Graphviz的布局使用手册

简体中文版

[dot,circo,fdp,neato,sfdp,twopi的布局手册页(Layout manual pages)]

名字

dot —用于绘制有向图的工具（filter：滤器）
neato —用于绘制无向图的工具
twopi—用于绘制径向布局的图的工具
circo—用于绘制环形布局的图的工具
fdp—用于绘制无向图的工具
sfdp—用于绘制大型无向图的工具
patchwork—方形树图工具
osage—基于阵列布局的工具

摘要

dot [选项] [文件]
neato [选项] [文件]
twopi [选项] [文件]
circo [选项] [文件]
fdp [选项] [文件]
sfdp[选项] [文件]
patchwork[选项] [文件]
osage [选项] [文件]

描述

这是一个用于绘图的程序集。实际上只有一个主程序，具体的布局算法是作为插件实现的。因此，他们很大程度上全部共享了相同的命令行选项。

dot 绘制有向图。他在有向无环图和其他的可以按照层次或者有一个自然的“流”进行绘制的图上工作的很好。

neato 使用“spring”模型和相关能量缩减(见 Kamada and Kawai, Information Processing Letters 31:1, April 1989)来绘制无向图。

twopi 使用径向布局(见 G. Wills, Symposium on Graph Drawing GD’97, September,1997)绘图。基本上，一个节点被选做中心并且放在原点，其余的节点被放在以原点为中心的一系列同心圆上。每一个圆圈间都有固定的径向距离。所有距离中心为1的节点都被放在第一个圆环上，所有与第一个圆环距离为1的节点都被放在第二个圆环上，以此类推。

circo 使用环形布局(见 Six and Tollis, GD ’99 and ALENEX ’99, and Kaufmannand Wiese, GD ’02.)绘图。该工具识别双连接的组件，并在圆上绘制组件的节点。然后使用递归径向算法布置块切割点树。通过在圆的周长上放置尽可能多的边，使圆内的边交叉最小化。特别地，如果该组件是外平面的，则该组件将具有平面布局。如果一个节点属于多个非平凡的双连接组件，布局会将该节点放在其中一个组件中。默认情况下，这是从根组件中搜索到的第一个非平凡组件。

fdp 使用“spring”模型绘制无向图。它依靠一种本着Fruchterman and Reingold (cf. Software-Practice & Experience 21(11), 1991, pp. 1129-1164)精神的力量导向的方式{a force-directed approach}。

sfdp 也使用上面描述的“spring”模型绘制无向图。但它使用一种多尺度方法在相当短的时间内产生大型图的布局。

patchwork 绘制方形树图。使用图的簇来指定树。

osage 使用它的簇结构绘图。 对于一个给定的簇，它的每个子簇都在内部布局。然后根据簇的pack和packmode属性重新定位子簇和剩余节点。

输出格式

Graphviz 在它的输出渲染上使用一个可扩展插件机制。因此你可以使用“dot -T：”并查看提示信息来知晓你安装的dot支持什么格式。同样，插件机制支持输出格式的多种实现。允许渲染器和格式化器的变化。查看一个具体输出格式的可用变体，例如使用“dot -Tpng：”若要强制使用特定变体，例如使用“dot -Tpng：gd”。

传统上，Graphviz支持下面的选项：
-Tdot （Dot格式来存储布局信息）
-Txdot （Dot格式来存储完整的布局信息） **-Tps** （postscript）
-Tpdf （pdf）
-Tsvg -Tsvgz （结构化向量图Structured Vector Graphics）
-Tfig （XFIG 图）
-Tpng （png bitmap 图）
-Tgif （gif bitmap 图）
-Tjpg -Tjpeg （jpeg bitmap 图）
-Tjson （xdot 信息被编码进入JSON）
-Timap （imagemap文件用于httpd服务器的每个节点或边有一个非空的**href**属性）
-Tcmapx （客户端侧的用于html和xhtml）
附加的不常见或者更多具体目的的输出格式可以被找到在： http://www.graphviz.org/content/output-formats

从格式附加“:”例如“-Tpng：”产生的错误消息中，可以找到支持给定输出格式的替代插件。列出的第一个插件总是默认的

-P开关可用于生成graphviz本地安装插件支持的所有输出变量的图表

图文件语言

这是图文件语言的概要，通常使用.gv作为后缀。作为图：

$$[strict](graph|digraph)名字\{语句列表\}$$

是顶层图。如果图是**strict**，那么同一对节点间不允许有多条边。如果是有向图那就使用**digraph**声明。那么边操作符必须是‘->’。如果是无向图那么使用**graph**声明，边操作符必须是“--”。

语句可以是：
名字=值;
node [名字=值];
edge [名字=值];
设置默认的图，节点，或边属性为*名字*到*值*。此后出现的任何子图、节点或边都继承了新的默认属性。

n0 [名称0=值0,名称1=值1,...];创建节点n0,并根据列表创建其属性。

$$n0--n1--\ldots--nn[属性名称0=属性值0,名称1=值1,\ldots];$$

创建边，并按照列表设置边属性，并按需要创建节点。

$$[subgraph名称]\{语句列表\}$$

创建子图，子图可以代替上述语句中的n0...nn来创建边。$[subgraphname]$是可选的,如果缺失的话,将会赋予一个内部名.

本语言接受两种C风格的注释:/**/和//

属性名和值是普通的(C风格)的字符串.下一节将描述控制图布局的属性.

一个更完整的语言描述可以在 http://www.graphviz.org/content/dot-language 找到.

图,节点和边属性

Graphviz使用*名称=值*属性,附加到图,子图,节点和边上,来定制布局和渲染.我们在后文列出了比较重要的属性,完整的列表在 http://www.graphviz.org/content/attrs

节点,边,簇和图共有的属性

1. **href=**url图像映射文件的默认url;在Postscript中,所有相关URL的基本URL,按照Acrobat Distiller 3.0及更高版本识别
2. **URL=**url (URL与href同义)
3. **fontcolor=**colorvalue 设置文本颜色标签

- colorvalue可以是"h,s,v"(hue色调,saturation饱和度,brightness亮度)介于0-1之间的浮点数.或者是X11颜色名像white,black,red,green,blue,yellow,magenta,cyan.或者是"#rrggbbb"(red红,green绿,blue蓝,每组都是两个16进制字符)值,更多的细节见http://www.graphviz.org/content/attrs#kcolor和http://www.graphviz.org/content/color-names for further details

4. **fontsize**=n 设置标签类型大小为n点(point像素?)
5. **fontname=**name 设置标签的字体家族名
6. **label=text** text中可能包含转义的换行符,\n,\l,\r将向中心,向左,向右调整行.字符串'\G'将被图形名替换,对于节点标签,字符串'\N'将被节点名称替换;对于边,如果子串'\T'在label中它将被尾节点名代替.如果字串'\H'在label中,它将被头节点名代替.如果子串'\E'在label中,它将被 尾节点名->头节点名或者无向图中的:尾节点名--头节点名 代替.
7. 对于构造复杂的节点内容,Graphviz也支持特殊的类html标签.全部相关描述在http://www.graphviz.org/content/node-shapes#html
8. 若节点有**shape=record**,标签可能包含由{ | }分隔的递归框列表.标签的端口标识由<>隔开

图属性

1. **size="x,y"** 指定绘图的最大边界框(英寸)
2. **ratio=f** 设置纵横比为f(可能是一个浮点数或者关键字fill,press,auto之一)
3. **layout=engine** 制定首选布局引擎(dot,neato,fdp,等等),覆盖命令的基名或-K命令行选项的默认值
4. **margin=f** 设置页边距(包括在页面大小中)
5. **ordering=out**根据文件序列约束子图中的外边缘顺序
6. **rotate=90** 设置景观模式(orientation=land是向后兼容,但过时了)
7. c**enter=n** 非零值将绘图居中显示在页面上
8. **color=colorvalue** 设置前景色(bgcolor设置背景色)
9. **overlap**=mode 这指定了当任何节点重叠时，应该执行什么算法
   - 如果mode是**false**,程序使用Prism算法来调整节点消除重叠(破坏对称但更紧凑)
   - 如果mode是**scale**,布局被均匀地放大，保留节点大小，直到节点不再重叠。(同时保持对称和结构)
   - 如果mode是**true**(默认),没有重新定位.
   - dot永远产生没有节点重叠的布局,该属性只针对其他布局
10. **stylesheet=**file.css 在-Tsvg和-Tsvgz输出中包括对样式表的引用,其他的输出格式则忽略该选项
11. **splines**如果设置为true,边将被画为 楔形,如果设置为polyline边将被画为折线,如果设置为ortho,边将被画为正交多段线,在所有情况下节点都不能重叠,如果splines=false或者splines=line,边将被画为线段,dor默认的是true,其他布局默认是false.
12. dot特有的属性:
    - **nodesep**=f 设置节点之间的最小距离
    - **ranksep**=f 设置层间的最小距离
    - **rankdir**=LR|RL|BT 要求一个左到右,右到左,底到上的绘制
    - **rank**=same (或者min或者max)在子图中，约束其节点的排名分配。如果子图的名称具有前缀cluster，则其节点将绘制在布局的不同矩形中。簇可以嵌套
13. neato特有的属性:
    - **mode**=val.布局中能量最小化的算法,neato默认使用应力优化(stress majorization),如果mode=KK,它使用梯度下降中的一个版本.
    - **model**=val. neato模型计算所有顶点对之间所需的距离。默认情况下,使用最短路径的长度.
      - 如果model设置为circuit,则使用电路电阻模型.
      - 如果model设置为subset,则使用其中边长度是恰好与边的一个顶点相邻的节点数的模型
    - **start**=val,请求随机初始位置和随机数生成器种子。如果val不是整数，则使用进程ID或当前时间作为种子
    - **epsilon**=n,设置求解程序的中止,默认是0.1
14. twopi特有的属性:
    - root=ctr,这指出用作布局中心的叶子节点,如果没有制定,将会选择一个最远的叶子节点, 其中叶节点阶数为1.如果没有叶节点存在,那就任意选一个节点作为中心
    - ranksep=val,指定环序列之间的径向距离（英寸）。默认值为0.75。
15. circo特有的属性:
    - root=nodename,指定根块中出现的节点的名称,如果图形断开连接，则根节点属性可用于指定其他根块。
    - mindist=value,设置所有节点之间的最小间距。如果未指定，则circo使用默认值值为1.0。
16. fdp特有属性:
    - K=cal,设置布局中的默认理想节点分隔
    - maxiter=val,设置用于布局图表的最大迭代次数
    - start=val,调整没有指定位置的节点的随机初始位置。如果val是一个整数，它用作随机数生成器的种子。如果val不是整数，则生成一个随机系统整数（如进程ID或当前时间）用作种子

节点属性

1. **height=d or width=d** 设置最小的高度和宽度.添加**fixedsize=true**强制使其为实际大小(忽略文本标签)
2. **shape=**builtin_polygon record epsf
   - builtin_polygon可以是plaintext, ellipse, oval, circle, egg, triangle, box, diamond,trapezium, parallelogram, house, hexagon, octagon, note, tab, box3d, or component
   - 多边形由以下节点属性定义或修改：规则(regular)、外围(peripheries)、边(sides)、方向(orientation)、扭曲(distortion)和倾斜(skew)
   - epsf使用节点的shape file属性作为要为节点形状自动加载的外部epsf文件的路径名。
   - 完整的节点形状描述见:http://www.graphviz.org/content/node-shapes
3. **color**=colorvalue 设置轮廓颜色,如果style=filled且未指定fillcolor则为默认填充色
4. **fillcolor=**colorvalue 当style=filled时设置填充色.如果没有指定,当style=filled时,填充色默认与轮廓色相同.
5. **style**=filled solid dashed dotted bold invis
6. **xlabel**="文本" 指定将放置在节点附近但在节点外部的标签。普通标签字符串放置在节点形状中
7. **target=**"target" 是客户端图像映射和SVG的目标字符串，当节点具有URL时有效.目标字符串用于确定用于URL的浏览器窗口 . 将其设置为"_graphviz"如果"_graphviz"不存在，它将打开一个新窗口；如果存在，它将重新使用。如果目标字符串为空，则输出中不包含默认的目标属性.子字符串'\n'和'\g'的替换方式与节点标签属性的替换方式相同。此外，子字符串'\l'由节点标签字符串替换。
8. **tooltip**="文本"是客户端图像映射和SVG的工具提示字符串，在节点具有URL时有效。工具提示字符串默认与标签字符串相同，但此属性允许不带标签的节点仍然有工具提示，从而允许更密集的图形。子字符串'\n'和'\g'的替换方式与节点标签属性的替换方式相同。此外，子字符串'\l'由节点标签字符串替换。

以下属性仅适用于多边形节点

- **regular**=n 如果n为非零，则多边形成为规则多边形，即围绕x和y轴对称，否则多边形将具有标签的纵横比。 不规则的*builtin_polygons*是通过此属性成为规则的。*builtin_polygons*是规则的则无效.(即它们不能不对称)
- **peripheries**=n 设置围绕多边形绘制的外围线数。此值取代*builtin_polygons*的外围线数
- **sides=**n 设置多边形边的数量,n<3的结果是椭圆.*builtin_polygons*忽略此属性.
- **orientation**=f 从垂直方向逆时针设置多边形第一个顶点的方向，单位为度。f可以是一个浮点数,标签的方向不受此属性影响.该属性被添加到*builtin_polygons*的初始化方向.
- **distortion=**f 设置多边形顶部的加宽和底部的缩小量（相对于其方向）,建议是介于-1到1之间的浮点值.*builtin_polygons*忽略此属性.
- **skew**=f 设置多边形顶部的右位移量和底部的左位移量（相对于其方向）.建议是介于-1到1之间的浮点值.*builtin_polygons*忽略此属性.

(circo特有属性)

root=true/false 这指定包含给定节点的块在布局中被视为生成树的根。

(neato和fdp特有属性)

pin=val 节点将保持在其初始位置

边属性

1. **weight=**val val是边的权重(cost).
   - 对于dot,weight必须是非负整数.大于1的值会缩短边；对节点排序时忽略权重为0的平边
   - 对于twopi,权重为0将导致在构建底层生成树时忽略边缘。
   - 对于neato和fdp,较重的权重将更加注重算法，使边缘长度接近边缘的len属性指定的长度。
2. **style**=solid dashed dotted bold invis
3. **color=**颜色值,设置边的线条颜色
4. **color=**颜色值列表,由“：”分隔的颜色值列表创建平行边，每种颜色对应一条边。
5. **dir=** forward,back,both,none 控制箭头方向
6. **tailclip,headclip=false** 禁用终结点形状剪辑。
7. **target=**"文本" 是客户端图像映射和SVG的目标字符串，在边具有URL时有效。如果目标字符串为空，为默认值，则输出中不包含任何目标属性。子字符串'\t'、'\h'、'\e'和'\g'的替换方式与边标签属性的替换方式相同。此外，用边标签字符串替换子字符串'\l'。

8. **tooltip=**"文本" 当边具有URL时，客户端图像映射的工具提示字符串是否有效。工具提示字符串默认与边标签字符串相同。子字符串'\t'、'\h'、'\e'和'\g'的替换方式与边标签属性的替换方式相同。此外，用边标签字符串替换子字符串'\l'
9. **arrowhead,arrowtail=none, normal, inv, dot, odot, invdot, invodot, tee, empty, inv empty, open, halfopen, diamond, odiamond, box, obox, crow.**分别指定边与头部或尾部节点接触处出现的沟纹(glyph)的形状。请注意，这只指定形状。dir属性决定是否绘制glyph。
10. **arrowsize**=val 指定箭头大小的乘法比例因子.(inv_length=6,inv_width=7,dot_radius=2)
11. **headlabel,taillabel=**文本, 用于出现在边的头节点和尾节点附近的标签。
    - labelfontcolor,labelfontname, labelfontsize用于头和尾标签.
    - 子字符串'\t'、'\h'、'\e'和'\g'的替换方式与边标签属性的替换方式相同。
    - 此外，用边标签字符串替换子字符串'\l'
12. **headhref=**"url" 设置imagemap、postscript和svg文件中头端口的URL。子字符串'\t'、'\h'、'\e'和'\g'的替换方式与边标签属性的替换方式相同。此外，用边标签字符串替换子字符串'\l'
13. **headURL=**"url" (与headhref同义)
14. **headtarget=**"headtarget" 是客户端图像映射和SVG的目标字符串，在边缘头有URL时有效。HeadTarget字符串用于确定浏览器的哪个窗口用于URL。如果headtarget字符串为空，则默认值为headtarget，默认值与边缘的目标值相同。子字符串'\t'、'\h'、'\e'和'\g'的替换方式与边缘标签属性的替换方式相同。此外，用边缘标签字符串替换子字符串'\l'
15. **headtooltip=**"tooltip" 当头端口具有URL时，客户端图像映射的工具提示字符串是否有效。工具提示字符串默认与headlabel字符串相同。子字符串'\t'、'\h'和'\e'的替换方式与边标签属性的替换方式相同。此外，用边标签字符串替换子字符串'\l'。
16. **tailhref=**"url" 在imagemap、postscript和svg文件中设置尾部端口的URL。子字符串'\t'、'\h'、'\e'和'\g'的替换方式与边标签属性的替换方式相同。此外，用边标签字符串替换子字符串'\l'
17. **tailURL**="url" 与tailhref同义
18. **tailtarget=**"tailtarget" 是客户端图像映射和SVG的目标字符串，在边尾部有URL时有效。tailTarget字符串用于确定浏览器的哪个窗口用于URL。如果tailTarget字符串为空，则默认值为tailTarget，默认值与边的目标值相同。子字符串'\t'、'\h'、'\e'和'\g'的替换方式与边标签属性的替换方式相同。此外，用边标签字符串替换子字符串'\l'
19. **tailtooltip=**"tooltip" 当尾部端口具有URL时，客户端图像映射的工具提示字符串是否有效。工具提示字符串默认与taillabel字符串相同。子字符串'\t'、'\h'、'\e'和'\g'的替换方式与边标签属性的替换方式相同。此外，用边标签字符串替换子字符串'\l'
20. **labedistance和labelangle**(in degrees CCW)指定头尾标签的位置
21. **decorate**从边到标签划线
22. **samehead,sametail**使用平均着陆点，将具有相同值的边对准相同的端口。

（dot特有属性）

- **constraint**=false 造成边在次序分配时被忽略
- **minlen=**n 其中n是适用于边长度的整数因子(标准边的次序，或平边的最小节点分隔)
- **xlabel=**"text" 节点中的边标签被视为特殊类型的节点，在节点布局期间为其分配空间。这有时会使边缘布线变形。如果使用XLabel，则在定位所有节点和边之后放置标签。反过来，这可能意味着标签之间有一些重叠。

（neato和fdp特有属性）

len=f 设置最优的边长度.默认是1.0

命令行选项

-G 设置默认的图属性
-N 设置默认的节点属性
-E 设置默认的边属性:例子:**-Gsize="7,8" –Nshape=box –Efontsize=8**
-Ifile 载入定制的PostScript库文件.通常其定义了定制的形状和风格.如果**-I**本身已给定,忽略标准库.
-Tlang 按照上面描述的设置输出语言
-n[1|2]如果设置了,neato假定节点已经定位，并且所有节点都有一个pos属性来给出位置。然后，根据“重叠”属性的值，执行可选调整以删除节点节点重叠，根据样条曲线属性的值计算边布局，并以适当的格式发送图形。如果提供num，则会发生以下操作：*num*=1等价于*-n num>1* 使用指定的节点位置，不进行调整以删除节点节点重叠，并使用已经由pos属性指定的任何边布局。neato为没有pos属性的任何边计算边布局。通常，边布局由spline属性引导。
-Klayout 重写命令名所暗示的默认布局引擎
-O 基于输入文件名和*-T*格式自动生成输出文件名 **-P** 生成一个当前可用插件图. **-v**(verbose) 打印用于调试的各种信息
-c 配置插件 **-m** 内存检测(观察无顶部生长{溢出?},完成后杀死) **-q**level 设置消息抑制级别。默认值为1。 **-s**fscale 按fscale输入比例，默认值为72 **-y** 在输出中反转Y坐标 **-V**打印版本信息并退出 **-?**打印用法并退出 一个完整的命令行选项描述在http://www.graphviz.org/content/command-line-invocation

例子

```
digraph test123 {
    a -> b -> c;
    a -> {x y};
    b [shape=box];
    c [label="hello\ nworld",color=blue,fontsize=24,
        fontname="Palatino-Italic",fontcolor=red,style=filled];
    a -> z [label="hi", weight=100];
    x -> z [label="multi-line\ nlabel"];
    edge [style=dashed,color=red];
    b -> x;
    {rank=same; b x}
}
graph test123 {
    a -- b -- c;
    a -- {x y};
    x -- c [w=10.0];
    x -- y [w=5.0,len=3];
}
```

告诫,附加说明

边样条线可能无意中重叠。
平边标签稍有破损。簇间边缘标签完全破损。
由于采用了无约束优化，节点框可能会重叠或接触不相关的边。所有现有的弹簧嵌入件似乎都有这个限制。
显然，合理尝试固定节点或调整边缘长度和权重会导致不稳定。

作者

Stephen C. North north@research.att.com
Emden R. Gansner erg@graphviz.org
John C. Ellson ellson@research.att.com
Yifan Hu yifanhu@yahoo.com
bitmap 驱动 (PNG, GIF 等) 由Thomas Boutell, http://www.boutell.com/gd 创作

请参阅(see also)

此手册页仅包含与graphviz布局程序相关的少量信息。
最完整的信息可以在http://www.graphviz.org/documentation.php上找到，尤其是在线参考页面上。
这些文档中的大多数也可在源和二进制包发行中找到。

dotty(1)
tcldot(n)
xcolors(1)
libcgraph(3)

E. R. Gansner, S. C. North, K. P. Vo, "DAG - A Program to Draw Directed Graphs", Software - Practice and Experience 17(1), 1988, pp. 1047-1062.
E. R. Gansner, E. Koutsofios, S. C. North, K. P. Vo, "A Technique for Drawing Directed Graphs," IEEE Trans. on Soft. Eng. 19(3), 1993, pp. 214-230.
S. North and E. Koutsofios, "Applications of graph visualization", Graphics Interface 94, pp. 234-245.
E. R. Gansner and E. Koutsofios and S. C. North, "Drawing Graphs with dot," Available at http://www.graphviz.org/pdf/dotguide.pdf.
S. C. North, "NEATO User's Manual". Available http://www.graphviz.org/pdf/neatoguide.pdf.
E. R. Gansner and Y. Hu, "Efficient, Proximity-Preserving Node Overlap Removal", J. Graph Algorithms Appl., 14(1) pp. 53-74, 2010.

翻译: Jonson-Sun
时间: 2019年2月7日,中午
原因: dot用起来真方便!
英文原版地址: https://graphviz.gitlab.io/_pages/pdf/dot.1.pdf