对象有效(图1); 另外为避免“对象-结构”树深度过大, 本文规定, 只有体子对象或者含有体成分的复合子对象可进一步包含子对象.

4 UKRM 建模应用

4.1 建模步骤

数据制作过程可简述为以下三步:

(1) Step 1. 确定所建对象是否直接由结构直接构建: 若是(相当于仅由一个子对象构成), 则转步骤Step 2A; 否则(即由多个子对象构成), 首先确定子对象数量(n), 转步骤Step 2B.

(2) Step 2A. 确定对象的结构类型: $DU(DCS$ 或者 $DB)$ 、函数或者映射? 根据所选择的结构类型, 直接构建对象, 终止建模过程, 结束返回.

(3) Step 2B. 从1到 n , 循环确定子对象结构, 并制作相应子对象数据.

(4) Step 3. 利用离散子对象结构, 将子对象合成为对象结构数据, 并生成对象模型(式(22)), 结束返回.

图7 UKRM 模型 UML 图

4.2 应用实例

本文给出了三个例子以证明UKRM模型对于不同类型地物建模的适应性: (1) 地质体建模——利用DCS结构表示地质体, (2) 采用DB、函数结构及映射的简单建筑楼房, (3) 采用DCS、DB、函数结构及映射结构表示复杂建筑。

4.2.1 矿体建模(例 1)

设图8矿体模型直接由离散曲面(DCS)结构构建, 即由两个三角扇: $[DCS_1]$ 和 $[DCS_3]$, 及一个闭合三角带 $[DCS_2]$ 构成矿体子对象, 下面首先简述非拓扑DCS结构(式(10a))矿体子对象模型建模过程(不制作二维环):

(1) Step 1, 描述三个DCS非拓扑结构数据

$[DCS_1]=\{ \text{"triangle fan"}, 8, [P] \}=\{ \text{"triangle fan"}, 8, P_0, P_1, P_3, P_5, P_7, P_9, P_{11}, P_{13} \}$,

$[DCS_2]=\{ \text{"triangle strip"}, 14, [P] \}=\{ \text{"triangle strip"}, 14, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}, P_{11}, P_{12}, P_{13}, P_1, P_2 \}$,

$[DCS_3]=\{ \text{"triangle fan"}, 8, [P] \}=\{ \text{"triangle fan"}, 8, P_{13}, P_2, P_4, P_6, P_8, P_{10}, P_{12}, P_2 \}$.

(2) Step 2, 根据公式(22d)定义矿体子对象 $[SO]$ 的非拓扑结构模型($structureType = \text{"non-topological DCS"}$)为

$[SO]=\{ \text{"non-topological DCS"}, \{ 3, \{ [DCS_1], [DCS_2], [DCS_3] \} \}$

$=\{ \text{"non-topological DCS"}, \{ 3, \{ \{ \text{"triangle fan"}, 8, P_0, P_1, P_3, P_5, P_7, P_9, P_{11}, P_{13} \}, \{ \text{"triangle strip"}, 14, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}, P_{11}, P_{12}, P_{13}, P_1, P_2 \}, \{ \text{"triangle fan"}, 8, P_{13}, P_2, P_4, P_6, P_8, P_{10}, P_{12}, P_2 \} \} \}$

图8 利用离散曲面(DCS)表示矿体

$\{ \text{"triangle fan"}, 8, P_{13}, P_2, P_4, P_6, P_8, P_{10}, P_{12}, P_2 \} \}$.

此外, 我们还可根据公式(10b)及(22d)构建矿体子对象 $[SO]$ 的DCS拓扑结构为

$[SO]=\{ structureType, \{ \{ np, [P] \}, \{ ndu, [DCS] \} \}$, 其中, 结构类型 $structureType = \text{"topological DCS"}$, 节点数 $np = 14$, 节点集族 $[P]=\{ P_0, P_1, \dots, P_{13} \}$; DCS数量 $ndu = 3$, DCS集族 $[DCS] = \{ [DCS_1], [DCS_2], [DCS_3] \}$; 设节点 $P_0, P_1, \dots, P_{13}$ 的编号分别是0, 1, $\dots$, 13, 每个DCS由步履类型及相应节点编号(整型)集合表示, 如集族 $[DCS_1]=\{ \text{"triangle fan"}, 8, 0, 1, 3, 5, 7, 9, 11, 1 \}$. 因此, $[SO]$ 的DCS拓扑结构模型可表示为

$[SO] = \{ \text{"topological DCS"}, \{ 14, \{ P_0, P_1, \dots, P_{13} \}, \{ 3, \{ \{ \text{"triangle fan"}, 8, 0, 1, 3, 5, 7, 9, 11, 1 \}, \{ \text{"triangle strip"}, 14, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 1, 2 \}, \{ \text{"triangle fan"}, 8, 13, 2, 4, 6, 8, 10, 12, 2 \} \} \} \}$.

4.2.2 带有球缺大楼建模(例 2)

图9显示了一个大楼主体(MB)及两个球缺(SS1, SS2), 可将它们分别表示为子对象, 最后合成为子对象(当然也可合成为对象), 具体建模步骤如下。

(1) Step 1, 分别利用函数结构及映射结构, 定义球缺SS1及SS2, 并分别将其描述为子对象 SO_1 和 SO_2 .

根据球缺函数(式(14)), 定义球缺SS1为: $[SS1]=\{ \text{"spherical segment"}, \{ \{ P_{01}, 6.0, 0.0, 0.0, 2.0 \}, \{ \geq, \leq \} \}$, 其中, 两个球缺所在球的半径 R 均为6.0 m, 球心到球缺平面的距离 $S=2.0$ m, 球缺平面与水平面平行, 有 $\alpha=0.0$; 因球缺所在房屋为正东西向, 故 $\beta=0.0$; P_{01}, P_{02} 分别为两个球缺的球心坐标, 操作符 $op_1=\text{"}\geq\text{"}$ 表示 $F \geq S$; 操作符 $op_2=\text{"}\leq\text{"}$ 表示 $r \leq R$, (其中 $r = \sqrt{x^2 + y^2 + z^2}$).

图9 带有两个球缺的大楼

大楼主体(MB), 两个球缺(SS1和SS2)

根据式(22d), 定义子对象 SO_1 , 不设环, 即

$[SO_1] = \{ \text{"function structure"}, \{ \text{"spherical segment"}, \{ \{P_{01}, 6.0, 0.0, 0.0, 2.0\}, \{ \geq, \leq \} \} \} \}$.

利用函数映射结构(式(21)), 只保留位移, 不记录姿态), 定义球缺 $SS2$ 为 $[SS2] = \{P_{02}, INDEX\}$, 其中, P_{02} 为球缺 $SS2$ 所在球的球心, 即位位移量 $OFFSET$; $INDEX$ 表示子对象 SO_1 的数据指针.

根据公式(22d), 定义子对象 SO_2 , 不设环, 即

$[SO_2] = \{ \text{"function mapping structure"}, \{P_{02}, INDEX\} \}$.

(2) Step 2, 采用直棱柱结构描述大楼主体(MBD), 并将其定义为子对象 SO_3 .

根据公式(8), 描述大楼主体 $[MB]$: $[MB] = \{ \text{"right prism"}, 4, \{ \{P2D_1, P2D_2, P2D_3, P2D_4\}, z_0, h \} \}$; 其中, "right prism"为DB结构步履类型, 4为直棱柱底面节点数量, $P2D_1 \sim P2D_4$ 为底面节点 $P_1 \sim P_4$ 的二维坐标, z_0 为直棱柱底面高程, h 为直棱柱高度.

根据公式(22d), 定义子对象 SO_3 , 不设环, 即 $[SO_3] = \{ \text{"DB structure"}, \{ \text{"right prism"}, 4, \{ \{P2D_1, P2D_2, P2D_3, P2D_4\}, z_0, h \} \} \}$, 这里结构类型 $structureType = \text{"DB structure"}$; 因为体子对象, 故 $[SA]$ 为空; 不设环, 故 nr 及 $[RING]$ 也为空.

(3) Step 3, 将大楼三个子对象合成为子对象模型 SO , 当然也可以合成为对象.

根据式(12b), 构造子对象结构数据(即离散对象结构数据), 并根据式(22d), 建立大楼子对象模型 SO (不设环), 即 $[SO] = \{ \text{"discrete object structure"}, \{3, \{[SO_1], [SO_2], [SO_3]\} \} \}$, 其中, 子对象的结构类型($structureType$)为离散对象结构($discrete object structure$).

4.2.3 复杂建筑物建模(例3)

图10d及e分别显示了一个包含三个单元的8层大楼的正面(阳面)及背面(阴面)图, 图10c为大楼的二维环. 大楼模型的建立可利用一个单元(单元1, $[SO_1]$)子对象模型, 之后采用索引映射结构建立另外两个单元(单元2、单元3)子对象模型($[SO_2]$, $[SO_3]$), 同时将大楼东西两侧窗户表示为两个子对象($[SO_4]$, $[SO_5]$), 最后将这5个子对象合成为大楼对象模型, 具体建模过程如下.

(1) Step 1: 单元1模型构建. 图10a和b分别是单

元1阳面和阴面立体图, 图中 w 与 d 分别表示窗户与门, h 为六面体, v 为楔体, r 为屋脊, h_1 大楼单元主体, r_1 、 r_2 与 r_3 为单元楼顶的三个屋脊. 单元阳面(图10a)中, h_2 为六面体门房, 1层有4个大窗户和1个门, 其他2~8层各有6个窗户(w); 单元阴面(图10b)中, h_3 与 h_5 分别为六面体门房, h_6 与 h_7 分别为门房下垫六面体, h_4 为侧房, v_1 、 v_2 分别为门房前楔体形阶梯, v_3 为侧房雨搭, r_4 与 r_5 为位于门房上的屋脊, 阴面2层有4个采光窗(w), 3~8层各有4个窗户(w). 整个单元1描述为面、体复合子对象 $[SO_1]$, 包括由所有离散体($h_{1 \sim 7}$, $r_{1 \sim 5}$, $v_{1 \sim 3}$)构成的体子对象 $[SO_{11}]$, 阳面1楼门窗构成的面子对象 $[SO_{12}]$, 阳面2~8层窗户构成的面子对象 $[SO_{13}]$, 阴面1楼门(d)和2楼采光窗构成的面子对象 $[SO_{14}]$, 以及3~8层窗户构成的面子对象 $[SO_{15}]$.

Step 1-1: 构建单元1离散体子对象模型, 分别用直棱柱DB步履(式(8))表示 $h_{1 \sim 7}$ 为 $[H_{1 \sim 7}]$, 利用屋脊DB步履(式(9))表示 $r_{1 \sim 5}$ 为 $[R_{1 \sim 5}]$ 利用楔体DB步履表示 $v_{1 \sim 3}$ 为 $[V_{1 \sim 3}]$. 然后根据公式(22d), 将这15个离散体描述为子对象 $[SO_{11}]$, 即

$[SO_{11}] = \{ \text{"DB structure"}, \{15, \{[H_1], [H_2], [H_3], [H_4], [H_5], [H_6], [H_7], [R_1], [R_2], [R_3], [R_4], [R_5], [V_1], [V_2], [V_3]\} \} \}$.

Step 1-2: 描述阳面及阴面门窗.

表示阳面一楼门窗. 定义为面子对象($[SO_{12}]$). 根据多点连续步履(MCT)(式(5)), 将一楼的2个窗户表示为 $[DCS_{11}]$, 1个门表示为另一 $[DCS_{12}]$. $[DCS_{11}] = \{ \text{"MCT"}, 2, 4, [P] \}$, 其中为步履类型($tripType$), 2与4分别表示子平面个数和没个子平面边的数量, $[P]$ 是节点集族; 类似地, 我们有 $[DCS_{12}] = \{ \text{"MCT"}, 1, 4, [P] \}$. 再根据公式(22d), 将2个离散曲面DCS描述为子对象 $[SO_{12}]$, 即 $[SO_{12}] = \{ \text{"DCS structure"}, \{2, \{[DCS_{11}], [DCS_{12}]\} \} \}$, "DCS structure"为结构类型, 这里2个离散曲面DCS使用的是离散单元结构(式(10a)).

表示阳面2~8楼的窗户, 描述为面子对象 $[SO_{13}]$. 利用门窗循环函数([WDL])(式(15b))表示2~8楼窗户, 即 $[WDL] = \{ \text{"window or door Looping"}, (0.0, -1.0, 0.0), 126.0, 7, 3.6, \{ \text{"MCT"}, 6, 4, [P] \} \}$, 其中窗户所在平面指向正南, 故 $UN = (0.0, -1.0, 0.0)$; h_0 (起始楼层海拔)=126.0m, nf (楼层数)=7, h (每层楼高度)=4.0m, ns (每层楼窗户数)=6, np (每个窗户边数)=4, $[P]$ 是二

图 10 复杂建筑物模型

(a) 大楼单元立体图(正面); (b) 大楼单元立体图(背面); (c) 大楼二维环(RING); (d) 大楼正面; (e) 大楼背面. d, w 分别代表门、窗; h_1, h_2 分别为门房和大楼单元主体; r_1, r_2 与 r_3 为三个屋脊

楼窗户节点集族.

再根据公式(22d), 将门窗函数所描述的结构数据表示为子对象 $[SO_{13}]$, $[SO_{13}] = \{ \text{"function structure"}, [WDL] \}$, 其中 $\text{"function structure"}$ 为函数结构类型(structureType). 类似地, 单元阴面一楼的2个门和二层的4个彩钢窗分别描述为两个DCS, 并将其描述为面子对象 $[SO_{14}]$; 将阴面3~8层的窗户用门窗循环函数描述为子对象 $[SO_{15}]$.

Step 1-3: 构建单元1子对象模 $[SO_1]$: 根据离散体结构(式(12b)), 即不设数据包围盒, 描述该单元子对象的结构数据 $[D-SO_1]$, 并将其定义为子对象 $[SO_1]$, 即 $[D-SO_1] = \{ 5, \{ [SO_{11}], [SO_{12}], [SO_{13}], [SO_{14}], [SO_{15}] \} \}$, 这里 $[SO_{11} \sim 15]$ 分别为上述5个子对象; 根据式(22d), 描述为该子对象 $[SO_1] = \{ \text{"discrete object structure"}, [D-SO_1] \}$, 不设环.

(2) Step 2, 利用映射结构, 构建单元2及单元3的子对象模型 $[SO_2]$ 和 $[SO_3]$.

利用位移映射(式(20)), 分别构建单元2, 3模型

的结构数据 $[D-SO_2]$ 与 $[D-SO_3]$, $[D-SO_2] = \{ (x_2, y_2, z_2), INDEX \}$, $[D-SO_3] = \{ (x_3, y_3, z_3), INDEX \}$. (x_2, y_2, z_2) 与 (x_3, y_3, z_3) 为单元2及单元3的局部坐标原点坐标(图2), $INDEX$ 为单元1模型结构数据 $[D-SO_1]$ 索引指针.

根据式(22d), 将单元2, 3模型描述为子对象 $[SO_2]$ 和 $[SO_3]$. $[SO_2] = \{ \text{"offset mapping structure"}, \{ (x_2, y_2, z_2), INDEX \} \}$, $[SO_3] = \{ \text{"offset mapping structure"}, \{ (x_3, y_3, z_3), INDEX \} \}$, 其中 $\text{"offset mapping structure"}$ (位移映射)为结构类型.

(3) Step 3, 采用门窗循环函数(式(15b)), 分别将大楼东西两侧窗户描述为子对象 $[SO_4]$ 和 $[SO_5]$:

$[SO_4] = \{ \text{"window or door looping"}, SN_1, h_{01}, nf, h, \{ \text{"MCT"}, ns, np, [P_E] \} \}$,

$[SO_5] = \{ \text{"window or door looping"}, SN_2, h_{02}, nf, h, \{ \text{"MCT"}, ns, np, [P_W] \} \}$,

其中 SN_1, SN_2 分别为大楼东侧和西侧窗门所在平面的单位法向量, $[P_E], [P_W]$ 分别为东西两侧起始层窗户的节点坐标集族, h_{01}, h_{02} 分别为起始层窗户海拔高程, nf 为楼层数, h 为楼层高度.

(4) Step 4, 合成并构建大楼模型[OBJ].

根据离散体结构(式(12b)), 大楼对象的结构数据 [D-OBJ] 表示为: [D-OBJ]={“discrete object structure”, {5, {[SO₁], [SO₂], [SO₃], [SO₄], [SO₅]}}}}, 其中“discrete object structure”为结构类型.

根据式(22c), 构建大楼对象模型[OBJ], 对象水平上设环(图10c), [OBJ]={objectNumber, “composite”, BOX}, {[D-OBJ], 1, [RING]}, 其中objectNumber为对象编号, “composite”为图形类型(shapeType)(即复合对象型), BOX为对象正六面体数据包围盒; 大楼二维环[RING] = {np, [P2D]}, np为环的节点个数, [P2D]为环上二维节点集族.

4.3 数据长度比较

由于CityGML2.0和IFC 2×4均采用松散的文本格式, 为更合理比较UKRM与二者几何表达效率, 我们将与几何表达无关的属性信息剔除, 将文本格式按如下方式变更为紧凑结构: 字符常量变更为枚举类型, 占4个字节; 序号或名称引用变更为指针类型, 占4个字节; 数值分为整型和双精度类型, 前者占4个字节, 后者占8个字节. 此外, UKRM中数据包围盒以提高空间查询和分析效率为目的, 非三维建模所必须, 故在示例中予以省略. 表1列出了上述例1~3不同建模方式下的数据量大小. 由于CityGML 2.0与IFC 2×4分别以非拓扑和拓扑结构表示地质体模型(例1), 故UKRM分别给出了两种结构的数据量以便于比较. 表1显示, UKRM模型数据量均小于其他两种模型. 本文示例细节层次等效于LOD3, 故CityGML 2.0仅统计LOD3的几何表达数据长度.

5 讨论与结论

(1) 表达模型. 为满足高效、方便地表达复杂、多样的三维世界, 且适合不同GIS应用需求, 本文提出的泛知识化表达模型(UKRM)包括了三大类结构

类型: 离散结构、函数结构和映射结构, 每种结构类型又包含若干子类, 新的结构(即基本表达单元)通过增加模块代码及其编号ID, 可方便地扩充与添加, 即UKRM模型采用模块化设计思想, 具有后向兼容性、可扩展性及可维护性(Bass等, 2003). 将这些异质性结构归一化为结构类型(structureType), 实现了多种表达(multi-expression)有效兼容, 一个空间对象可由一种或几种组合表达, 用户可方便地根据自己的应用需求及爱好选择合适的构模方式. 如例1采用不同DCS的组合, 例2中离散体DB、函数与映射结构的组合, 例3则使用了DCS, DB和函数以及映射结构. 例子表明UKRM模型表达效率高于常用建模方法, 无论规则还是非规则几何体, 其数据量均小于CityGML及IFC, 比CityGML模型小几十倍, 甚至更多(表1). 所要指出的是, 不同表示方式的结合并不影响语义的使用, 语义属性的赋值可以在对象、子对象及结构(即基本表示单元)不同层次上进行. 基于UKRM模型, 可形成新一代开放式的知识化三维GIS数据结构标准, 用户对三维GIS构模问题无需一事一议, 仅需重用已有结构或维护新增结构即可.

(2) 离散结构. 本文所提出的三种结构类型各有优点, 都从不同方面充分反映了空间对象规则性几何特征. 例如, 作为离散结构的主要形式, 离散单元结构(包括离散面DCS和离散体DB结构)不仅包含了对象或子对象离散数据(节点等), 还包含了几何特征参数, 如多点连续步履中包括子平面数(ns), 每个子平面节点数量(nsp)等; 而屋脊、直棱柱步履等都采用参数定义, 当然基本图形单元也可不使用参数(即常用建模方式). 由于DCS单元中, 子平面共性特征进行了统一描述, 子平面表达效率大幅度提高, 例如三角扇中子平面数与节点数接近1:1, 而在传统GIS建模方式中, 子平面数与节点数至少为1:3, 而例1非拓扑模型中, 所存储节点数量小于矿体节点数量的两倍, 因此离散曲面结构DCS极大降低了传统边界表达(B-rep)的数据冗余. DCS可以是三角扇、三角带、双直纹面等, 它既包含拓扑结构又包含非拓扑结构, 可广泛应用于面及由规则、非规则面所包围体的描述(如地质体). 相对于DCS, 离散体DB则更适合表示建筑物(如楼房)等规则体. 迄今为止, 人类已经积累了大量几何知识, 并已广泛应用于工程、建筑等行业, 如屋脊、楔体、直棱柱等等, 这些知识都可用于DB

表1 不同建模方式下的数据量比较

数据模型	不同空间对象文件长度(字节)			
	地质体模型(例1)		简单建筑(例2)	复杂建筑物(例3)
UKRM模型	拓扑	非拓扑	192	4172
	488	752		
CityGML 2.0		2304	31720	51336
IFC 2×4	736		536	10168

结构的描述. 事实上, 四面体网(Pilouk, 1996)、广义直棱柱(GTP)(Wu, 2004)等也是特定形式的DB结构. 与传统GIS所定义的一般多面体(如Arens等, 2005)不同, DB结构所定义的多面体具有特定形态且被人类熟知, 因而体元内大量低维要素无需显式给出, 如直棱柱、屋脊中的线、面要素均无需存储, 节点有时只需存储二维数据(x, y), 部分节点甚至也可省略, 故可大幅度降低数据冗余. 基于DB结构, 我们可利用二维建筑图纸构建三维建筑模型. 离散体DB一旦定义, 人们都会理解其几何形态(即具有形态语义), 即DB所存储的基本几何单元与人类认知方式一致. 当然, DCS结构的基本描述单元也可以是三角形或平面多边形(见多点独立步履(式(6))), 而DB也可以四面体表示非规则体, 即离散单元结构可兼容当前GIS数据模型, 保证了空间数据表达的完整与适应性. 此外需要明确的是: DCS与DB不会发生冲突, 无递归引用或相互调用.

(3) 函数结构. 函数结构灵活、高效, 其参数可以是一般变量、参数、操作符、离散单元(DCS或DB)、索引数据的指针等(即函数映射). 函数定义的各类函数不同于数值计算, 它不是一个数, 而是一个空间对象或子对象. 函数结构的引入, 改变了三维空间构模中由低维逐级表达高维的传统方法, 可在低维数据缺失的情况下直接表达三维空间对象, 如函数表示的球体, 既没有节点, 也没有线和面; 同时函数表达具有尺度不变性, 提高了空间表达精度与效率; 函数的使用也将促进GIS与其他三维软件数据无缝转换, 有利于实现GIS对于不同数据源的集成与整合. 所要指出的是: 当面缺失点后, 将不能再靠点的时针顺序确定面的内外属性, 而是靠内在属性, 因此对象模型中增加了面的几何属性项(见公式(22)). 一般而言, 面绝大多数为外面, 若将碉室看作一个对象, 碉室也是体, 碉室表面也是外面.

(4) 映射结构. 基于DCS、DB及函数定义的基本表示单元(即积木块, building blocks), 空间对象模型构建方便简洁, 可通过积木搭建方式快速生成(例1~3). 而映射对这些积木块或者由此搭建的空间数据, 通过直接引用或者姿态、位移变换方式实现数据重用, 如例2对函数进行重调用, 例3则对已构建的整个建筑单元模型数据, 通过位移变换加以引用, 其数据无需重复制作与存储. 映射表达模型吸取了建筑

及工程界知识, 接近于工程制图和计算机软件语言, 极大减少数据存储量、简化数据制作过程、减少人为输入带来的错误、使资料得以重用. 映射结构可在不同类型对象(多点、线、面、体和复合对象)、不同尺度上使用, 如小区多个相同大楼的引用, 而一座大楼内部不同单元、楼层, 以及各个结构一致的房屋单元或者大楼外形态的阳台等均可利用映射构建, 映射的引入因而有望突破当前三维GIS多尺度(楼内外)、海量要素建模的技术瓶颈. 空间对象走向(orientation)等参数在二维空间分析中具有重要作用, 在三维空间中, 姿态的描述将更具普遍意义, 如建筑物不同尺度的表示离不开映射的使用. 由于存储及网络数据传输量小, 故映射也将适合网络环境下GIS应用. 需要说明的是: 在时空数据结构中, 运动物体的局部坐标本身就是一个活动标架(moving framework)(陈省身和陈维恒, 1983), 一切静止的对象可以看成运动对象的瞬时状态, 这样就可以把映射对象理解为运动对象在不同时间的瞬时状态的表现形式, 故映射结构的纳入不仅可有效表示静态数据, 还建立了静态GIS向动态GIS拓展的接口. 对于动态目标(如卫星)轨迹的计算也将不可避免的用到函数, 用以计算、预测空间对象的轨迹、时间等, 而动目标姿态的描述则需要映射结构, 从这个角度讲, 函数和映射结构的引入是GIS发展的必然结果, 是时代产物.

(5) 三维空间属性. UKRM模型属性结构为: 类属性, 存储于文件头, 属于整个类的专题属性; 对象属性, 存储于对象数据头, 缺省为文件属性, 其作用域为整个对象, 所有子对象、离散单元(DCS或DB)均拥有该属性, 无需逐一赋值; 同样, 子对象或离散单元(DCS和DB)属性, 仅作用于相应子对象或离散单元(DCS和DB)上; 此外, 我们还分别对每个DB的子平面和点做了编号, 用户可将属性(包括纹理贴图、顶点颜色信息、光照材质等)直接赋值其中的某个或者某些子平面或者点上, 从而实现从“类→对象→子对象→离散单元→子平面→节点”的多类多层次属性管理.

(6) UKRM模型数据存储及管理. UKRM模型支持文件与数据库两种存储方式: 当采用文件存储时, 其结构为: 文件头、{数据头, 图形数据}, 这里{数据头, 图形数据}(式(22))为对象数据的集合, 而“文件头”则存储图层信息; 当存储于数据库时, 结构数据作为二进制大字节存放列到一列中, 其后存储其他

属性(包括用户自定义属性). 对于复杂场景, UKRM模型采用分层分类存储与管理, 如建筑物图层、地形图层等.

(7) 数据合法性检验、唯一性及完备性. 合法性: 由传统的使用Euler公式检查(张树清等, 2004), 转变为使用与函数、映射、各种步履相关的公式进行检查. 例如对DCS显性步履, 可以用 $tripType$, np , $[P]$ 之间的关系, 进行合法性检验, 再如, 各种立体步履也可采用相应公式进行检验. 唯一性: 离散结构与函数结构的精度差异是在GIS允许范围之内, 当表示同一空间对象时, 无论采用离散结构还是函数结构, 其几何体的位置都是“唯一”的; 映射结构是由离散或函数结构经线性变换(姿态、位移)形成, 可归结为其中结构类型之一, 所表示的空间位置也是“唯一”的(见例3). 完备性: UKRM模型可有效描述三维实体的几何、语义及拓扑, 其中拓扑为隐性表达为对于离散结构, 经遍历两个几何体的点、线、面、体相交, 可计算其在何处、以何种方式相交(Zhang S Q和Zhang J Y, 2010); 函数结构, 若由点、线、面包围的函数几何体, 其拓扑关系计算同上, 而对于存在与非离散曲面几何相交情形, 采用数学分析可求解其相交关系; 映射结构, 拓扑计算可转化为上述方法之一. 拓扑关系不因几何表达类型(即结构)而改变.

(8) 二维环. 环的主要用途是表示对象的二维边界, 如: 矿体、地质体、自然地物边界和建(构)筑物边界等, 对于大地地形这方面的作用显得尤为重要. 通过环绘制地物二维地形投影的边界, 与三维数据存储在同一文件之中, 而不必额外增加存储空间. 环的定义, 为2D/3D GIS一体化表达奠定了理论依据. 与CityGML的三维环(即地形相交曲线(TIC))相比, 本文所定义的二维环更具普适性: 它可以描述悬浮于地上或者深埋于地下的任意空间对象的二维环, 而TIC可通过几何计算产生, 无需依靠人力制作预先存储.

(9) 空间查询与分析. 泛知识化(UKRM)数据模型提升了数据表达粒度, 且采用树状存储结构, 降低了数据聚集操作, 从而可提高GIS的数据管理、查询、分析能力, 如Zhang等(2010)证明了“面-子面-子平面”树状结构几何求交的高效性, 另外UKRM在对象(式(22a))及含有较多节点的子对象(式(12a))设数据包围盒, 也是为提高查询及分析效率. 所要指出的是由于

基本单元的知识化表示, 使基于形态语义的空间查询成为可能, 如“XX市拥有球缺的大楼(图9)有多少栋? 分别位于何处?”此类查询, 便于空间定位及分析. 此外, 由于具有面向对象模型特征, 对象间相对独立, 故适合高性能、云计算等现代计算环境的开发与应用. 限于篇幅, 有关GIS功能及多用途(户)方面的深入研究留待后续开展.

总之, 本研究根据人类认知特征集成与拓展了已有离散化GIS建模方式, 包括离散面(DCS)、离散体(DB)等, 离散结构可采用参数, 同时提出了函数、映射两种新的结构类型, 并在灵活、可扩展的建模理论框架下集成三类结构, 深化并整合了多种表示方式(包括边界表达B-rep、构建实体几何CSG、参数/函数、映射结构等), 提高了数据重用性, 实现了2D/3D一体化建模, 为GIS用户提供了多种建模选择. 空间对象可由任意一种或者多种结构的组合加以表示, 以有效利用各种建模方式的优势, 充分捕捉对象不同几何规则性特征, 降低了数据冗余, 提高建模效率; 空间数据即可存储数据文件中, 也可存储于面向对象数据库或者具有二进制大对象的关系数据库, 以充分发挥数据库数据管理功能, 从而可灵活、高效地表达与分析复杂三维、多样化的世界, 满足GIS多用途(户)应用需求, 为知识化GIS的发展奠定了理论基础.

致谢 作者衷心感谢三位匿名审稿专家宝贵而富有建设性的意见.

参考文献

- 陈省身, 陈维恒. 1983. 微分几何讲义. 北京: 北京大学出版社. 178
- 李清泉, 李德仁. 1998. 三维空间数据模型集成的框架研究. 测绘学报, 27: 325-330
- 阎国年, 袁林旺, 俞肇元. 2013. GIS技术发展与社会化的困境与挑战. 地球信息科学学报, 15: 483-490
- 肖乐斌, 钟耳顺, 刘纪远, 宋关福. 2001. 三维GIS的基本问题探讨. 中国图形图像学报, 9: 30-36
- 袁林旺, 俞肇元, 罗文, 周良辰, 阎国年. 2010. 基于共形几何代数的GIS三维空间数据模型. 中国科学: 地球科学, 40: 1740-1751
- 张树清, 张俊岩, 张柏. 2004. GIS拓扑关系中广义Euler公式的理论导出及应用. 中国科学: 地球科学, 34: 61-68
- 朱庆, 胡明远. 2008. 基于语义的多细节层次3维房产模型. 测绘

- 学报, 37: 514–520
- 朱庆. 2014. 三维GIS及其在智慧城市中的应用. 地球信息科学学报, 16: 151–157
- Arens C, Stoter J, Van Oosterom P. 2005. Modeling 3D Spatial Objects in a Geo-DBMS using a 3D Primitive. *Comput Geosci*, 31: 165–177
- Bass L, Clements P, Kazman R. 2003. *Software Architecture in Practice*. Boston: Addison-Wesley Longman Publishing. 49
- Berlo L V, Laat R D. 2011. Integration of BIM and GIS: The development of the CityGML GeoBIM extension. *Intern 3D GeoInfo Conf*, 5: 197–210
- Breunig M, Zlatanova S. 2011. 3D Geo-database research: Retrospective and future directions. *Comput Geosci*, 37: 791–803
- Coors V. 2003. 3D GIS in networking environments. *Comput Environ Urban*, 27: 345–357
- Gong J Y, Cheng P G, Wang Y D. 2004. Three-dimensional modeling and application in geological exploration engineering. *Comput Geosci*, 30: 391–404
- Goodchild M F. 2009. Geographic information systems and science: Today and tomorrow. *Ann GIS*, 15: 3–9
- Gröger G, Kolbe T H, Nagel C, Häfele K H. 2012. OGC City geography markup language (CityGML) encoding standard Version 2.0. <http://www.opengis.net/spec/citygml/2.0>
- Herring J R. 2001. The OpenGIS Abstract Specification, Topic 1: Feature Geometry (ISO 19107 Spatial Schema) Version 5. OGC Document Number 100-101
- Kuttig D. 2003. Potential and limits of functional modeling in the CAD process. *Res Eng Des*, 5: 40–48
- Mallet J L. 2002. *Geomodeling*. New York: Oxford University Press. 624
- Molenaar M. 1991. Formal data structures, object dynamics and consistency rules. In: Ebner H, Fritsch D, Heipke C, eds. *Digital Photogrammetric Systems*. Karlsruhe: Wichmann Verlag. 262–273
- Pilouk M. 1996. *Integrated Modeling for 3D GIS*. Doctoral Dissertation. Netherlands: Wageningen Agricultural University and ITC. 58
- Reitsma F. 2013. Revisiting the ‘Is GIScience a science?’ debate (or quite possibly scientific gerrymandering). *Int J Geogr Inf Sci*, 27: 211–221
- Shi W Z, Yang B S, Li Q Q. 2003. An object-oriented data model for complex objects in three-dimensional geographic information systems. *Int J Geogr Inf Sci*, 17: 411–430
- Wu L X. 2004. Topological relations embodied in a generalized tri-prism (GTP) model for a 3D geosciences modeling system. *Comput Geosci*, 30: 405–418
- Zhang S Q, Zhang J Y. 2010. Theoretical analytics of stereographic projection on 3D-objects’ intersection predicate. *Int J Geogr Inf Sci*, 24: 25–46
- Zhou L C, Lü G N, Sheng Y H, Xu H B, Wang H X. 2008. A 3D GIS’s spatial data model based on cell complex. In: *The International Archives of the Photogrammetry. Remote Sensing Spat Inform Sci*, 21: 905–908
- Zlatanova S, Tempfli K. 2000. Modeling for 3D GIS: Spatial Analysis and Visualization through the Web. In: *Proceedings of the XIX Congress of ISPRS*. 1257–1264
- Zlatanova S, Rahman A A, Shi W Z. 2004. Topological models and frameworks for 3D spatial objects. *Comput Geosci*, 30: 419–428