

# NLAPT: 二维平面上网络布局算法性能评测软件

何 胜<sup>1,2</sup>, 柳益君<sup>1,2</sup>, 叶飞跃<sup>1,2</sup>, 潘 瑜<sup>1,2</sup>, 冯新翎<sup>1,2</sup>

(1. 江苏理工学院计算机工程学院, 江苏 常州 213001)

(2. 常州市云计算与智能信息处理重点实验室, 江苏 常州 213001)

**[摘要]** 网络布局算法是网络可视化领域的主要研究内容,一般集成于通用的可视化平台。网络布局算法性能的评测为新算法的设计及不同算法性能相互比较提供重要参考。针对可视化平台上布局算法性能评测问题,设计一种基于二维平面的评测软件 NLAPT,该软件基于网络布局的边-边交叉率、节点-边交叉率、相对边长度 3 个主要测试指标,设计与平台无关的接口函数和测试算法,采用 Java 语言编写。实验结果表明评测的正确性和有效性,软件能方便应用于不同的网络可视化平台,为研究者提供有效的评测。

**[关键词]** 网络可视化,布局算法,算法评测

**[中图分类号]** TP393 **[文献标志码]** A **[文章编号]** 1672-1292(2016)03-0076-05

## NLAPT: A Software for Performance Test of Network Layout Algorithm

He Sheng<sup>1,2</sup>, Liu Yijun<sup>1,2</sup>, Ye Feiyue<sup>1,2</sup>, Pan Yu<sup>1,2</sup>, Feng Xinling<sup>1,2</sup>

(1. School of Computer Engineering, Jiangsu University of Technology, Changzhou 213001, China)

(2. Key Laboratory of Cloud Computing & Intelligent Information Processing of Changzhou, Changzhou 213001, China)

**Abstract:** In the field of network visualization, it is essential that research network layout algorithms are integrated usually into visualization platforms. Evaluating effectively the performance of algorithm will be of great help for design of new algorithm and comparison among different algorithm. For the purpose of testing the performance of layout algorithm on visualization platforms, a software named NLAPT is proposed with Java language to compute ratio of edge-edge, node-edge crossings, and relative edge lengths on 2D plane and the generalized software interface and test algorithm are implemented. Experiments show the effectiveness and validity of NLAPT. NLAPT can be integrated conveniently into various visualization platforms, providing users useful software to evaluate the performance of layout algorithm.

**Key words:** network visualization, layout algorithm, algorithm test

网络大数据时代,通过网络建模能够将数据抽象成节点和边等网络元素,从而构建社会关系网络、生化网络、学术论文互引网络等,网络可视化技术能方便展现这些复杂网络的直观图景,帮助人们理解网络结构并寻找其中蕴藏的规律<sup>[1-3]</sup>。网络布局算法根据网络节点的拓扑连接关系,通过优化目标函数计算出网络节点的几何位置并绘图,是网络可视化技术的重要研究内容<sup>[4-6]</sup>。一般来说,一个性能优越的网络布局算法所产生的绘图结果应当具有较少的边-边交叉率和节点-边交叉率、较小的相对边长度等特性,如何有效评测网络布局算法性能是网络布局算法设计和应用需要面对的首要问题<sup>[5,7-9]</sup>。当前流行的可视化平台包括 cytoscape<sup>[10]</sup>、Pajek<sup>[11]</sup>、visAnt<sup>[12]</sup>等,研究者常常需要基于这些平台研究并开发网络布局算法。然而在上述流行的软件平台中,目前都没有提供评测算法的工具,因此研究者不得不自行编写评测算法,如 Inoue<sup>[13]</sup>等基于 MATLAB 编写评测函数评估所开发的网络布局算法性能,这些针对特定编程环境的评测函数往往只能应用于算法开发者所定义的特定函数接口,缺乏灵活性和通用性。

为了能在不同的可视化平台上方便研究者评测算法,本文通过设计与平台无关的评测软件接口,针

收稿日期:2016-07-17.

基金项目:国家社会科学基金一般项目“基于大规模网络分析方法和内存计算技术的高校图书馆大数据应用模式与实证研究”(15BTQ016).

通讯联系人:何胜,博士,副教授,研究方向:数据挖掘、网络可视化. E-mail:hs@jsut.edu.cn

对上述有关算法性能评测的 3 个重要指标编写算法流程,在二维平面上,采用 Java 语言编写名为网络布局算法性能评测(Network Layout Algorithm Performance Test, NLAPT)的软件,为相关领域研究人员提供通用的评测网络布局算法性能的软件工具。

## 1 NLAPT 评测指标分析

NLAPT 软件工具主要从边-边交叉率、节点-边交叉率、相对边长度<sup>[13-15]</sup>3 个指标来评测网络布局算法的性能,在 NLAPT 中以程序函数方式体现,涵义简述如下。

### 1.1 边-边交叉率

边-边交叉率(ratio of edge-edge crossings)对应的函数名称为 `edgeEdgeCrossing`,是指布局中实际存在的边-边交叉数和所有节点的可能组合数比值,较低的边-边交叉率能有效降低视觉复杂度。

### 1.2 节点-边交叉率

节点-边交叉率(ratio of node-edge crossings)对应的函数名称为 `nodeEdgeCrossing`,是指布局中实际存在的节点-边交叉数除以全部可能节点-边的交叉数,即总边数和总节点数的乘积,在较低的节点-边交叉率绘图中,由于边在无节点之间的“贯穿”减少,从而在视觉上避免对节点间边连接的误解。

### 1.3 相对边长度

相对边长度(relative edge length)对应的函数名称为 `relateEdgeLength`,定义为所有边长度(一般采用节点间的欧几里德距离)的总和除以边数与布局空间的面积的乘积,反应了节点在布局空间的分布是否“平衡”,好的布局应该有较小的相对边长度。

## 2 NLAPT 接口设计

NLAPT 采用软件包形式引入到可视化软件平台,为了方便在不同平台上应用,评测函数的接口需要仔细设计。一般来说,节点的坐标准确定义了节点在二维平面上的位置,而节点之间的连接关系则确定了节点的拓扑连接关系,可用网络邻接矩阵表示。若将节点坐标和邻接矩阵这两者唯一确定下来,则网络布局也能被唯一确定。因此 NLAPT 评测函数的接口选用网络邻接矩阵和节点的位置坐标数组作为接口参数。设计方案如图 1 所示。

第一步,产生评测函数的接口,这里评测函数的接口是指用于评测函数的评测参数,从网络文本文件和软件平台中的网络布局图解析得出。其中由网络文本解析网络邻接矩阵,由网络布局图获取节点坐标值(以节点横坐标  $x$  和纵坐标  $y$  对应的数组表示),用于 3 个评测函数的统一的函数接口。第二步,计算评测函数值。其中通过统计边-边交叉数目、节点-边交叉数计算 `edgeEdgeCrossing` 和 `nodeEdgeCrossing`;依据节点坐标值,通过对所有边的欧几里德距离求和来计算相对边长度 `relateEdgeLength`。以下简述评测参数的设计思路。

### 2.1 网络邻接矩阵 $M$ 的设置

假设网络经某个算法布局及绘图如图 2(a)所示,图中标记了节点名称(标签)、对应坐标和节点间连接关系。该网络图也可以用如图 2(b)网络文本文件表示,其中每一行表示图 2(a)中的相应两个节点之间存在一条边连接。由网络文本文件可以创建网络邻接矩阵,简述如下:

该网络的邻接矩阵显示网络中所有节点的连接,如图 3 所示,若两个节点间有连接,方形矩阵中相应的行列交叉点上的值设置为 1,否则为 0。如节点  $A$  连接到节点  $B, C, E$ ,在矩阵  $M$  的第一行中  $(A, B), (A, C), (A, E)$  的位置设为 1,其他连接情况以此类推。邻接矩阵

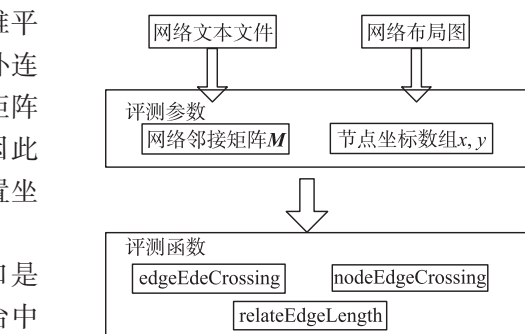


图1 NLAPT 软件接口设计

Fig.1 Design of NLAPT software interface

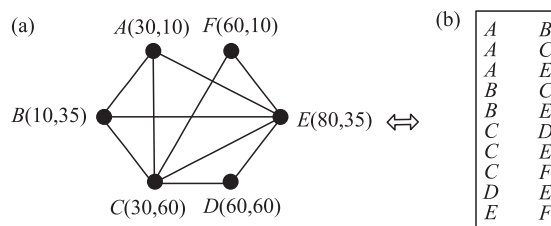


图2 一个例子网络布局图及文本形式的网络文件

Fig.2 An example network layout & its text pattern

阵  $M$  为对称矩阵,在本次设计中不考虑节点到自身的连接,因此邻接矩阵  $M$  的对角线元素都设置为 0.

特别需要注意的是,每个节点标签都分配一个自然数作为节点的序号(如  $A \leftrightarrow 0; B \leftrightarrow 1; C \leftrightarrow 2; D \leftrightarrow 3; E \leftrightarrow 4; F \leftrightarrow 5$ ),以方便邻接矩阵表示成二维数组  $M[][]$ ,其中  $M$  数组的下标(即  $[]$  中的数字)即为节点的序号,图 2(a)的网络对应数组值的设置如图 3 所示.

## 2.2 节点坐标数组的表示和获取方法

节点坐标包括横坐标和纵坐标共两组参数,采用一维数组表示.如图 4 所示,数组的下标为节点的序号,对应图 2(a)的网络,如  $x[0]=30, y[0]=10$  下标为 0,该序号对应  $A$  节点,因此  $(30, 10)$  即为节点  $A$  的坐标.同理,  $x[1]=10, y[1]=35; x[2]=30, y[2]=60$  表示节点  $B$  坐标  $(10, 35)$  和节点  $C$  坐标  $(30, 60)$ . 可视化平台一般都提供节点坐标的函数,方便用户获取坐标值.

## 3 NLAPT中评测函数算法

设二维平面上的图  $G=(V, E)$ ,  $V, E$  为节点和边的集合,节点数为  $n$ ,边数为  $m$ ,节点的坐标一旦确定,网络布局即确定,节点  $i$  坐标记为  $(x[i], y[i])$ ,其中节点数组  $x=(x_1, x_2, \dots, x_n), y=(y_1, y_2, \dots, y_n)$ . 依据网络邻接矩阵数组和节点坐标数组,设计测试函数的算法如下:

**算法 1** 边-边交叉率算法(edgeEdgeCrossing)(输入: $M, x, y$ )

- Step 1 初始化交叉  $E\text{Number}=0$
- Step 2 对所有  $M[i][j]=1$ (存在连接),创建边的二维数组  $E[m][2]$ ,  $E$  中存放节点对应的序号
- Step 3 对数组  $E$  中的每个边,依据其目标节点和源节点的坐标,计算边交叉数
  - 3.1 for  $i \in E$ ,依据对应源节点和目标节点序号,从  $x$  和  $y$  数组中取出  $i$  的横坐标和纵坐标
  - 3.2 for  $j \in E$ ,取出另一条边的对应节点的横坐标和纵坐标
  - 3.3 如果在边集合  $E$  中存在边交叉,  $E\text{Number}=E\text{Number}+1$
  - 3.4 end for
  - 3.5 end for
- Step 4 输出  $E\text{Number}/\binom{2}{n}$

算法 1 先基于邻接矩阵  $M$ ,确定存在边连接的节点对,并创建边数组  $E$ ,然后操作边数组  $E$ ,应用二重循环,逐一判断是否存在线段交叉,并累积统计,最后计算边-边交叉率.

**算法 2** 节点-边交叉率算法(nodeEdgeCrossing)(输入: $M, x, y$ )

- Step 1 初始化节点-交叉数  $N\text{Number}=0$ ;依据  $M$  产生节点数组  $N[n]$
- Step 2 对所有  $M[i][j]=1$ (存在连接),创建边的二维数组  $E[m][2]$
- Step 3 对节点数组  $N$  中的每个节点及数组  $E$  中的每个边,计算节点-边交叉数
  - 3.1 for  $i \in$  点集合  $N[n]$ ,取出源节点和目标节点的坐标
  - 3.2 for  $j \in$  边集合  $E$
  - 3.3 如果在点  $i$  在边  $j$  上,存在点  $i$  和边  $j$  的交叉,  $N\text{Number}=N\text{Number}+1$
  - 3.4 end for
  - 3.5 end for

		$A$	$B$	$C$	$D$	$E$	$F$	节点标签
		0	1	2	3	4	5	矩阵列序号
$A$	0	0	1	1	0	1	0	
$B$	1	1	0	1	0	1	0	
$C$	2	1	1	0	1	1	1	
$D$	3	0	0	1	0	1	0	
$E$	4	1	1	1	1	0	1	
$F$	5	0	0	1	0	1	0	

矩阵行序号

$M[0][1]=1 M[0][2]=1 M[0][4]=1$   
 $M[1][0]=1 M[1][2]=1 M[1][4]=1$   
 $M[2][0]=1 M[2][1]=1 M[2][3]=1 M[2][4]=1 M[2][5]=1$   
 $M[3][2]=1 M[3][4]=1$   
 $M[4][0]=1 M[4][1]=1 M[4][2]=1 M[4][3]=1 M[4][5]=1$   
 $M[5][2]=1 M[5][4]=1$

图3 邻接矩阵  $M$  的元素值设置

Fig.3 Design for value of adjacency matrix  $M$

$A: x[0]=30$	$y[0]=10$
$B: x[1]=10$	$y[1]=35$
$C: x[2]=30$	$y[2]=60$
$D: x[3]=60$	$y[3]=60$
$E: x[4]=80$	$y[4]=35$
$F: x[5]=60$	$y[5]=10$

图4 节点坐标的设计

Fig.4 Design for node coordinates

Step 4 输出  $N\text{Number}/(n*m)$

算法 2 需要依据邻接矩阵  $M$  产生一维节点数组,该数组的下标与邻接矩阵  $M$  中横向或纵向的下标一一对应,这样映射到坐标数组  $x$  和  $y$  上,从而方便确定节点的坐标值.然后对所有节点和边应用二重循环,逐一判断点  $i$  是否在边  $j$  上,累加统计后,计算点边交叉率.

**算法 3** 相对边长度算法(relatedEdgeLength)(输入:  $M, x, y$ )

Step 1 统计  $x, y$  中的坐标最大和最小值,产生布局区域边界  $s$  的范围,并计算布局区域面积,初始化边总长度  $\text{total}=0$

Step 2 对所有  $M[i][j]=1$ (存在连接),创建边的二维数组  $E[m][2]$

Step 3 对边数组中的所有的边,计算边的长度  $d_i$

3.1 for  $i \in$  边集合  $E$ ,取出对应源节点坐标  $(x_1, y_1)$  和目标节点坐标  $(x_2, y_2)$

3.2  $d_i = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$

3.3  $\text{total} = \text{total} + d_i$

3.4 end for

Step 4 输出  $\text{total}/(m*s)$

算法 3 是先对所有的边的欧几里德距离求和,然后除以总节点数和总的布局面积的乘积.

## 4 软件实现和测试结果

NLAPT 软件<sup>[16]</sup>采用 Java 语言编写,独立打包,能方便集成于各种不同的软件平台.

### 4.1 简单网络布局评测结果

容易看出,该网络布局的边边交叉数为 3,6 个节点所有可能组合数为 15,因此边边交叉率为 0.2;节点边交叉数为 0,因此节点边交叉率为 0;0 相对边长度按相应方法计算,对应图 2(a)的网络,工具软件运行的测试结果如图 5 所示.

### 4.2 网络布局算法布局性能评测结果

图 6 是采用 GLayout 可视化平台中集成的网络布局算法 grid layout<sup>[9,17]</sup>计算并自动绘制的酵母细胞周期调控网络(yeast cell cycle regulatory network)图,其中 grid layout 是一种将网络节点布局到网格上以避免节点之间相互重叠的算法.

应用 NLAPT 评测软件对该算法的性能评测结果如图 7 所示.

## 5 结语

本文为网络布局算法领域的研究者提供统一和便捷的测试软件.长期以来,研究者们不断提出新算法或改进原算法,并开发了诸多软件平台集成这些算法以展现布局结果,NLAPT 软件基于 3 个常用的网络可视化评测函数,给出定量的评测指标,为相关算法的研究者评测算法或调试算法参数提供了方便的工具.

本文 3 个评测指标偏重于网络可视化美学标准,其他的美学指标还包括:对称性(结构相同的子图尽量在视图界面中心平衡布局)、边平衡布局(同一节点的多条边尽量以该节点为中心平衡布局)、高度节点居中(高度节点应尽量布局在绘图中心)等.非美学标准的指标主要有网络的模块度量(拓扑上密切联系的节点形成的模块是否与图形界面显示的几何模块相一致),模块度量指标在分析生物网络功能、社交网

Ratio of edge-edge Crossings=0.2  
Ratio of edge-edge Crossings=0.0  
Relate edge length=0.7575546966969513

图 5 图 2 的网络布局评测结果

Fig.5 Result of network layout test for Fig.2

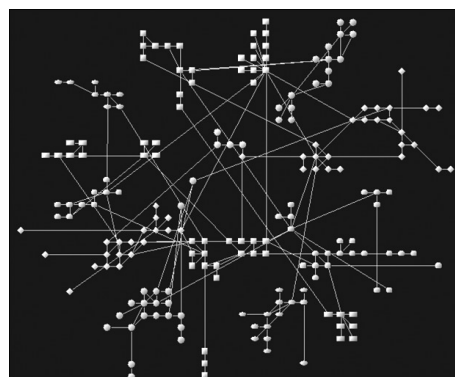


图 6 应用网络布局算法对酵母细胞周期调控网络布局

Fig.6 Layout of yeast cell cycle regulatory network by grid layout

Ratio of edge-edge Crossings=0.009 045 226 130 653 266  
Ratio of node-edge Crossings=0.001 234 567 901 234 567 9  
Relate edge length=0.091 234 300 534 324 05

图 7 grid layout 算法评测结果

Fig.7 Result of network layout test for grid layout algorithm



络的社团结构等方面具有重要意义<sup>[8]</sup>。

NLAPT 的 1.0 版本已经取得国家版权局的软件著作权登记证书,接下来的工作将结合不同评测需求以及各类指标,加入更多的评测函数,不断完善该软件。

### [参考文献](References)

- [1] JEONG H, TOMBOR B, ALBERT R, et al. The large-scale organization of metabolic networks[J]. Nature, 2000, 407(10): 651-654.
- [2] 唐杰, 陈文光. 面向大社交数据的深度分析与挖掘[J]. 科学通报, 2015, 60(5/6): 509-519.  
TANG J, CHEN W G. Deep analytics and mining for big social data[J]. Chin Sci Bull, 2015, 60(5/6): 509-519. (in Chinese)
- [3] 王元卓, 贾岩涛, 刘大伟, 等. 基于开放网络知识的信息检索与数据挖掘[J]. 计算机研究与发展, 2015, 52(2): 456-471.  
WANG Y Z, JIA Y T, LIU D W, et al. Open web knowledge aided information search and data mining[J]. Journal of computer research and development, 2015, 52(2): 456-471. (in Chinese)
- [4] DOGRUSOZ U, BELVIRANLI M E, DILEK A. A circular spring embedder layout algorithm[J]. IEEE transactions on visualization and computer graphics, 2013, 19(6): 953-966.
- [5] 水超, 陈涛, 李慧, 等. 基于力导向模型的网络图自动布局算法综述[J]. 计算机工程与科学, 2015, 37(3): 457-465.  
SHUI C, CHEN T, LI H, et al. Survey on automatic network layouts based on force-directed model[J]. Computer engineering & science, 2015, 37(3): 457-465. (in Chinese)
- [6] 吴鹏, 李思昆. 适于社会网络结构分析与可视化的布局算法[J]. 软件学报, 2011, 22(10): 2 467-2 475.  
WU P, LI S K. Layout algorithm suitable for structural analysis and visualization of social network[J]. Journal of software, 2011, 22(10): 2 467-2 475. (in Chinese)
- [7] GANSNER E R, HU Y, NORTH S. A maxent-stress model for graph layout[J]. IEEE transactions on visualization and computer graphics, 2013, 19(6): 927-940.
- [8] 孙扬, 蒋远翔, 赵翔, 等. 网络可视化研究综述[J]. 计算机科学, 2010, 37(2): 12-30.  
SUN Y, JIANG Y X, ZHAO X, et al. Survey on the research of network visualization[J]. Computer science, 2010, 37(2): 12-30. (in Chinese)
- [9] HE S, MEI J, SHI G, et al. LucidDraw: efficiently visualizing complex biochemical networks within MATLAB[J]. BMC bioinformatics, 2010, 11(1): 31-41.
- [10] SHANNON P, MARKIEL A, OZIER O, et al. Cytoscape: a software environment for integrated models of biomolecular interaction networks[J]. Genome research, 2003, 13(11): 2 498-2 504.
- [11] VLADIMIR B, ANDREJ M. Networks/pajek program for large network analysis[EB/OL]. [2016-05-24] <http://vlado.fmf.uni-lj.si/pub/networks/pajek/>.
- [12] HU Z, SNITKIN E S, DELISI C. VisANT: an integrative framework for networks in systems biology[J]. Brief bioinform, 2008, 9(4): 317-325.
- [13] INOUE K, SHIMOZONO S, YOSHIDA H, et al. Application of approximate pattern matching in two dimensional spaces to grid layout for biochemical network maps[J]. PLoS ONE, 2012, 7(6): e37739.
- [14] KOJIMA K, NAGASAKI M, MIYANO S. An efficient biological pathway layout algorithm combining grid-layout and spring embedder for complicated cellular location information[J]. BMC bioinformatics, 2010, 11(1): 335.
- [15] 刘大有, 金弟, 何东晓, 等. 复杂网络社区挖掘综述[J]. 计算机研究与发展, 2013, 50(10): 2 140-2 153.  
LIU D Y, JIN D, HE D X, et al. Community mining in complex networks[J]. Journal of computer research and development, 2013, 50(10): 2 140-2 153. (in Chinese)
- [16] He S. NLAPT[EB/OL]. [2016-08-06]. <http://jsjxy.jstu.edu.cn/Detail.aspx?DepartColumnId=14>.
- [17] LI W, KURATA H. A grid layout algorithm for automatic drawing of biochemical networks[J]. Bioinformatics, 2005, 21(9): 2 036-2 042.

[责任编辑: 严海琳]