

# 空间布局的约束图方法\*

王英林 吴慧中

(南京理工大学计算机系 南京 210094)

**摘要** 空间布局的自动化是智能CAD领域的研究方向之一,旨在为设计师提供智能辅助工具,用于求解在建筑、厂房设备、大规模集成电路以及产品包装等等若干领域出现的布局问题,图论方法是空间布局研究的主要途径之一。以往的布局模型由于难以充分表达知识与约束,使得设计过程与结果难以控制。本文提出一种可应用于2D及3D布局的规范约束图及层次约束图模型,给出了约束图解的存在条件。该模型可以表示精细的布局知识与约束,在一定程度上克服了以往图模型不能充分表达布局知识与约束的不足。本文还给出了基于约束图的布局生成算法。

**关键词** 计算机设计自动化,人工智能,布局。

**中图法分类号** TP301.72, TP18

从60年代起,已有许多研究者对利用计算机辅助布局设计进行了广泛研究。<sup>[1~4]</sup>由于布局问题中的相关约束与知识难以用一种数学形式加以充分表达,布局问题求解常常呈现组合爆炸,至今这一研究尚未找到十分令人满意的解决办法。

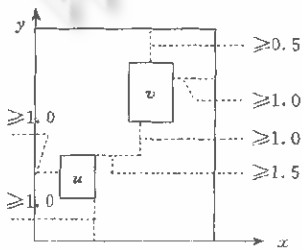


图1 物体u在v的下方

基于模型的方法是布局研究的主要方法,已先后提出了象域模型、墙模型、相邻拓扑图模型等,其中相邻拓扑图模型是用图论方法表示布局的典型方法。Mitchell与Roth J等人先后研究了该方法的布局表示与生成<sup>[5~7]</sup>,因为该方法仅考虑了“相邻”这一拓扑关系,而忽略了方向、距离等重要因素,故难以充分表达布局知识及约束。此外,由于该方法涉及图的平面化问题,难以将其扩充用于3D布局。

文献[8]提出了三维正交结构图模型,通过进一步研究发现,该模型仍不能充分表达布局知识及约束,如图1中“物体u在v的左下方”,用三维正交结构图表示时,只能从左右或上下关系中取其一。此外,以往的图模型表达布局时,

节点只与布置物整体相对应,这种表达是粗略的,不能表述由布置物的边之间的关系所反映的约束,为了更精细地表达空间关系,可允许图节点与布置物边界相对应。本文出于这一考虑,构造了规范约束图与层次约束图,给出了这两种图间的构造关系及表示布局的规则,给出了约束图解的存在条件。

本文考虑的布局对象为长方形布置物,规定其边分别与坐标轴平行。

## 1 规范约束图

由图1可以看出,任意两个长方形间的关系可分解为x方向与y方向的朝向与距离关系,并且在每一方向上都存在两个边界。可将各长方形及这些边界抽象为有向图中的节点,长方形及其边界间的朝向关系可以用连接这些节点的有向边加以表示。根据朝向关系的不同,可将这些有向边分为x向边与y向边。

### 定义1. 有源图

若一个有向图 $G=(V, E)$ 中存在两个节点 $s, z$ ,使得图 $G$ 中任意一个节点都在从 $s$ 指向 $z$ 的至少某一路径上,则称图 $G$ 为有源图,其中 $s$ 为源点, $z$ 为终点, $s$ 与 $z$ 统称为边界点,其它诸点称为内节点。

### 定义2. 正交有源图

\* 本文研究得到国家自然科学基金资助。作者王英林,1964年生,工程师,博士生,主要研究领域为计算机辅助设计,专家系统,空间布局。吴慧中,女,1942年生,教授,博士生导师,主要研究领域为智能CAD,面向对象知识库,计算机图形学,虚拟现实技术。

本文通讯联系人:王英林,南京 210094,南京理工大学计算机系

本文1997-01-03收到原稿,1997-04-28收到修改稿

若  $G=(V, E)$  可分解为两个子图  $G_x=(V_1, E_1), G_y=(V_2, E_2)$ , 且  $E_1 \cap E_2 = \emptyset, E_1 \cup E_2 = E, G_x, G_y$  均为有源图,  $G_x$  与  $G_y$  中的边具有不同方向, 则称图  $G$  为正交有源图.  $G_x, G_y$  中的边分别称为  $x$  向边与  $y$  向边,  $G_x$  与  $G_y$  分别称为图  $G$  的  $x$  向子图与  $y$  向子图.

**定义 3. 规范约束图**

若正交有源图  $G$  中两个子图  $G_x$  与  $G_y$  的内节点完全相同, 任意两个内节点间至少有  $x$  方向或  $y$  方向的某一条有向路径; 图  $G$  中的所有边权值非负 (正交有源图的有向边允许带有权值, 权值可为任意实数), 无环路存在, 则称该图  $G$  为规范约束图.

用规范约束图表示布局约束的规则为:

- (1) 用内节点表示各长方形布置物;
- (2) 用边界点分别表示布局区域的各边界, 一般用  $s_x, t_x$  表示左、右边界, 用  $s_y, t_y$  表示上、下边界;
- (3) 加权有向边表示两长方形物体在某一方向的朝向与距离约束, 规定  $x$  方向有向边  $v_i \xrightarrow{x, w_{ij}} v_j$  (或记为  $(v_i, v_j, w_{ij})_x$ ) 表示不等式约束:  $v_{j,l}(x) - v_{i,r}(x) \geq w_{ij} (w_{ij} \geq 0)$ . 其中  $v_{j,l}(x), v_{i,r}(x)$  分别表示长方形  $v_j$  的左边界与长方形  $v_i$  的右边界的  $x$  坐标. 这一约束的几何意义是:  $v_j$  在  $v_i$  的右边, 且两者间的最小距离 (指相邻边界间的距离) 为  $w_{ij}$ , 如图 2 所示. 同理, 对  $y$  方向有向边作类似约定. 图 1 布局的规范约束图表示如图 3 所示.

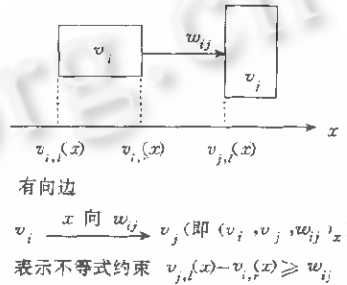


图2 不等式约束的有向边表示

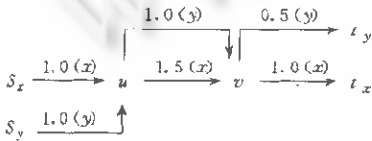


图3 图1布局的规范约束图表示

根据这一表达规则, 规范约束图定义中规定无环路的存在, 避免了  $v_i$  在  $v_j$  的右方, 同时  $v_j$  在  $v_i$  的右方的矛盾; 任意两内点间至少有某一方向有向边且权值非负的要求, 保证了在该方向其代表的长方形是分离的, 因而不会重叠.

由于规范约束图可分解为两个相对独立的子图, 在考虑其一致性时可分开考虑. 若两个子图均无矛盾时, 该规范约束图是无矛盾的 (一般仅对  $x$

向子图进行分析).

**定义 4. 规范约束图的子图  $G_x$  的解**

设  $G_x$  的内点  $v_i$  所代表的长方形  $x$  向的尺寸之长为  $v_{i,x} (i=1, 2, \dots, n)$ , 若有数值集合  $\{v_{i,l}(x); i=0, 1, \dots, n+1\}$  及  $\{v_{i,r}(x); i=0, 1, \dots, n+1\}$ , 满足  $v_{i,l}(x) + v_{i,x} - v_{i,r}(x) <$  (规定  $v_0, v_{n+1}$  分别为源点  $s_x$  与终点  $t_x$ , 故令  $v_{0,x} = v_{n+1,x} = 0$ ) 以及图  $G_x$  中所有有向边所表示的不等式约束, 则称数值集合  $\{v_{i,l}(x); i=0, 1, \dots, n+1\}$  为  $G_x$  的一个解. 显然  $G_x$  的解对应于节点所代表的长方形的左边界的  $x$  坐标. 如果  $G_x$  存在一个解, 其每一个分量  $v_{i,l}(x)$  均小于或等于  $G_x$  的其它解的相应分量, 则称这个解为  $G_x$  的最小解 (规定源点  $s_x$  的  $x$  坐标  $s_x(x) = 0$ ).

**定义 5. 规范约束图两节点间路径长**

若从  $v_i$  到  $v_j$  存在  $x$  向路径, 则定义该路径长为  $\sum_i d_{v_i}(x) + \sum_l w_l$ . 其中  $d_{v_i}(x)$  表示该路径上除端点  $v_i, v_j$  外其它点  $v_k$  所代表的长方形在  $x$  方向的尺寸长度,  $w_l$  则表示该路径上的边  $l$  的权值,  $\sum_l w_l$  即为该路径所有边的权值之和.

**结论 1.**  $G_x$  为某一规范约束图的  $x$  向子图, 设  $f_i(x)$  为从源点  $s_x (=v_0)$  到点  $v_i (i=0, 1, \dots, n+1, \text{注: } v_{n+1}=t_x)$  的最长路径的长, 记  $\overline{v_{i,l}}(x) = f_i(x) (i=0, 1, \dots, n+1)$ , 则  $\{\overline{v_{i,l}}(x), 0 \leq i \leq n+1\}$  是图  $G_x$  的最小解.

证明, 首先说明最长路径的存在性. 因为  $G_x$  无环路, 因此任意两节点间的路径数有限, 因而两点间最长路径必存在, 故对于规范约束图  $G$  的子图  $G_x, f_i(x)$  总有意义.

其次说明  $\{\overline{v_{i,l}}(x)\}$  是  $G_x$  的一个解, 即它应满足各不等式约束, 即对于每一权值为  $w_{ij}$  的有向边  $(v_i, v_j, w_{ij})_x$ , 有  $\overline{v_{j,l}}(x) - \overline{v_{i,r}}(x) \geq w_{ij}$ . 注意到应有  $\overline{v_{i,l}}(x) + v_{i,x} = \overline{v_{i,r}}(x)$ , 上式可转化为  $\overline{v_{i,l}}(x) + (v_{i,x} - w_{ij}) \leq \overline{v_{j,l}}(x)$ .

假定 (反证)  $G_x$  中有一条边  $(v_i, v_j, w_{ij})_x$ , 使得  $\overline{v_{i,l}}(x)$  与  $\overline{v_{j,l}}(x)$  不满足约束, 也即  $\overline{v_{i,l}}(x) + (v_{i,x} + w_{ij}) > \overline{v_{j,l}}(x)$ .

上述不等式表明, 从  $s_x (=v_0)$  到  $v_j$  的最长路径不经过边  $(v_i, v_j, w_{ij})_x$ . 因为不然的话,  $\overline{v_{i,l}}(x) + (v_{i,x} + w_{ij})$  应等于  $\overline{v_{j,l}}(x)$ , 这样, 从  $s_x$  到  $v_j$  的包含从  $s_x$  到  $v_i$  最长路径与边  $(v_i, v_j)$  的路径之长大于从  $s_x$  到  $v_j$  的最长路径长, 这与  $\overline{v_{j,l}}(x)$  的定义相矛盾, 因此  $\{\overline{v_{i,l}}(x)\}$  满足  $G_x$  规定的所有不等式约束, 是  $G_x$  的一个解.

下面说明  $\{\overline{v_{i,l}}(x)\}$  是  $G_x$  的最小解. 设  $\{v'_{i,l}(x), 0 \leq i \leq n+1\}$  是图  $G_x$  的另一个解, 对于一个给定的  $v_i$ , 假定从  $s_x$

( $=v'_{0,i}(x)$ )到  $v_i$  的最长路径中的边依次为  $(s_x, v_{i_1}), (v_{i_1}, v_{i_2}), \dots, (v_{i_{k-1}}, v_i)$ , 其中  $v_i = v_i$ . 由于解  $\{v'_{i,i}(x)\}$  满足由这些边所表达的不等式约束, 故

$$\begin{aligned} v'_{i,i}(x) + (v_{0,x} + w_{0,i_1}) &\leq v'_{i_1,i}(x) \\ v'_{i_1,i}(x) + (v_{1,x} + w_{1,i_2}) &\leq v'_{i_2,i}(x) \\ &\dots \dots \\ v'_{i_{k-1},i}(x) + (v_{k-1,x} + w_{k-1,i_k}) &\leq v'_{i_k,i}(x) \end{aligned}$$

将以上各式相加, 得

$$v'_{0,i}(x) + \sum_{j=1}^k (v_{j-1,x} + w_{j-1,i_j}) \leq v'_{i_k,i}(x)$$

注意到规定边界点  $v_0 (=s_x)$  的  $x$  坐标为 0, 故  $v'_{0,i}(x) = 0$ , 根据  $\overline{v_{i,i}(x)}$  的定义  $\overline{v_{i,i}(x)} = \sum_{j=1}^k (v_{j-1,x} + w_{j-1,i_j})$ , 故  $\overline{v_{i,i}(x)} \leq v'_{i,i}(x) (=v'_{i_k,i}(x))$ , 因此  $\{\overline{v_{i,i}(x)}\}$  是  $G_i$  的最小解. 若将以上规范约束图表达布局的规则以及路径长等概念扩展, 用于正交有源图, 则容易得到以下结论.

**结论 2.** 在 0 环路不计的情况下, 边权值非负的正交有源图有解的充要条件是各子图 ( $x$  向与  $y$  向) 均无环路 (所谓 0 环路是指路径长为 0 的环路, 证明略).

由于正交有源图的解即为各子图的解的并, 在不规定图中任意两内点均有非负的某方向有向边的情况下, 其解可能对对应物体间重叠的布局, 因此, 保证带非负权值边的正交有源图能够表达有效布局的充要条件是该图为规范约束图 (在不计 0 环路的情况下).

## 2 层次约束图

以上分析表明, 在节点表示长方形整体时, 规范约束图可以表达布局中物体间的方位与最小距离要求, 但需要表达更细致的约束时, 就需要考虑两物体边之间的关系, 因此, 我们允许用图节点表示长方形的边界. 如图 4(a) 中, 长方形  $u$  与  $v$  在  $y$  方向上相邻且具有长度最小为 1, 最大为 1.5 的公共边, 可用图 4(b)、(c) 的图表示, 其中  $u(1), u(2), u(3), u(4)$  顺序表示长方形的 4 个边界.

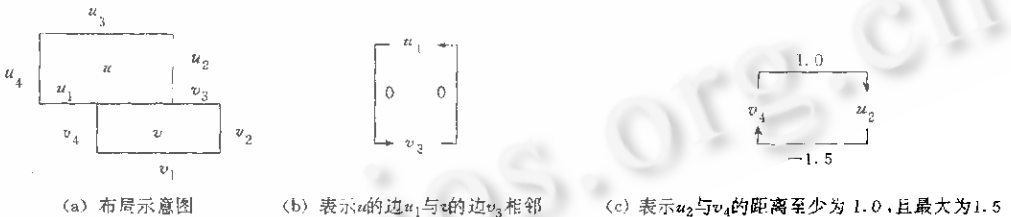


图4

在布局中, 可以先将布置物作为一个整体, 考虑其布局约束, 得到其规范约束图, 然后在此基础上引入新模型以表达更细致的约束, 如边间的最大、最小距离约束.

下面给出这种层次约束图的构造性定义.

### 定义 6. 层次约束图

设  $G$  为规范约束图, 将其中每一内节点  $v$  扩展为两对节点  $v(1), v(3)$  以及  $v(2), v(4)$ , 在它们之间连接有向边 (非负权值):  $x$  向边  $(v(2), v(4), w_{v(2),v(4)})$ , 与  $y$  向边  $(v(1), v(3), w_{v(1),v(3)})$ , 新增  $x$  向边、 $y$  向边分别与原规范图中的  $x$  向边、 $y$  向边顺次首尾相接, 这样得到的新图  $H$  称为层次约束图, 根据边的方向可将  $H$  分解为两个子图,  $x$  向子图  $H_x$  及  $y$  向子图  $H_y$ . 由规范约束图中同一节点扩展而成的这两对新节点, 称为一个相关节点组. 允许在每一子图, 比如  $H_x$  的节点之间再增加任意边权值的  $x$  向有向边, 这样增加有向边后的图仍称为层次约束图.

由以上构造性定义, 层次约束图不仅能够反映原规范约束图所具有的不同节点组间的约束关系, 而且能够更细致地表达同一节点组中的节点间关系以及不同节点组中的节点间关系. 对于图 4 所示布局例子, 其规范约束图与层次约束图表示如图 5(a)~(c) 所示 (为简单计, 其中省略了许多权值).

层次约束图表达布局的规则为:

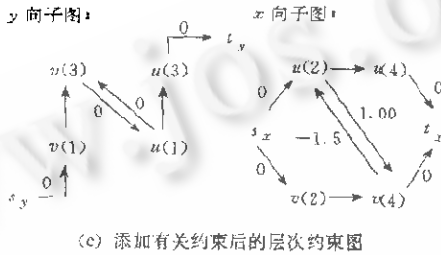
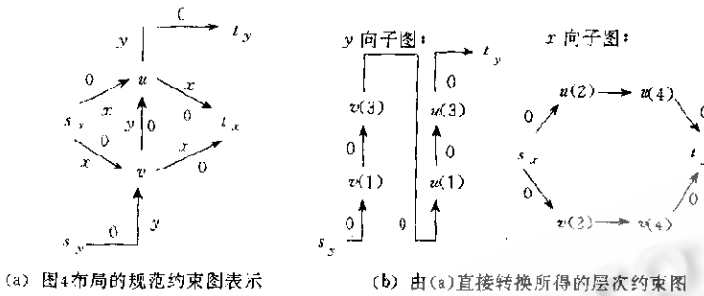


图5 从规范约束图到层次约束图的过渡

图6 矛盾约束

- (1) 每一节点组与布局中的长方形布置物相应;
- (2) 每一节点组  $v$  中的节点  $v(2), v(4)$  分别表示长方形  $v$  的右边界与左边界,  $v(1), v(3)$  分别表示  $v$  的下边界与上边界;

(3) 除了允许边权值为任意实数外,边所表达的不等式约束与规范约束图类似,即若有边  $(u, v, w_{u,v})$ ,则表示  $v(x) - u(x) \geq w_{u,v}$ . 其几何意义为:当  $w_{u,v} \geq 0$  时,表示  $v$  在  $u$  的右方,与  $u$  的距离不超过  $w_{u,v}$ ;当  $w_{u,v} < 0$  时,表示  $v$  在  $u$  的右方,或者虽然  $v$  在  $u$  的左方,但与  $u$  的距离不超过  $|w_{u,v}|$ .

在层次约束图中,由于允许增加任意加权有向边,因此其表达的约束可能是矛盾的.例如,在图6中同时有  $v(x) - u(x) \geq 3, u(x) - v(x) \geq -2$  是矛盾的.

定义7. 层次约束图的解

层次约束图  $H$ , 其子图  $H_x, H_y$ , 规定  $H_x$  的源点  $s_x$  的  $x$  坐标  $s_x(x) = r_0(x) = 0$ . 如果其各点的  $x$  坐标集合  $\{r_i(x); i = 0, 1, \dots, 2n, 2n+1\}$  分别满足  $H_x$  中所规定的的不等式约束, 则称  $\{r_i(x)\}$  为  $H_x$  的一个解.  $H$  的解由  $H_x$  与  $H_y$  的解合并而成. 其最小解的含义与规范约束图类似.

结论3. 若层次约束图的  $x$  向子图  $H_x$  中无正数环(正数环指边权值之和大于0的环路), 则该图有解, 且从  $s_x$  到每一节点的最长路径长的集合  $\{r_i(x); i = 0, 1, \dots, 2n, 2n+1\}$  即为其最小解. (层次约束图的路径长定义为该路径上所有边的权值之和).

证明: 首先应说明最长路径的存在性. 在无环路时, 从某一点  $r_i$  到另一点  $r_j$  的路径数有限, 在有限数目的路径中必有最长路径; 若从  $r_i$  到  $r_j$  的路径中有环路存在, 则每绕该环路多转一圈就得到从  $r_i$  到  $r_j$  的一条新路径, 故路径数是无限的. 但因无正数环, 即该环路权值之和  $\leq 0$ , 故多转一圈不会使路径变长, 故从  $r_i$  到  $r_j$  的最长路径(可以)不经过环路. 而这样的路径数有限, 故存在最大者(证明的其它部分类似于结论1的论证, 此处省略).

结论4. 层次约束图若有解, 其解必对应一个无重叠的长方形物体布局.

证明: 在前述规范约束图与布局的对应规则下, 规范约束图的解与某一无重叠的长方形物体布局相对应. 由于层次约束图是由规范约束图演化而来, 其有向边约束隐含了后者的约束, 因此, 层次约束图的解必满足相应规范约束图的约束, 因此, 层次约束图的解也必对应无重叠的长方形物体布局.

3 基于约束图的布局生成

布局设计可视为一个逐步求精的过程. 首先在较抽象的模型空间(规范约束图)求解, 逐步过渡到较具体的模型空

间(层次约束图),最终转化为具体坐标图,这种由粗到细的过程与人们求解复杂问题的策略是一致的.层次约束图的提出,为逐步细化提供了可能.以下是这种基于约束图的逐步求精布局生成算法.

(1) 若某两个布置物尚无(方位、距离)关系,则根据设计要求确定其关系;若设计要求对其关系无明确规定,则根据经验或随机选择某一关系.重复这一过程(若出现违背约束情况,回溯.若已搜索过所有可能情况且均违背约束或已超出某一时间阈值,失败退出),直至所有布置物均有方位及距离关系为止,这样便得到规范约束图.

(2) 将规范约束图转化为层次约束图,并将此阶段可以表达的设计要求及约束(而在规范约束图阶段无法表达)在层次约束图中表达出来.

(3) 求解层次约束图,若有解且满足设计要求及约束,成功退出;否则,回溯.

#### 4 实验与比较

我们采用约束图方法对房间布局进行了初步实验,并与其它方法进行了比较,结果表明:

(1) 约束图模型可以求解以往布局模型所难以表达的许多布局问题.由于以往的布局模型(如相邻拓扑图、墙模型、正文结构图等)不能充分表达布局知识与约束(布局知识及约束的表示是布局设计的关键问题),因此难以对布局问题进行描述与求解,使布局设计结果难以控制.而用层次约束图可以很方便地表达象“长方形 $u$ 与 $v$ 有公共墙”,“ $u$ 与 $v$ 相邻且相邻边的公共部分长度为1.5米”等以往模型所难以表示的约束.

(2) 通过约束图模型,使设计者能够通过添加具体的设计约束对设计过程及结果加以控制,这一模型可为设计师提供交互自动化布局设计工具.设计师可通过该模型表达其约束,然后由机器自动生成布局结果,通过修改约束,对结果进行控制,直到满意为止.

(3) 人在求解复杂问题时,一般采取由粗到细、逐步求精的策略,以降低问题复杂度.约束图为在布局设计中模拟人类这一求解策略提供了便利.以往的布局模型或过于抽象(如相邻拓扑图模型),或过于具体(如象域模型),缺少中间层次,因而在从问题空间到结果空间的转换过程中,存在求解的不平滑(跳跃)现象.在包括约束图作为子模型之一的多模型布局求解中,约束图可起到连接高层抽象子模型与布局坐标图、缓解跳跃现象、充分利用布局知识的作用.

(4) 在求解可由其它模型解决的简单约束布局问题时,在有适量布局知识情况下,约束图方法呈现出较高的效率.在求解同一问题时,其求解复杂度低于正文结构图,比基于相邻图的三角图平面化方法平均时间复杂度降低了18%左右.

(5) 约束图可以表达任意长方形组合而成的形体,因此可用于复杂形体的布局问题.

本文对约束图模型的讨论同样适合于三维情形,因此,约束图可用于处理三维物体布局问题.

#### 参考文献

- 1 Levin P H. Use of graphs to decide the optimum layout of buildings. *Architecture's Journal*, 1964, 140: 809~815
- 2 Auerbach R, Lin B, Elsayed E. Layout aid for the design of VLSI circuits. *Computer Aided Design*, 1981, 13(5): 271--276
- 3 Liggett R S, Mitchell W J. Interactive graphic floor plan layout method. *Computer Aided Design*, 1981, 13(5): 289~297
- 4 Aksoy M, Saglam G. A shape-grammar for courtyard houses. In: Beheshti M R, Zreik K eds. *Advanced Technologies*. Amsterdam, Netherlands: Elsevier Science Publishers B. V., 1993. 251~258
- 5 Mitchell W J, Steedman J P, Liggett R S. Synthesis and optimization of small rectangular floor plans. *Environment and Planning B*, 1976, 3: 37~70
- 6 Roth J, Hashimshony R, Wachman A. Turning a graph into a rectangular floor plan. *Building and Environment*, 1982, 17: 163~173
- 7 Roth J, Hashimshony R. Comparison of existing three-room apartments plans with computer-generated layouts. *Environment and Planning B*, 1987, 14: 149~161
- 8 吴慧中, 王英林. 一种立体空间布局模型及其布局算法. *计算机学报*, 1994, 17(11): 835~841  
(Wu Hui-zhong, Wang Ying-lin. A solid spatial layout model and the layout algorithm. *Chinese Journal of Computers*, 1994, 17(11): 835~841)

### The Method of Constraint Graphs Used in Spatial Layout

WANG Ying-lin WU Hui-zhong

(Department of Computer Science Nanjing University of Science and Technology Nanjing 210094)

**Abstract** Spatial layout automation is one of the research directions in intelligent CAD domain. The aim is to provide tools for designers to solve the layout problems emerged in building planning, facility planning, VLSI floor planning, product packaging and such like the fields. Human designers can interact with the tools and find some new creative solutions. The graph theoretic method is one of the main approaches in the research of spatial layout. One of the difficulties in layout design is the representation of layout knowledge and constraints. Former layout models can not fully represent layout knowledge and constraints. The drawbacks of former models lead the design process uncontrolled. This paper represents a normal-constraint-graph model and a hierarchical constraint-graph model that can be used in 2D or 3D layout design. The two models combined can be used to represent the detailed knowledge and constraints in layout design. Thus, it can overcome the drawbacks of the former layout models. In this paper the condition of the existence of the solution of normal and hierarchical constraint graphs is discussed and the layout-plan-generation method based on the normal and hierarchical constraint graphs is given.

**Key words** Computer aided design, artificial intelligence, layout.

**Class numbers** TP391.72, TP18