一种基于图分解的几何约束求解方法

何 伟 唐 敏 董金祥 何志均
(浙江大学计算机系,浙江大学人工智能研究所,杭州 310027)

摘 要 为了提高几何约束求解的效率和鲁棒性,对基于图的构造方法进行了改进,即加入虚约束进行扩展和过约束问题的一致性判定,提出了一种基于图分解的方法,用此方法可以处理包括完全约束、过约束和欠约束等多种情况的约束求解问题,另外,在该方法中还通过引入分解树将约束求解的范围由整体下降到局部,使大部分求解过程能够采用几何求解实现,提高了求解和后续修改的效率,通过实验数据测试证明,该方法对于大型约束求解问题可以达到实时处理的效果,具有较强的实用性.

关键词 计算机图形学(520·6030) 变量化设计 几何约束 约束求解
中图法分类号:TP391 **文献标识码**:A **文章编号**:1006-8961(2003)08-0926-06

A Constraint Solving Approach Based on Graph Decomposition

HE Wei, TANG Min, DONG Jin-xiang, HE Zhi-jun
(Dept. of Computer Science & Technology, Zhejiang University, Artificial Intelligence Institute, Zhejiang University, Hangzhou 310027)

Abstract To improve the efficiency and robustness of constraint solving algorithm, the paper presents a graph-constructive approach. In constructive phase, the algorithm can classify well-constrained, over constrained and under constrained configurations. By adaptively adding virtual constraints, the algorithm can handle all the above configurations, and can overcome the ill-condition constraint which occurs frequently in practice. Decomposition trees are used to reduce the scale of constraint solving problem, decompose it from global scope into many local sub-problems. By avoid solving large-scale equations, the algorithm can separately calculate the sub-problems geometrically with great improvement in efficiency and robustness. It also improves the efficiency for successive modification on constraints or geometric elements. Comparing to conventional algebra algorithms, symbolic algorithms, and rule-based algorithms, the algorithm is more practical and can be used in real-time constraint solving environment. It has been implemented in VC++ on Windows/NT platform, and been used as geometric constraint solving kernel of a feature based parametric modeling system GS-CAD, which is a commercial CAD system developed by Zhejiang University.

Keywords Computer graphics, Variational design, Geometric constraint, Constraint solving

0 引 言

变量化设计在CAD/CAM系统中占有重要的地位.它允许用户对产品设计的几何、拓扑信息定义约束,并能通过改变约束参数值方便地对现有设计进行修改.其中建立一个高效、鲁棒的几何约束求解器在变量化设计中占有中心地位.

对几何约束求解问题,国内外许多学者提出了各种方法.这些方法大致可以分为数值方法[1~3]、符号方法[4,5]、基于图的构造方法[6~10]3类.

数值方法是首先将约束转化为一组代数方程,继而采用Newton-Raphson等数值迭代方法[1]进行求解.这种方法通用性好,可以处理复杂的约束关系,但却不能良好地处理过约束和欠约束问题,并且由于采用数值迭代方法,对初值的要求比较高.

基金项目:浙江省自然科学基金项目(6001107);国家教育部博士点基金项目(2000033554)
收稿日期:2002-10-10;改回日期:2003-03-30

符号方法也是先将约束转化为一组代数方程，然后采用符号推理的方法，如 Wu-Ritt 分解算法或 Grobner 基方法将方程转化为易解的形式，最后采用数值方法求解. 这种方法可以求解一般的非线形方程，但是由于它一般具有指数阶复杂度，不适宜于大型约束系统.

基于图的构造方法一般可分为分析、构造两个阶段. 在第 1 阶段，建立约束图并对此图进行分析，同时建立一系列构造步骤；在第 2 阶段，则执行这些构造步骤，计算出几何元素的最终位置. Owen 通过分析图的三连通分量，可以解决基于直尺、圆规的设计问题[6]. Bouma 则利用分析刚体簇的方法，可以处理较复杂的设计问题[7]. Kramer 通过分析几何元素的自由度，利用动力学的思想对三维的几何约束满足问题进行求解[8].

本文提出了一种基于图的构造方法进行约束求解，这种方法的主要有以下特点：

(1) 能够高效地处理各种约束情况，包括约束过载问题和约束不足问题，并可以以交互速度处理较大型的约束系统.

(2) 引入了分解树的概念，将整个系统的求解转化为对各个子部分和它们之间关系的求解，提高了初始求解和后续修改的效率.

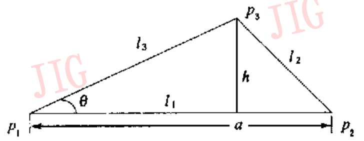
(3) 在构造阶段能够对约束过载问题的一致性进行判别.

1　基于图分解的几何约束求解

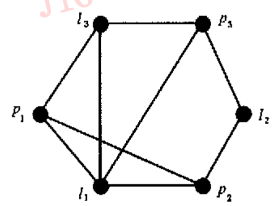
几何约束问题可以通过一组几何元素以及定义在这组元素上的关系来定义. 几何元素类型包括点、直线、线段、圆、圆弧，约束类型则包括重合、距离、角度、平行、同心及相切等等. 根据文献[11]，约束问题可以简化成只包含点和直线两种几何元素类型，约束类型也限制为点点距离、点线距离和线线角度 3 种，因此几何约束问题的形式化表示如下：

给定一个包含 n 个点和线的集合 $\boldsymbol{V}$ 以及它们之间的两两约束所组成的集合 $\boldsymbol{E}$，找到一个满足约束 $\boldsymbol{E}$ 的直观答案. 其中两两约束有以下 3 种类型：点线距离、线线角度以及非 0 点点约束.

通过一个约束图 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E})$ 可以表示这个简化的约束问题，其中图的节点 $\boldsymbol{V}$ 表示几何元素，图的边 $\boldsymbol{E}$ 则表示几何元素两两之间的约束关系. 图 1(a) 是一个简单的约束问题，而图 1(b) 则是它对应的约束图.



(a) 约束问题



(b) 约束图

图 1　一个约束问题及其约束图

约束求解方法包括分析、构造两个阶段. 在分析阶段，首先根据给定的几何元素和约束集建立约束图，然后将约束图分解为三连通分量，并在这个过程的同时建立一棵分解树；而在构造阶段，系统则根据分解树确定实际的构造步骤. 如果分解树所对应的图是欠约束图，那么系统要确定求解欠约束问题时所需增加的约束；如果分解树所对应的图是过约束图，那么系统还需要确定此过约束图是否是一致性的. 系统根据产生的构造步骤，采用代数方法或者数值迭代方法求出所有几何元素的位置.

1.1　分析阶段

分析阶段主要包括建立约束图和对约束图进行分解并建立分解树，其中，建立约束图比较简单，具体参见文献[11]，以下主要阐述分解过程.

首先介绍将要用到的一些图论[12]方面的基本概念. 假设 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E})$ 是一个连通图，如果存在两个顶点 u 和 v，在它们之间的所有路径中都包含有中间顶点 a，则称顶点 a 为关节点. 如果一个连通图中不存在关节点，则称这个图为双连通图. 显然，根据关节点可以将一个图分解为一些子图，其中每一个子图都是双连通图，这种分解称为 **I** 型分解. 与之相类似，如果一个图根本不是连通图，将其分解为一些连通子图，那么称这种分解为 I 型分解.

设 a 和 b 是一个双连通图 $\boldsymbol{G}$ 中的两个顶点，可

以据此将图 G 中的边分成几个等价类．两条边在一个等价类的条件是：如果存在一条路径包含这两条边，并且点 a 和 b 不是这条路径的中间顶点，那么这两条边在同一个等价类中．由此可以定义双连通对的概念：如果在一个双连通图 G 中存在两个点 a 和 b，可以将图 G 中的边分为两个以上的等价类，或者将图 G 中的边分为两个等价类并且每个等价类包含一条以上的边，则称这两个点为双连通对；如果一个双连通图中不包含双连通对，则将之称为三连通图．

由于双连通对可以将一个双连通图中的边分解为等价类 $E_1,E_2,\cdots,E_k$，可以将图 G 分解为子图 $G_1,G_2,\cdots,G_k$，其中图 $G_i(V(E_i),E_i\cup\{(a,b)\})$，并称之为Ⅲ型分解．当然，Ⅲ型分解后得到的子图仍是连通图．注意，在分解的时候，如果原来图 G 不存在边 ab，则将把边 ab 加到子图中，这时称边 ab 为虚边，它将在后续求解中发挥功用．

分解过程的主要目的是将约束图逐步分解为尽量简单的三连通分量，并且根据分解的过程将原约束图表示为一棵分解树．该方法有些类似于在求解欠约束问题时提出的分解算法[13]，但该方法不局限于寻找三角形虚边，对约束图的分解更加彻底，这样可以适应更复杂的约束情况．主要步骤如下：

(1) 将整个约束图设为当前约束图 G_c，建立一个只包含根节点的分解树 T_c，并设此根节点为当前节点，以代表当前约束图．

(2) 如果当前图 G_c 不是连通图，采用Ⅰ型分解的方法将它分解为 m 个连通图($m>1$)．标记当前节点为Ⅰ型节点，并在当前节点下增加 m 个子节点，分别代表每一个连通图；然后分别将每一个子节点置为当前节点 T_c，其所代表的子图设为 G_c，并执行下一步．

(3) 如果当前图 G_c 不是双连通图，则找出所有的关节点，并采用Ⅱ型分解的方法将它分解为 m 个双连通图($m>1$)．标记当前节点为Ⅱ型节点，在当前节点记录下这些关节点，并在当前节点下增加 m 个子节点，分别代表每一个双连通图；然后分别将每一个子节点置为当前节点 T_c，其所代表的双连通图设为 G_c，并执行下一步．

(4) 如果当前图 G_c 含有双连通对，则根据此两点采用Ⅲ型分解的方法将当前图分解为 m 个子连通图($m>1$)．标记当前节点为Ⅲ型节点，在当前节点记录下这对双连通点，并在当前节点下增加 m 个子节点，分别代表每一个连通图；然后分别将每一个子节点置为当前节点 T_c，其所代表的双连通图设为 G_c，并执行第3步．

(5) 如果当前图 G_c 是三连通图，则看一下这个图中有没有虚边，如果存在虚边，则将这条虚边在图中去掉，然后执行第4步．

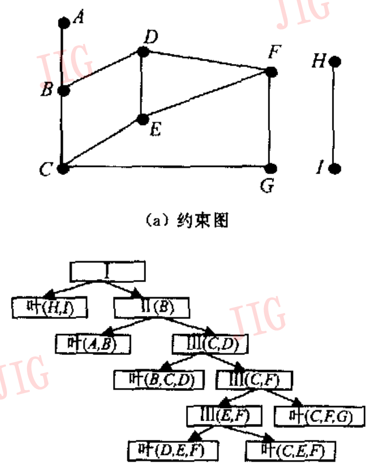
(6) 如果分解树上所有的叶节点所代表的图都是三连通图，且都不包含虚边，则算法停止．

当算法停止时，分解树中的所有叶节点都代表了一个三连通图，而中间节点则分为3种类型，分别代表前面所进行的3种分解．另外，从建立分解树的过程可以发现，分解树有以下几个重要性质：

(1) Ⅰ型节点总是出现在树的第1层，即根节点；

(2) Ⅱ型节点的父节点是Ⅰ型节点；其子节点要么是叶节点，要么是Ⅲ型节点．

图2表示了一个约束图和对它进行分析得到的分解树，其中分解树的节点注明了这个节点的类型，而括号中则注明了其所对应的关节点（Ⅱ型节点）、双连通对（Ⅲ型节点）或包含的节点（叶节点）．



(a) 约束图

(b) 图2(a)的分解树

图2 一个约束图及其分解树

1.2 构造阶段

构造阶段主要包括两个部分：第1部分在整个算法中占有中心地位，它负责对分解树进行分析，判断约束图是完全约束问题、约束不足问题还是约束过载问题，如果是约束不足问题，则自动增加所需的约束，如果是约束过载问题，则判别它是否是一致性过约束．在对分解树进行分析的同时，系统还将产生

一系列构造步骤；而在第 2 部分，系统将执行第 1 部分产生的构造步骤，计算出几何元素最后的位置.

1.2.1　基本思想

对分解树进行分析的顺序主要是采用自底向上的方法：首先计算最下面的叶节点，即三连通子图中几何元素之间的相对位置；然后再逐步对子节点进行合并，通过确定各子节点之间几何元素的相对位置得到高层节点的解；最后到达根节点，从而求出整个约束图内几何元素之间的相对位置. 这样由于已经分析出整个约束图中所有几何元素之间的连接关系，不仅可以使求解过程由总体下降到局部，提高求解的效率和可靠性，而且系统可以在出现病态约束（约束过载或约束不足）的情况下，方便地确定病态约束的原因，从而使通过自动或者手工的方法进行修正变得简单.

采用哪种方法求叶节点内几何元素之间相对位置主要看其所对应的三连通图的大小. 图 3 是完全约束的三连通图拓扑结构的几种最常见情形，对于其中第 1 种和第 2 种结构，可以采用简单的代数方法进行求解；而对于第 3 种结构，则可以采用 Bouma 的扩展方法[7]对之进行求解；对于更复杂的情况，由于可能产生高于四次的方程，用代数方法进行求解比较困难，因此可以将约束转化为方程，并通过数值迭代的方法进行求解.

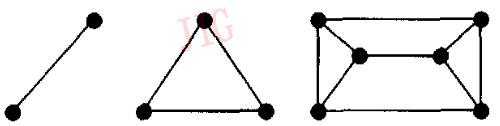


图 3　三连通图

得到叶节点内几何元素的相对位置以后，可以通过对子节点进行合并完成对父节点的求解. 根据前面对分解树的建立过程的分析可以知道，Ⅰ型节点只有在根节点才有可能出现，并且由于Ⅰ型分解的子图之间不存在任何约束，使这些子图间几何元素的相对位置关系可以任意确定，因此在以下叙述中将省略Ⅰ型节点，仅考虑Ⅰ型节点下的子树，这些子树仅包含Ⅱ型和Ⅲ型两种中间节点. 现在将合并过程分为以下几种情况：

（1）Ⅱ型节点

这种节点下面的子节点之间是通过关节点进行连接的，即使它们的子节点都是完全约束节点，所得到的子图也一定是欠约束子图，并且每一个关节点都代表着增加一个自由度，因此只需要使它下面所有节点都是完全约束节点，由于Ⅱ型节点的子节点都是Ⅲ型节点或叶节点，因此这种规定很容易实现. 由于每个关节点都需要增加一个自由度才能补充它所产生的自由度亏损，因此通过对每个关节点增加一个约束的方法，将它下面的子节点两两归并，而达到合并的效果.

（2）Ⅲ型节点

对于这种节点来说，如果它的父节点也是Ⅲ型节点，将直接考察它的父节点，因此最后实际考察的是一棵子树，这棵子树中所有中间节点都是Ⅲ型节点，叶节点由原分解树的叶节点或Ⅱ型节点组成. 如图 4 所示，中间节点 a、c、d 是Ⅲ型节点.

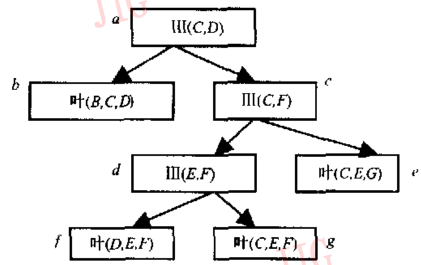


图 4　Ⅲ型节点子树

Ⅲ型节点下面的子节点是通过一对双连通对关联起来的，并且它的直接子节点都包含这两个双连通点，因此Ⅲ型节点下面的子节点之间的唯一联系就是这两个关节点对之间的约束关系，只需要求出这个约束关系，结合子节点内部的几何元素的相对位置信息，就可以通过简单的几何变换将子节点结合起来.

求解方法是在子树中找到所有已经完全约束（即 C 值等于 0）并且不包含虚边的叶节点，并求出其祖先节点所对应的双连通点之间的相互位置关系，如果本来这两个点之间就存在边，则只需要对其兄弟节点进行计算，然后通过简单的平移变换把这些子节点拼接起来，但如果这两个点之间并不存在边，这时在前面增加的虚边将发挥用处，因刚才增加的虚边并不存在参数值，所以在这时需要通过双连通对的相对位置信息得到其参数值，从而计算出其兄弟节点，最后通过平移变换将所有子节点组装到一起. 通过这种方法，可以将树上的所有节点都求解出来.

图 4 所示是图 2(b)分解树中的一个子树，叶节

点 f 是一个完全约束节点，且不包含虚边，因此可以先求出其中 3 个几何元素之间的相互关系，从而得到 EF 之间的相互位置关系，但是由于 EF 在这里并不是虚边，因此直接对其兄弟节点 g 进行求解；g 节点是一个Ⅲ型节点，但是包含一个虚边 CF 没有求解，于是增加约束边 CF 使其成为完全约束节点，然后通过几何变换将节点 g 和 f 合并到一起，就可以得到节点 d 所包含的几何元素的相对位置。知道了 CF 之间的相对位置关系，并且 CF 是节点 e 的虚边，可以由此计算出 CF 这个约束边的参数值，并对节点 e 进行求解，然后通过几何变换将节点 e、d 合并起来，得到节点 c，并得到 C、D 之间的相对位置关系。由于 CD 也是虚边，可以通过得到 CD 的参数值求出节点 b 内几何元素的相对位置；然后通过几何变换将 b、c 结合起来，从而求出节点 a 包含的所有几何元素的相对位置。

1.2.2 过约束问题一致性的判定

对于一般的约束问题来说，如果其约束图的子图中有一个子图满足不等式 $|\boldsymbol{E}|>2|\boldsymbol{V}|-3$，就是过约束问题。但是在实际设计过程中，设计人员经常会增加冗余约束，使之成为过约束问题，但是这种约束的特定参数值一般仍然保证约束问题存在解，这就是一致性过约束问题。图 5 所示，就是一个简单的一致性过约束问题，8 个几何元素之间存在 14 个约束，但仍然存在解。

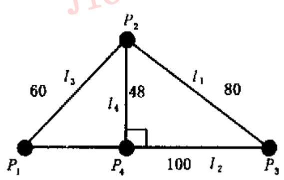


图 5 一致性过约束问题

在Ⅲ型分解过程中，如果所得到的子节点有一个以上的完全约束节点，这些节点都不存在虚边，并且Ⅲ型分解所对应的双连通对之间没有边，那么可以判定这时处于过约束状态，可以对这些完全约束节点进行计算，得到在每一个节点中双连通对的相对位置关系。Ⅲ型分解的子节点之间的唯一联系是双连通对的相对位置关系，因此，可以比较一下在每一个完全约束节点中双连通对节点的相对位置关系，如果是相同的，则可以判定这是一致性过约束问题；如果不相同，则说明约束集存在矛盾，由系统通知用户错误信息。

图 5 所对应的约束图如图 6 所示，可以先通过双连通对 P_2 和 l_2 将约束图分为两部分，即图 6 中用虚线勾勒的那两部分，再对这两部分做进一步的分析发现，这两部分都是完全约束图，且不存在虚边，即是过约束图。但是当对这两部分分别进行计算后发现，P_2 和 l_2 在这两部分的相对位置都是一样的(点线距离 48)，因此这两部分并不矛盾，是一致性过约束问题。

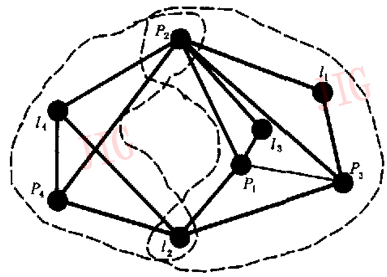


图 6 图 5 对应的约束图

2 实例与结论

图 7 是一个测试的实例，图中共有 200 多个几何元素，约 400 个约束方程，利用该方法，在联想 PⅡ350机器上只需要 5.8s，而采用文献[2]中的数值算法，它在同样的平台上处理该草图需耗时 24s，效率提高的关键在于，原文献[2]的数值方法涉及到

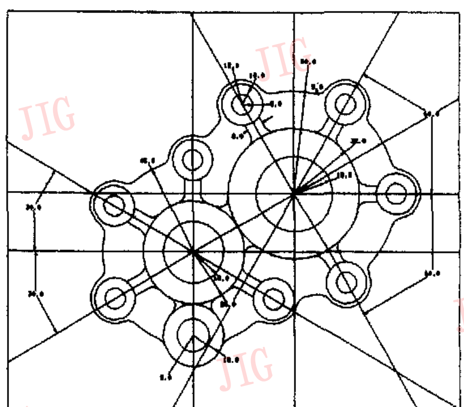


图 7 约束求解的实例

大型联立代数非线性方程组的求解,而该方法采用了图构造的方法,并适当加入了虚约束,将需要大型联立代数非线性方程组的求解的问题,分解为一系列局部的几何求解问题,由于主要的尺规作图问题都可以直接进行几何求解,从而大大地提高了求解效率. 对于其他方法,如符号方法[4,5],由于该类方法一般具有指数阶复杂度,不适宜于大型约束系统,因此不具有可比性. 文献[2]是数值方法中最经典的方法,具有效率高,并可以求解欠约束的情况,因此被用于作为对比参考. 基于图分解的几何约束求解方法能够处理各种在实际设计过程中能够遇到的约束情况,包括完全约束、欠约束和过约束情况. 该方法已在自行开发的基于特征的参数化造型系统 GS-CAD 中实现,并取得了良好的效果.

参 考 文 献

1 Light R A, Gossard D C. Modification of geometric models through variational geometry [J]. Computer-Aided Design, 1982,14(4):209~214.

2 Aldefeld B. Variational of geometries based on a geometric-reasoning method[J]. 1988,20(3):117~126.

3 陈立平,彭小波,王波兴等. 一种面向欠约束几何系统求解的二部图匹配优化处理方法[J]. 计算机学报,2000,23(5):523~530.

4 Buchberger B, Collins G, Kutzler B. Algebraic methods for geometric reasoning[A]. In: Annual Review Computer Science [C], Palo Allo: Annual Reviews Inc. ,CA, USA, 1998, 3: 85~120.

5 Kondo K. Algebraic method for manipulation of dimensional relationships in geometric models[J]. Computer-Aided Design, 1992,24(3):141~117.

6 Owen J C. Algebraic solution for geometry from dimensional constraints [A]. In: ACM Symposium Foundations of Solid Modeling[C], New York: ACM, 1991: 397~407.

7 Bouma W, Fudos I, Hoffmann C M *et al.* A geometric constraint solver[J]. Computer Aided Design, 1995, 27(6): 487~501.

8 Kramer G A. Solving geometric constraints systems: A case study in kinematics[J]. Cambridge, MA: MIT Press, 1992.

9 董金祥,葛建新,高屹等. 变参绘图系统中约束求解的新思路[J]. 计算机辅助设计与图形学学报,1997,9(6):513~519.

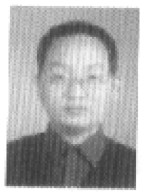
10 李海龙,董金祥,何志均等. 一种利用有向图优化约束求解的方法[J]. 软件学报,1997,8(增刊):155~163.

11 Fudos I. Editable representations for 2D geometric design[D]. Department of Computer Science, Purdue University, Indiana, USA, 1993.

12 王朝瑞. 图论(第 2 版)[J]. 北京:北京理工大学出版社,1997.

13 Fudos I, Hoffmann C M. A graph-constructive approach to solving systems of geometric constraints[J]. ACM Transactions on Graphics, 1997,16(2):179~216.

何　伟　1972 年生,现为浙江大学计算机系博士生. 主要研究方向为几何造型、约束求解、网络安全.



唐　敏　1974 年生,1999 年获浙江大学计算机系博士学位,现为浙江大学计算机系副教授. 主要从事 CAD、几何造型等领域的研究工作.

董金祥　1945 年生,浙江大学计算机系教授,博士生导师. 主要从事 CIMS、用户界面、图形学等领域的研究工作.

何志均　1923 年生,浙江大学人工智能所教授,博士生导师. 主要研究方向为人工智能、软件工程等.