

一个任意图的输出和编辑系统*

李文辉 庞云阶

(吉林大学计算机科学系 长春 130023)

摘要 本文给出了一个对任意图的输出算法,这个算法同其它的同类算法相比,具有更小的时间复杂性,并且由于此算法的参数是可控制的,所以对图的输出也是可控制的.另外,本文描述了一个图的显示和编辑系统 GLEAM,这个系统是基于所给出的新算法的,并且在本系统中使用了约束技术,使图的输出更美观、合理. GLEAM 是一个通用且可以扩充的系统,用户只要加入相应的具体领域知识,就可将其变为一个专用的系统.

关键词 图,图输出,一般图,约束,模拟退火.

在计算机科学中,图是一种非常普通的数据结构,也是非常重要的数据结构.尤其是在数据库设计、状态图表示、自动机理论和软件工程中,图更是必不可少的概念.图的图形化输出则是理解和解决图问题的一个很重要的手段,常言道:“一幅画顶一千句”,文献[1]就给出了这样一个很好的例子.传统上,图是通过人工交互的方式来绘制的,且要求图的输出尽可能的简单明了.在处理大的图结构时,要做到图的输出合理,使点和边能够合理地放置来达到图的美观,并不是件容易的事.事实上,在大多数情况下,要想得到一个最优的结果几乎是不可计算的,所以往往使用启发式或概率来使处理问题过程快速而简化,以达到较好的综合效果.由于存在众多种类的图及给出一个通用的问题解决方法的困难,使大部分的研究者都把它们的研究局限于某些特殊的情况,如树、平面图或特殊的输出方式,比如层次输出及网格输出.在这些方面已经有了很多的研究成果,但几乎没有人涉及到处理一般的图的问题,即对通用解法的研究.在 GLEAM 系统中,我们给出了通用的图的输出算法,它使用了模拟退火方法,与同类问题的算法相比,具有更小的时间复杂性.

GLEAM 系统是可扩充的,它是一个应用平台,用户在此基础上,可以加入具体的领域知识和要求,使其很方便地被改变为适应用户的特殊要求的系统.

1 图及图的显示

我们知道,图是一个抽象的数据结构.它的基本定义可简述为 $G = \{V, E\}$,其中 V 为图

* 本文研究部分得到国家教委回国留学人员科研基金、国家自然科学基金和国家教委“符号计算与知识工程重点实验室”资金资助.作者李文辉,1961年生,博士,副教授,主要研究领域为计算机图形学,人工智能.庞云阶,1939年生,教授,主要研究领域为计算机图形学,人工智能.

本文通讯联系人:李文辉,长春 130023,吉林大学计算机科学系

本文 1996-01-22 收到修改稿

G 的点集, 而 E 为图 G 的边集. $e=(v_i, v_j) \in E, v_i, v_j \in V$. 如果在边中, v_i, v_j 的顺序是有意义的, 则 e 称为有向边, 那么由有向边组成的图, 则叫作有向图; 否则称为无向图. 把这样一个抽象的结构集合转化为可见的图形表示, 就被称为图的输出. 对于图的输出, 存在许多的问题需要研究. 例如, 节点的形状和大小及它们之间的位置关系和相互连线的形式. 以手工方式来画小的图是可以接受的, 而画大的图, 则是非常痛苦的事, 烦燥乏味且不说, 更重要的是极易出错且结果图也往往不能被接受. 另一个类似的困难是, 当一个给出的图输出, 要对其进行增加或减少来进行修改时, 以人工的方式来修改, 则与绘制整个图一样是乏味的, 因为它常常需要重新放置许多点和边. 当考虑更加复杂的图形客体, 如超图(Hypergraphs)或高图(Highgraphs)时, 则变得更加困难, 而这种复杂图已在系统设计、数据库理论和知识表示中得以应用.

图的显示也是有许多方式的, 如图的树形显示、平面图显示、分层次显示等等. 对于图的显示算法也是非常多的, 文献[2]给出了很好的总结. Eades 首先提出了弹性模型^[3], 后来, Kamada^[4]和 Fruchterman^[5]又分别对此模型提出了进一步的改进, 它们的算法是针对无向图的显示来设计的, 对树结构的输出, 文献[6]给出了相应的算法. 对于图的显示是否美观这一问题, 可以说, 没有确定的审美标准, 它往往因用户的不同的需要也有不同的标准, 我们总结出一些图的显示标准:

- * 最少边交叉: 尽量使图中交叉的边数成为最少, 也叫图形显示的平面化.
- * 节点分布一致化: 使显示图中的节点空间分布一致.
- * 边长一致性: 图中的边长应该一致.
- * 节点的层次: 应使具有同性质的节点在同一层上.
- * 显示宽度: 图的显示应具有最小的宽度.
- * 最大化对称性: 应使图的分布具有对称性.
- * 同型性子图: 具有相当结构的子图, 应以相同的图形显示.
- * 最小化总边长: 使图中的全部边长的总和为最小.

以上只是一些最基本的标准, 当然对不同的用户, 可能给出不同的美学标准, 这些标准往往也不能同时被满足, 因为很多标准是互相冲突的. 我们在 GLEAM 系统中, 引进了一个新的图显示算法, 就是利用模拟退火技术来解决通用的图显示问题.

2 模拟退火技术

模拟退火 SA(simulated annealing)是一个灵活的优化方法^[7], 它适合于大的组合优化问题. 基于统计理论, 且已被成功地应用到一些典型的组合优化问题求解, 例如, 对 VLSI 的设计和显示. SA 被有效地应用于那些具有非常大的离散空间的问题. 一般的递归方法是先给定系统初始值, 然后, 依据一定的算法选择一个新的状态, 并对其估值, 然后可能用其来置换先前的状态, 重复执行这一动作, 直到某些停机条件达到为止. 程序结束在一个最小状态, 但一般说来是局部最小, 而不是所要求的全局最小. SA 方法试图通过使用从液体冷却成晶体状态的类似规则来跳过局部最小, 这一过程称为退火(Annealing).

SA 方法可简单地定义如下:

- (1) 选择一个初始状态 σ 和一个初始温度 T , 且给出一个正整数 N ;
- (2) 重复下列操作 N 次;
 - (a) 从 σ 的邻近状态中选一个新的状态 σ'
 - (b) 设 E 和 E' 分别对应于 σ 和 σ' 的代价函数值.
如果 $E' < E$, 或随机数 $\alpha < e^{(E-E')/T}$, 则置 $\sigma \leftarrow \sigma'$, 其中 $0 < \alpha < 1$.
- (3) 降低温度 T ;
- (4) 如果结束规则被满足, 则停止; 否则返回(2).

对代价函数有一个要求, 就是它不应该过陡地下降, 如果一个含有所需要的最小点的峰过窄, 则此最小点被发现的概率就很小. 为了可由 SA 算法来求解, 问题也必须允许在最优解附近的解应是可接受的, 因为 SA 往往很难发现最小值本身. 另一个重要的要求是, 用代价函数来计算一个新的状态的执行必须简单, 应可以利用旧值来计算新值, 这是因为这些计算总要重复去做, 且它是 SA 实现的重要计算工作.

3 GLEAM 系统中的图输出算法

在 GLEAM 系统中, 我们把模拟退火技术嵌入到图的输出算法中, 用它来解决图的输出通用算法问题. 在我们的算法中, 使用了如下几个图的输出标准: (1) 节点的平均分布; (2) 图中边的一致性; (3) 最小边交叉; (4) 节点不能太靠近显示区域的边.

下面讨论这一基于模拟退火技术的图输出算法.

3.1 初始状态与相邻状态

我们这个算法的输入是一个邻接表或邻接矩阵, 它来表示图中节点的关系. 图中点的初始值是由随机函数任意给定的, 它就构成了 SA 算法的初始状态. 所谓相邻状态, 就是当前态的下一个状态. 我们要求相邻状态必须与原状态是相似的. 在此情形下, 它们只表示对系统状态微小扰动.

在算法的执行过程中, 如何选取合适的相邻状态是很重要的, 一般是把点的移动量限制在以原节点的位置为圆心的同心圆上, 给出一定的半径, 以确定选择新位置的区域. 在处理开始时, 此同心圆半径很大, 但随着算法的执行, 则逐渐变小. 状态变得越好, 则允许的扰动就越小. 每次求相邻状态, 我们都是针对某一个节点, 其它的节点保持不变. 我们选取对能量变化影响大的节点为需要调节的节点, 至于新的位置, 则可根据能量公式的计算求得. 图 1 为 1 个任意的初始状态, 通过计算, 确定节点 2 为应首先调节的节点, 它的调节半径如图中的虚线圆区域.

3.2 代价函数

在应用模拟退火过程中, 最重要的步骤是定义能量或代价函数, 在我们的情况下, 它控制一个看起来美观的显示的含义, 这样, 最终图的全部所需要的特征都必须被转换为对此函数的贡献, 并附有它们的相对重要性的权值. 下面我们给出方程中要使用的标准, 并给出可行的实现.

A. 节点的分布: 一个美观图的一个基本的标准是图中

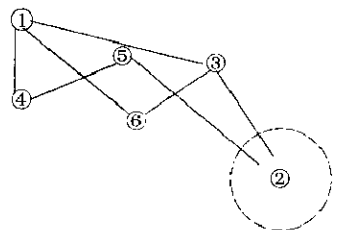


图1 相邻状态的选择

的节点要均匀分布,节点之间的距离不必完全一致,但图应占有绘图空间的有效部分,可能的话,节点不应挤得太满.这一点可由我们的能量方程的 2 个部分来完成.第 1 个可以阻止节点间太靠近,第 2 个则解决绘图空间的边界问题.对每一节点有 $h_{ij} = \lambda_1/d_{ij}^2$, 把其加入到能量公式,其中 d_{ij} 为 i, j 节点的距离, λ_1 是定义此标准相对于其它标准的重要性的正规化因子,增加 λ_1 使节点间的距离变小.

B. 边界线:在物理上真正最小化势能可能引进元素间的无限制地分散,为了避免这种现象,在能量方程中加入解决边界的功能部分,即加入

$$f_i = \lambda_2 (x1_i^{(-2)} + x2_i^{(-2)} + x3_i^{(-2)} + x4_i^{(-2)}).$$

其中 $x1_i, x2_i, x3_i, x4_i$ 分别代表点 i 到右、左、上、下边界的距离.这样,点被控制不能太靠近边界,但它们也不能呆在绘图空间的中央,这是由能量方程中的第 1 项决定的.很清楚,当 λ_1 相对于 λ_2 增加时,点更聚向中央;同时,减小它将使使用更多的靠边界的空间.

C. 图中边的长度:一般图的输出的要求是使边尽可能地短,但不要引起整个图太集中.可用下面的方法来实现此标准,即对每一个边 k , 长度为 d_k , 在能量方程中加入一项 $g_k = \lambda_3 d_k^2$, 其中 λ_3 是一个适当的正规化因子.用此短边标准可以使图中的一些不必要的交叉点被消除,但有些图还是需要更直接的对交叉边的处理.

D. 边交叉:一般说来,最小化边交叉是一个重要的目标,但是实现起来非常困难.产生没有交叉边的图或平面图的算法是存在的,但此标准往往与图的美观性标准相矛盾,所以,常常需要人们在这两者间进行选择.

我们通过简单地增加每 2 个交叉边的一个惩罚常数 λ_4 于代价函数中,并把此代价函数引入算法.当然,增加 λ_4 意味着对消除边交叉的要求更强烈,将导致图中更少的交叉边出现,可是,这也可能会使其它的标准受到削弱.

E. 点-边距离:前边讨论的情况惩罚了边交叉,但却没有惩罚那些特别靠近但没有交叉的边.在一些情况下,这种情形已被节点的分布项所考虑了,但我们仍愿意给予更多的注意.我们定义一个点到一个边的距离为最短距离.这一距离对代价函数的贡献是反比于它的平方,即 $R_{ij} = \lambda_5/m_{ij}^2$, m_{ij} 为点 i 到边 j 的距离.

3.3 冷却方法

初始温度:由于初始状态是任选的,所以,我们置初始温度足够高,以接受几乎所有的在开始时的移动.在某些情况下,如果我们知道初始状态与最终状态有一些相似,则我们可以选择较低的温度,以保持已有的部分结果.

温度的降低:SA 算法是通过不断地降低温度来改变系统的状态,以期达到最佳.一般对每一温度移动次数的选择,都采用线性逼近的方法.我们通过实验发现,状态的改变,大都在最初的几次,后面的阶段,则状态改变的可能性很小,为了加快算法的执行,我们在算法执行过程中,记住成功改变状态的频率,来控制动作是否继续.当此频率非常之低,则可以降低温度,但对每一温度的测试的上限应该规定.

我们的冷却规则是几何的,如果 T_p 是第 p 阶段的温度.那么,在下阶段的温度为

$$T_{p+1} = rT_p.$$

其中 $0.5 \leq r \leq 0.99$, r 是用来控制冷却的速度.冷却太快将导致偏离局部最小的不稳定性.即结果往往与优化的相差很远;冷却太慢,则要增加运行时间.此算法的 2 个图的输出示例

如图 2 所示.

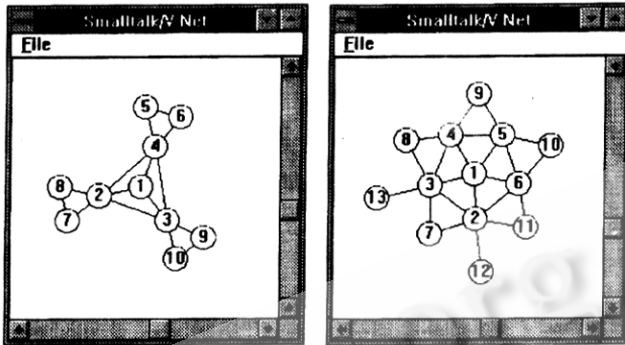


图 2 算法的输出图例

4 图的编辑工具

本系统提供了一个非常友善的用户界面,它可以方便地定义和编辑图,并能浏览大的图.下面我们分别来讨论此编辑工具的主要功能.

4.1 图的交互编辑功能

此系统可以由交互编辑模块来实现对图的定义、输入、修改等操作.

首先,此系统既可接受由邻接表结构表示的图,也可以由人机交互的方式来完成图的输入.它的主要过程是,人可随意地在屏幕上选定位置作为图中的节点位置,节点间的关系由节点间的连线来实现.用户可以通过鼠标器来选择要拾取的节点,然后在它们之间定义关系,并加上路径的长度.

对于一个已经存在的图,用此模块可以对其进行修改,例如,可以增加一些节点或删除一些节点,也可增加或删除图中节点之间的关系,即连线.可以通过节点形状的菜单来选择节点的形状,并可以调用字处理模块对节点进行标识.

在主菜单中,有一个对图的显示方式的选择项,用户可以根据实际情况选择不同的显示方式,如树形显示、层次显示和同心圆方式的显示等等.

利用此界面还可以对图进行放大、缩小、旋转和平移等变换.这些操作要事先选取被操作的目标,然后选取具体的变换方式,并给出参数,这样则完成了一次变换.这些变换可以是级连,其参数的选取也是由菜单图示来直观给出的,而不需要用键盘键入.

4.2 图的浏览

图的显示画面不局限于屏幕的尺寸,用户可根据需要来自己定义显示区域.对超出屏幕的部分,可以采用上、下、左、右滚动的方式来实现观看.如对某一区域感兴趣,则可选择此区域进行放大,也可以通过缩小来观看整个区域,以了解图的全貌.

在图的浏览过程中,用户可以对节点的属性值和坐标值进行察看、修改.由于本系统使用了基于模拟退火的解自动输出算法,那么图的输出显示完全系统决定,不需要人工来进行干预,这对大图的显示是非常有意义的.

4.3 图输出中的约束

约束是用来定义输出图中节点间的相互关系,它是体现用户要求的有效方法,把约束技术应用到输出算法中^[6,9],也是本系统的一个特点.

在图编辑过程中,也可以对约束进行编辑,具体地说,可以通过选取菜单中约束一项,来选择约束类型,然后选择被约束的客体,这样就给客体加上了约束.也可以删除某个已有的约束,使客体之间的关系得以释放.这种用交互图示的方式来定义约束,是非常方便的.它摆脱了以往用程序编码来定义约束,或以方程的输入方式来给出约束的方法,使约束的定义更直观、更简单.

由于约束技术的使用,使图输出具有了稳定性的优点.在一般的图输出系统中,因为缺乏稳定性,往往对结果图的微小改动,会使结果完全不同,使用户对图的输出结果没有办法预测.在本系统中,系统把每一个输出结果都自动加上约束,这样,当对某一结果进行微小改动后,由于约束的作用,就使得前后2个结果有很大的相似性,保证了输出结果的稳定性.

5 GLEAM 系统的实现与应用

本系统是用面向对象的语言 Smalltalk 来实现的.整个系统分为几个大的模块,主要有图的输出算法模块、人机界面模块、约束的定义和解决模块、图形处理模块、知识存储和处理模块.这些模块有机地结合,构成了 GLEAM 系统的完整功能.

本系统中,我们用 Smalltalk 语言把要处理的客体定义成一些类(Class),对每一个类都定义了一些方法,这些方法可具体实现对此类的操作.主要的类有: GraphLayout, Constraint, GraphEdit, GraphView, Application 等.这些是主要的类,每一类下面还有一些子类,由于面向对象的语言有很好的继承性,使子类的定义和扩充变得更方便,例如:类 Constraint 的子类(Subclass)有: LineConstraint, CircleConstraint, BoxConstraint, Parral-Constraint 等,这些子类除了继承它们的超类(Superclass):Constraint 的特性外,又增加了特殊的方法来处理各自的特殊的要求.

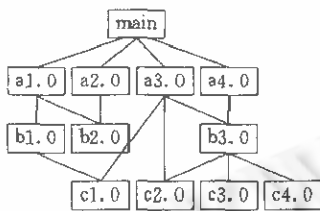


图3 应用示例:程序调用图

由于本系统是一个可以处理一般图结构的系统,即它具有有一般性.那么,对于特殊领域的用户,可以利用这个平台,很方便地构造出自己的专用系统.例如,我们把它应用于软件设计的可视化,通过定义类 Application 的子类 Software 来实现把具体的知识和要求加入到本系统中来.因为软件设计,

主要是模拟各功能块之间的调用关系,一般表现为层次关系,也就是说,它所要求的图是具有很强的分层次概念,那么,我们把这种特点,以一个因子的形式加入到代价函数中去,当执行此算法时,系统则可保证输出结果满足用户的要求.图3是软件设计中的程序调用的模拟.

6 结束语

本文给出的 GLEAM 图的输出和编辑系统,是一个通用的可扩充的对图这一抽象数据结构进行处理的系统.本系统使用了新的图的显示技术,即把模拟退火方法应用到图的算法中去,实践证明这是一个有益的尝试.

致谢 本课题是吉林大学与德国 Bremen 大学的联合科研项目. 它部分地由德国基础科研基金和 Bremen 大学科研基金资助. 这一工作是本文第 1 作者在德国工作其间完成的, 在本工作的完成过程中, 得到了 Bremen 大学 Frieder Nake 教授的指导和帮助, D. Harel 教授对此工作提出了许多建设性的意见, 在此一并表示感谢.

参考文献

- 1 Gabriel Robins. The ISI grapher; a portable tool for displaying graphs pictorially. Helsinki, Finland, Symbolikka'87, 1987. 17~18.
- 2 Eades P, Tammasia R. Algorithms for drawing graphs; an annotated bibliography. Technical Report CS-89-09. Dept. of Computer Science, Brown University, Oct. 1989.
- 3 Eades P. A heuristic for graph drawing. Cong. Numer., 1984, 42:149~160.
- 4 Kamada T, Kawai S. An algorithm for drawing general undirected graphs. Inf. Proc. Lett., 1989, 31:7~15.
- 5 Furtherman T M G, Reingold E. Graph drawing by force-directed placement. Software-Practice and Experience, 1988.
- 6 Reingold E M, Tilford J S. Tider drawing of trees. IEEE Trans. Software Eng., 1981, SE-7(2):223~228.
- 7 Kirkpatrick S. Optimization by simulated annealing. Science 220, 1983.
- 8 Li Wenhui, Pang Yunjie. Interactive graph layout with constraints. In: Proc. of the 4th International CG/CAD Conf., 1995.
- 9 Li Wenhui. CAGL, a constraint based automatically graph layout system. In: Proc. of Golden West International Conf. of Intelligent System, Nevada, USA, 1992.

A GRAPH LAYOUT AND EDIT SYSTEM

LI Wenhui PANG Yunjie

(Department of Computer Science Jilin University Changchun 130023)

Abstract This paper describes GLEAM a graph layout and edit system. The authors present an algorithm for drawing general graphs. It needs fewer time than other algorithms for drawing graphs, and can lay graph nicely out. The system is extendible. User can add some special knowledge into the system easily, and make the system for special application.

Key words Graph, graph layout, general graph, constraint, simulated annealing.