

使用 PSTricks 创建图论中的树和图*

Manjusha S.Joshi

Bhaskaracharya Institute in Mathematics,Pune,India

2011 年 11 月 24 日

摘要

对研究者、老师们来说，绘制图论（数学）中的树和图是一项要求。这其中包含了环、弧、节点、边的权重的绘制。本文旨在首先通过记住两个命令即 `pstree` 和 `psmatrix` 以及他们的最有用的选项，借助它们，读者可以画出精致的图形、环；可以对节点进行标记；在边上增加权重。在完成图形之后，读者可以将这些图加入到 T_EX 文件中。这样，图论中绘图的应用知识就可以掌握了。

目录

1 初步准备	2
2 psmatrix	2
2.1 边的类型：线，弧和环	4
2.2 边的方向和权	5
2.3 补充的 PSTricks	7
3 pstree	8
4 插图	11
5 最常见的错误和一些建议	11
A 更多的图	12

*译者：ddswhu,希望大家不吝指正,ddswhu@gmail.com

1 初步准备

预备知识: L^AT_EX 的基本知识

预 期: 在阅读完本文之后, PSTricks 新手可以创建他/她自己的图形文件
并将能够将其插入到他/她的 T_EX 文档中。

系统要求: 有 PSTricks 宏包的 L^AT_EX 系统

网 站: 你可以从以下网站获得需要的宏包以及信息

<http://tug.org/PSTricks/>

导言区需要加入的宏包: pstricks,pst-node,pst-tree

2 psmatrix

在 psmatrix 命令中涉及到“矩阵”这个词, 我们首先来看一下矩阵。我们写矩阵 $A_{3 \times 4}$ 意味着这个矩阵有 3 行 4 列。这被称为矩阵的维度。

$$\begin{matrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{matrix}$$

矩阵中的每一个元素可以被“元素位于哪行和哪列”来确定。比如, a_{23} 指的是第二行第三列的元素, 顺序是先行数后列数。

现在来看一下 `\psmatrix` 环境

```
$  
\psmatrix  
  
\endpsmatrix  
$
```

虽然这个环境不是以 `\begin` 开始, 但是它以 `\end` 命令结束。注意这个环境与其他 T_EX 环境的不同之处。

要使用这些特殊的 PSTricks 命令, 需要加入 `\usepackage{pstricks,pst-node,pst-tree}`。这里有个包含了在导言区必须的最少命令的代码。

```
\documentclass{article}  
\usepackage{graphics,graphicx}  
\usepackage{pstricks,pst-node,pst-tree}  
\begin{document}
```

```

$
\psmatrix
u&v&z\\
x&y
\endpsmatrix
$
\end{document}

```

这个代码的输出像这样：

$$\begin{array}{ccc}
 u & v & z \\
 \\
 x & y &
 \end{array}$$

注意，这里垂直距离和水平距离都是默认的。我们可以通过加入 `psmatrix` 命令的可选参数 `\psmatrix[colsep=2.5cm,rowsep=1.5cm]` 来控制它们，保持两列之间的距为 2.5cm, 两行之间的距离为 1.5cm。

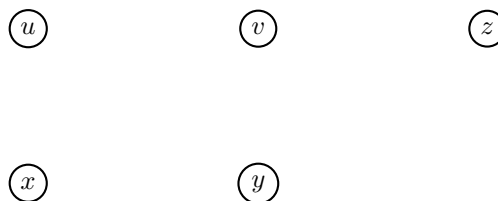
而且，我们经常喜欢用小圆圈包围字母来表示这些点,比如 \textcircled{a} 。这可以通过 `\psmatrix[mnode=circle]` 来实现。这个参数表示选用圆形节点。所以，加入这两个命令之后，我们的代码将会是

```

\documentclass{article}
\usepackage{graphics,graphicx}
\usepackage{pstricks,pst-node,pst-tree}
\begin{document}
$
\psmatrix[colsep=2.5cm,rowsep=1.5cm,mnode=circle]
&u&v&z\\
3
&x&y
\endpsmatrix
$
\end{document}

```

这个代码的输出像这样：



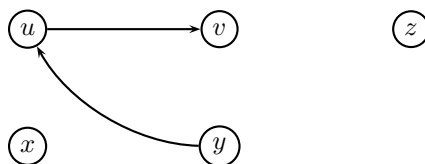
这样的输出可以用来绘制图的顶点。

2.1 边的类型：线，弧和环

现在我们想用边来连结顶点。边的类型有：线、弧或者环。为了连结顶点，需要记住矩阵的方式才能对元素定位。这种方法会延 续到后面的内容”引用一个元素”。

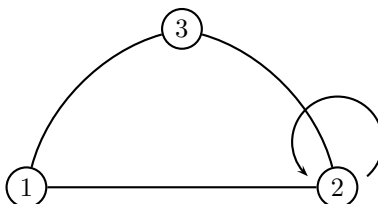
现在我们想要连结上述图中的元素 u 和 v 。 u 位于哪里？它位于第一行第一列，所以用 $(1,1)$ 来引用它。现在因为 v 位于第一行第二列，引用它用 $(1,2)$ ；记住顺序是 (行,列)。最后，使用 `\ncline{1,1}{1,2}` 命令画出位于两个节点之间的直线。很多时候，只是这条线是不够的，我们也要求表示出方向。为此，加上 `{<->}`，表示从哪点到哪个点。如果方向是 $v \rightarrow u$ ，那么需要改为 `{1,2}{1,1}`。

```
\documentclass{article}
\usepackage{graphics,graphicx}
\usepackage{pstricks,pst-node,pst-tree}
\begin{document}
$
\psmatrix[colsep=2cm,rowsep=1cm,mnode=circle]
u&v&z\\
x&y
\ncline{->}{1,1}{1,2}
\ncarc[arcangle=-30]{<-}{1,1}{2,2}
\endpsmatrix
$
\end{document}
```



`\narc[arcangle=-30]{<-}{1,1}{2,2}` 这个命令在绘制第一行第一列元素与第二行第二列元素之间的圆形的边的时候很有用。

为了获得一个节点的环，使用 `\nccircle{->}{1,1}{.5cm}` 这个命令。如果你想表示出方向，你可以在 `{->}` 之后加上节点（用它的行、列引用），然后再加上半径。



```
$
\psmatrix[colsep=1.5cm,rowsep=1.5cm,mnode=circle]
&3\\
1&&2
\ncline{2,1}{2,3}
\narc[arcangle=-30 ]{2,3}{1,2}
\narc[arcangle=-30]{1,2}{2,1}
\nccircle[nodesep=4pt]{->}{2,3}{.6cm}
\endpsmatrix
$
```

2.2 边的方向和权

方向可以用选项 `{->}{<-}{<->}` 来显示。所有的这些箭头都是在边的端点处显示。这可以用 2.3 节讨论的宏包 `pstricks-add` 解决。并且，箭头的大小是固定的，但是，如果你想增加它，参阅 2.3 节。

箭头和节点之间的距离默认是 0。如果需要将其分离开，可以使用选项 `[nodesep=1pt]`。

你可以将权加到边上。`\ncline{1,1}{1,2}^{\textit{1}}` 将可以显示权 ‘1’ 在连结节点 u 和 v 的边的上方。这些选项可供选择：

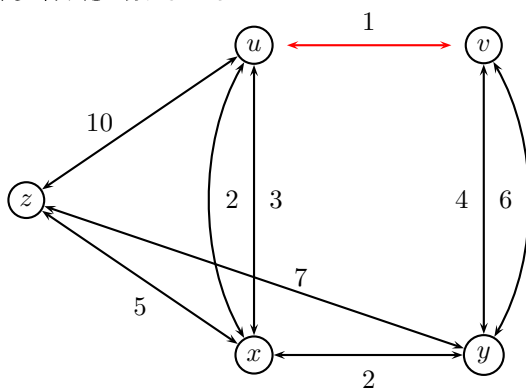
- < 边的左边
- > 边的右边
- ^ 边的上面
- _ 边的下面

```

\documentclass[a4paper,12pt]{article}
\usepackage{pstricks,pst-node,pst-tree}
\begin{document}
$
\psmatrix[colsep=2.5cm,rowsep=1.5cm,mnode=circle]
&u&v\\
z\\
&x&y
\ncline[nodesep=5pt]{<->}{1,2}{1,3}^{\{1\}}
\ncline{<->}{3,2}{1,2}>\{3\}
\ncline{<->}{3,2}{3,3}_\{2\}
\ncarc[arcangle=-30]{<->}{3,3}{1,3}<\{6\}
\ncarc[arcangle=-30]{<->}{1,2}{3,2}>\{2\}
\ncline{<->}{2,1}{1,2}<\{10\}
\ncline{<->}{2,1}{3,2}_\{5\}
\ncline{<->}{3,3}{2,1}>\{7\}
\ncline{<->}{1,3}{3,3}<\{4\}
\endpsmatrix
$
\end{document}

```

使用 `\ncarc` 选项，可以定义‘弧角’的大小。如果是逆时针方向，则角度位于 0 到 180 之间。如果是顺时针方向，则角度在 0 与 -180 之间。如果你不需要箭头，你可以省去参数 `{<->}`。



注意 u 和 v 之间的红线，它与这两个节点分离开了。这就是节点分离的效果。

2.3 补充的 PSTricks

在上述图形中，观察到箭头很小而且在边的末端。有时，我们可能需要箭头位于边的中间而不是线的末端。并且，大的箭头可以突出边的方向。

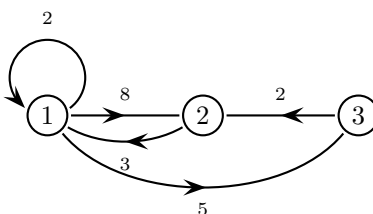
使用 `pstricks-add` 宏包，我们可以获得在边中间的箭头而不是在边的端点处。并且，可以增加箭头的大小。

导言区应该加入的宏包: `pstricks-add`

确定自己系统里有这个宏包，否则，加入它将会导致错误的出现。从 PSTricks 网站下载这个宏包，之后把它加到你的工作目录下，检查你的代码没问题。然后将其保存在一个合适的目录下。

```
\documentclass[a4paper,12pt]{article}
\usepackage{pstricks,pst-node,pst-tree,pstricks-add}
\begin{document}
\pagestyle{empty}
$
\psmatrix[colsep=1.5cm,rowsep=1.5cm,mnode=circle]
1&2&3
\psset{arrowscale=2}
\psset{ArrowInside=->,nodesep=1pt}
\everypsbox{\scriptstyle}
\ncline{1,3}{1,2}^{2}
\ncline{1,1}{1,2}^{8}
\ncarc[arcangle=30]{1,2}{1,1}_{3}
\ncircle{->}{1,1}{.5cm}^{2}
\ncarc[arcangle=-50]{1,1}{1,3}_{5}
\endpsmatrix
$
\end{document}
```

使用 `PSTricks-add` 之后，我们有选项 `arrowscale=2`。默认 `arrowscale=1`。当我们想改变它的时候，我们可以按照我们的要求对其赋值像 2、3。一般来说，`arrowscale=2` 就足够了。`pstricks-add` 中一个更有趣的命令是 `ArrowInside=->`。这个命令把箭头设置为边的里面而不是端点。这种方式它们更加直观。

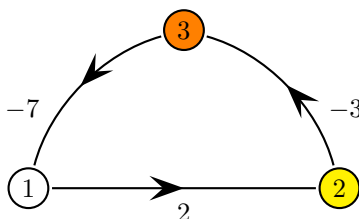


```

$
\psmatrix[colsep=1.5cm,rowsep=1.5cm,mnode=circle]
&[fillstyle=solid,fillcolor=orange]3\\
1&&[fillstyle=solid,fillcolor=yellow]2
\psset{arrowscale=3}
\psset{ArrowInside=->,nodesep=1pt}
\ncline{2,1}{2,3}_{2}
\ncarc[arcangle=-30 ]{2,3}{1,2}>{-3}
\ncarc[arcangle=-30]{1,2}{2,1}<{-7}
\endpsmatrix
$

```

观察到节点的更多选项 `[fillstyle=solid,fillcolor=orange]`，使这个节点填充橙色。



3 pstree

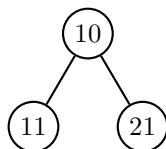
在导言区再加入一个宏包 `\pstree`。它具有树形的结构：一个根部和许多分支。我们可以决定分支的高度。树根据它拥有的分支的数目和它的健康程度即它分支的大小来扩展。

我们的树最重要的命令 `\pstree` 开始。它的第一个参数是树的根部，如果有分支的话，第二个参数就是分支。和真实的树不同的是，它向下生长。树的分支在一对大括号即 `{}` 内。每一个分支可以看做是新的树。但是它应该在大括号里面，直到整个树的结束。

分支的高度称为层级 (levels)。它可以由 `pstree` 的可选参数 `\pstree[levelsep=35pt]` 来设置。它具有整体效果。你可以通过 `\pstree[thislevelsep=20pt]` 来控制局部层级的高度。它只有有限的效果，仅限于特定的树。

使用 `\Tcircle`, 可以在 `pstree` 中获得圆形的节点。在这对大括号的 的参数是你想在节点打印出来的内容。

```
\documentclass{article}
\usepackage{graphics,graphicx}
\usepackage{pstricks,pst-node,pst-tree}
\begin{document}
\pstree[levelsep=35pt]{\Tcircle{10}}
{
\Tcircle{11} \Tcircle{21}
}
\end{document}
```



观察:

1. `pstree` 不需要 `$`, 但是如果需要, 你可以使用 `$ $`。
2. `pstree` 默认以线段连结节点即端点。
3. `levelsep` 控制分支的高度。

你可以在两节点之间得到虚线、点线或者“无线”。节点也有不同的类型, 比如三角形、椭圆形等。使用命令 `\Toval` 可以得到椭圆形, 使用 `\Tdia{abc}` 可以得到菱形的节点。为了得到三角形节点, 使用 `\Ttri{abc}`。注意, 大写 **T** 之后还有一个小写 **t**。有人或许会认为三角形 (triangle) 中含有 ‘tri’, 所以 ‘Tri’ 就行而忽略另外的 ‘t’。不! **T** 是为了使各种类型节点能够作为树的节点使用的。因此, **T** 是必须的。

`\Tr{10}` 将产生一个在 10 周围没有任何形状 of 节点。以下代码说明了树的各种节点的法。

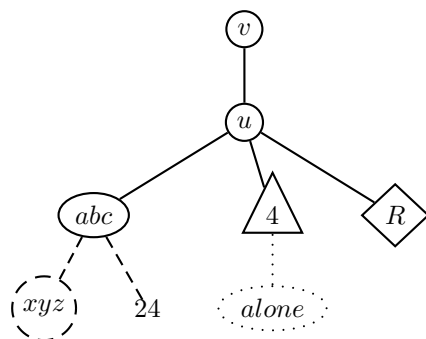
```
$
\pstree[levelsep=35pt]{\Tcircle{v}}{
\pstree{
\Tcircle{u}}{
\psset{linestyle=solid}\pstree{\Toval{abc}}{
```

```

\psset{linestyle=dashed}\Tcircle{xyz} \Tr{24}
}
\pstree{\Ttri{4}}{\psset{linestyle=dotted}\Toval{alone} }\Tdia{R}
}}
$

```

在这个代码中,使用了 `\psset` 命令。它允许我们设置和线型、宽度、箭头等相关的东西。线型总共有四种,无、虚线、点线、实线。对于线的粗细的设置,使用 `linewidth=1.5pt`。



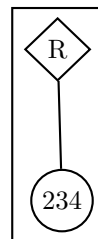
注意到,无论节点的线型是什么,连结根部和节点之间的边的线型和其一致。如果我们设置线型为无,节点类型为圆形将会发生什么呢?让我们来检验一下。

首先绘出线型为实线的树然后绘制线型为无的树:

```

\pstree{\Tdia{R}}{\psset{linestyle=solid}\Tcircle{234}}

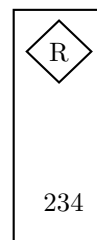
```



```

\pstree{\Tdia{R}}{\psset{linestyle=none}\Tcircle{234}}

```



只要我们声明了 `linestyle=none`，它将移除 R 和 234 之间的边，并将在 234 周围绘制一个看不到的圆。

4 插图

一种插图的方式是在你运行的文本中写代码。在你的 .tex 文件按往常的方式编译之后，`\pstree{\Tdia{R}}{\psset{linestyle=solid}\Tcircle{234}}` 也会被自动编译，并且输出在相应位置。是的！你可以不用声明 `\includegraphics{}` 而直接把代码加进去。

并且，你可以把这个代码插入：

```
\begin{figure}
\pstree{\Ttri{ABC}}{}
\end{figure}
```

将生成



另外一种方法是将这个代码单独写在一个文件名比如说 figno.tex 的文件中，然后通过编译得到 .ps 文件。通过一些软件比如 Gimp 或者 Ghostview 可以将 .ps 文件得到 .eps 文件。然后像插入其他 .eps 文件一样：`\includegraphics{figno.eps}` 插入 .eps 文件。

如果你想把你的 \TeX 文件和图发送给其他人，那么这个方法将会有优势。将它（指图）以 eps 形式发送是一种好的想法，接收者可以调整图形的大小和位置，这样图形将保持完整。记住要将 `\usepackage{graphics,graphicx}` 加在导言区。

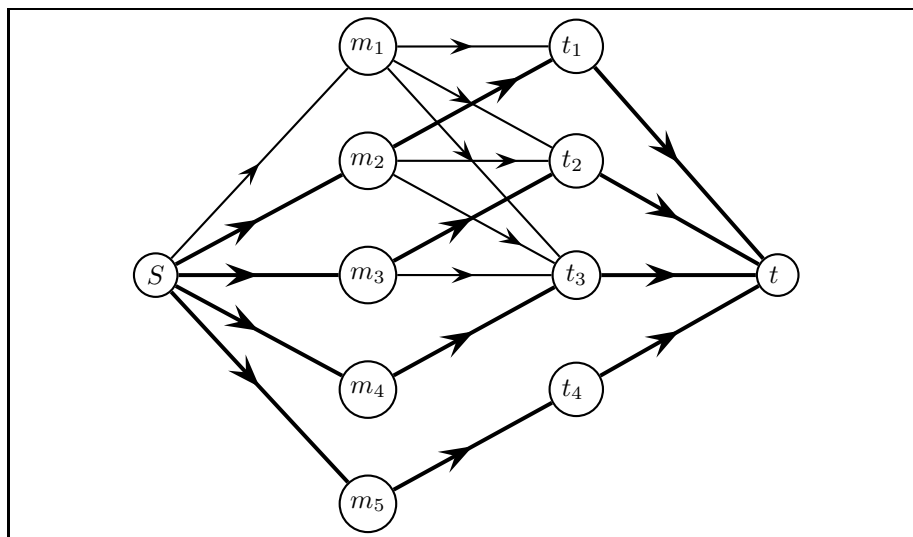
将你的文件转换为 .ps 格式。如果你要 pdf 格式，它可以通过 `pstopdf` 命令得到。

5 最常见的错误和一些建议

- 首先在你的系统中尝试编译一些简单的代码以确任需要的宏包在你的系统中已经安装了。
- 确定你在导言区加了图形宏包 (`graphics`、`graphicx`)。

- 在一对 $\$$ 内使用 `psmatrix`。
- 在 `psmatrix` 中不要留空行；它将导致错误。
- 有时我们插入命令 `\ncline{2,2}{2,3}` 希望在输出中两个节点连结起来，但是我们发现在两个节点之间没有线。检查一下这两个节点在你的矩阵中真实的存在。如果它们中的一个点没有，在不存在的节点之间的线怎么可能被画出呢？这个错误会当我们在节点之间有空节点的时候发生。
- 当我们使用 `pstree` 环境时，我们经常会写一对大括号 `{}`，并且在括号内以新的 `\pstree` 开始。使用编辑器比如 `kile`（在 Linux 系统中）可以突出显示对应的括号。它很有帮助。
- 每次在写新嵌套树的时候另起新行，这将使得你的代码更有结构性，并且它将使得更容易编辑。
- 在写完每个嵌套的 `pstree` 之后编译代码。如果因为某些错误你的代码出错而且你又不解决，删除或者注释掉最新嵌套的代码，然后再编译一次。如果此时代码没有错误，它意味着最新的那个 `pstree` 有错需要修改。现在再次加入新的嵌套树的代码即可。

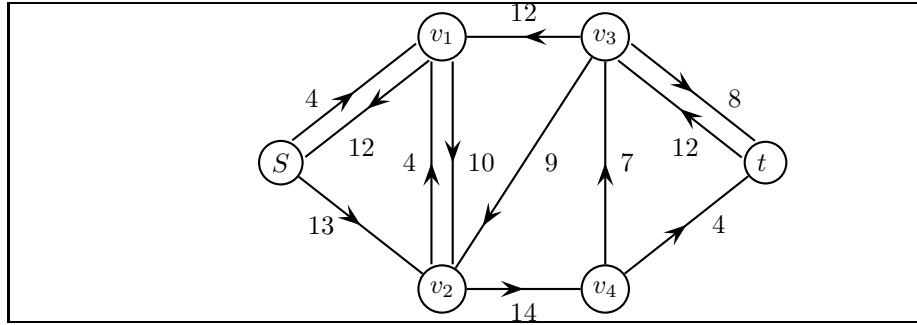
A 更多的图



- 观察到,在图中有些线是比较粗的。这里使用了 `\ncline` 的可选参数 `[linewidth=1.5pt]`。我们可以通过使用这种方法加粗某些路径从而使其

在其他路径中突出。线宽默认为 `linewidth=1pt`。

```
$
\psmatrix[colsep=2cm,rowsep=.7cm,mnode=circle]
&m_1&t_1\;\;
&m_2&t_2\;\;
S&m_3&t_3\;\;
&m_4&t_4\;\;
&m_5&[mnode=none,name=A]
\psset{ArrowInside=->,arrowscale=2}
\ncline{3,1}{1,2}
\ncline[linewidth=1.5pt]{3,1}{2,2}
\ncline[linewidth=1.5pt]{3,1}{3,2}
\ncline[linewidth=1.5pt]{3,1}{4,2}
\ncline[linewidth=1.5pt]{3,1}{5,2}
\ncline[linewidth=1.5pt]{3,2}{2,3}
\ncline{3,2}{3,3}
\ncline[linewidth=1.5pt]{4,2}{3,3}
\ncline[ArrowInsidePos=.8]{2,2}{3,3}
\ncline[ArrowInsidePos=.8]{2,2}{2,3}
\ncline[linewidth=1.5pt,ArrowInsidePos=.8]{2,2}{1,3}
\ncline{1,2}{3,3}
\ncline{1,2}{2,3}
\ncline{1,2}{1,3}
\ncline[linewidth=1.5pt]{5,2}{4,3}
\ncline[linewidth=1.5pt]{1,3}{3,4}
\ncline[linewidth=1.5pt]{2,3}{3,4}
\ncline[linewidth=1.5pt]{3,3}{3,4}
\ncline[linewidth=1.5pt]{4,3}{3,4}
\endpsmatrix
$
```

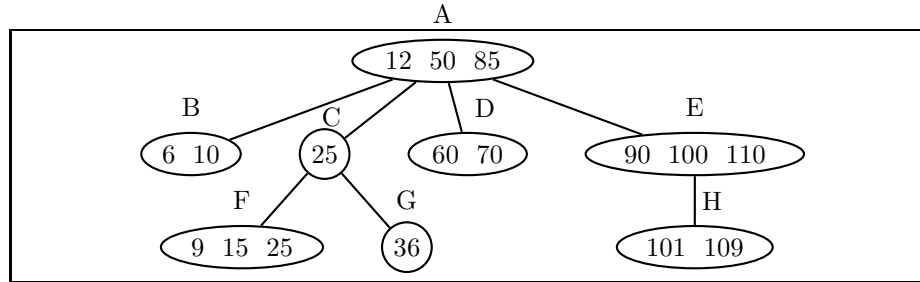



- 在这个图中，相同节点对之间有不同方向的箭头。这可以使用 **ArrowInside** 突出显示。如果相同节点对之间的两条线重合了，那么我们将标示不出线的相反方向，如果存在线的权，同样标示不出。使两条边有一个小的分离会更好。使用选项 **offset=4pt** 来达到两条边的分离。

```

$
\psmatrix[colsep=1.5cm,rowsep=1cm,mnode=circle]
&&v_1&v_3\\
&S &&t\\
&&v_2&v_4
\psset{ArrowInside=->,arrowscale=2}
\ncline{1,4}{1,3}^{12}
\ncline{3,4}{1,4}>{7}
\ncline{3,4}{2,5}>{4}
\ncline{3,3}{3,4}_{14}
\ncline{2,2}{3,3}<{13}
\ncline[ArrowInsidePos=.8]{1,4}{3,3}>[npos=.7]{9}
\psset{nodesep=0pt,offset=4pt}
\ncline{1,3}{3,3}>{10}
\ncline{3,3}{1,3}<{4}
\ncline{1,4}{2,5}>{8}
\ncline{2,5}{1,4}_{12}
\ncline{2,2}{1,3}<{4}
\ncline{1,3}{2,2}_{12}
\endpsmatrix
$

```

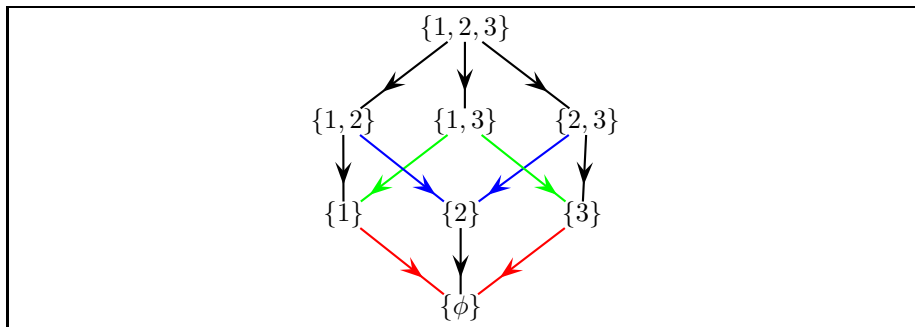


- 观察到 `\SpecialCoor` 的使用使得可以通过引用给定的名字对事物进行标记。对节点命名可以使用节点的可选参数命令。比如：`Toval[name=d]{12}`，这里 12 进入到椭圆形中如 , 并且这个节点被标记为 ‘d’ 以供内部使用。所以读者看不见它。

```

\pagestyle{empty}
\SpecialCoor
\pstree[levelsep=35pt,name=a]{\Toval[name=b]{12~~50~~85}}{
\Toval{6~~10}
\pstree{\Tcircle[name=c]{25}}{\Toval[name=f]{9~~15~~25} \Tcircle[name=g]{36}}
\Toval[name=d]{60~~70}
\pstree{\Toval[name=e]{90~~100~~110}}{\Toval[name=h]{101~~109}}
\uput{.5cm}[90](b){A}
\uput{.5cm}[90](a){B}
\uput{.37cm}[80](c){C}
\uput{.5cm}[70](d){D}
\uput{.5cm}[90](e){E}
\uput{.5cm}[90](f){F}
\uput{.5cm}[90](g){G}
\uput{.5cm}[70](h){H}

```

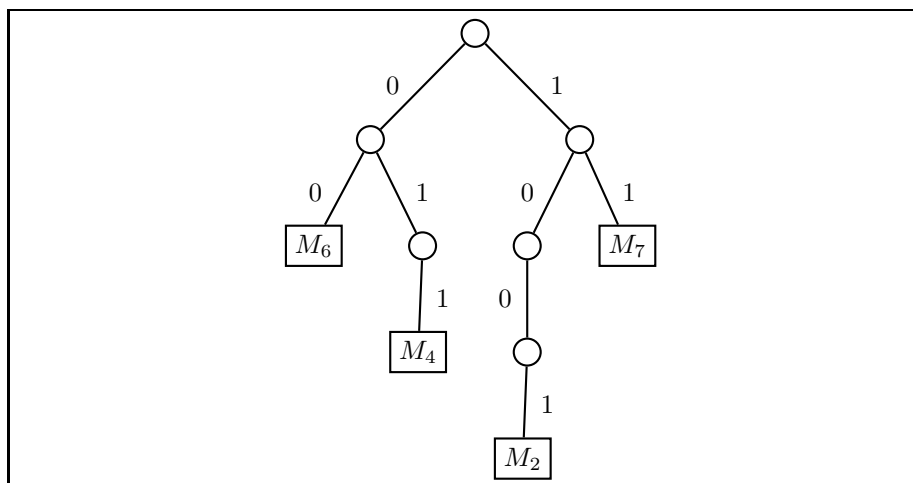


- 在上述图中，黑色的线是树的一部分，在 $\{2,3\}$ 和 $\{2\}$ 之间的线实际上是不同层级的两个节点之间的线。为了识别节点，节点的‘名字’被命名为‘TThree’ ‘Two’ 等。通过引用这些名字，用 `\ncline{TThree}{Two}` 可以画出不同层级节点之间的线。彩色的线是不同层级的节点之间的线。

```

\pstree[levelsep=35pt,ArrowInside==>,ArrowInsidePos=.75,
arrowscale=2,labelsep=14pt]
{\Tr{$\{1,2,3\}$}}{
\pstree{\Tr[name=OT]{$\{1,2\}$}}{
\Tr[name=One]{$\{1\}$}
}
\pstree{\Tr[name=OTThree]{$\{1,3\}$}}{
\psset{linestyle=none,ArrowInside=none}{
\pstree{\Tr[name=Two]{$\{2\}$}}
{\psset{linestyle=solid,ArrowInside==>}\Tr[name=Phi]{$\{\phi\}$}}
}
\pstree{\Tr[name=TThree]{$\{2,3\}$}}{
{\Tr[name=Three]{$\{3\}$}}
\ncline[ArrowInsidePos=.9]{TThree}{Two}
\ncline[ArrowInsidePos=.9]{OTThree}{One}
\ncline[ArrowInsidePos=.9]{OTThree}{Three}
\ncline{One}{Phi}
\ncline{Three}{Phi}
\ncline[ArrowInsidePos=.9]{OT}{Two}
}
}

```

- 为了得到矩形或者正方形的节点，一种可以的方式是使用 `\psframebox{Example}`，它将在括号内的参数的周围形成一个盒子，`\psframebox{Example}` 将生成 Example。

```

\newcommand{\pbox}[1]{\psframebox{#1}}
$
\pstree[levelsep=40pt]{\Tcircle{~}}{
\pstree[levelsep=40pt]{\Tcircle{~}^{0}}{
\Tr{\pbox{M_6}}^{0}\pstree[levelsep=40pt]{\Tcircle{~}_{1}}{
\Tr{\pbox{M_4}}_{1}}}\pstree[levelsep=40pt]{\Tcircle{~}_{1}}{
\pstree[levelsep=40pt]{\Tcircle{~}^{0}}{
\pstree[levelsep=40pt]{\Tcircle{~}^{0}}{
\Tr{\pbox{M_2}}_{1}}}\Tr{\pbox{M_7}}_{1}
}}
$

```

这并不是所有。很多命令现在并没有讨论到，但是你已经能够开始使用 PSTricks 来得到图论中精致的图形。慢慢地，你可以根据个人需要，通过浏览网站来探索新的命令。更多的 PSTricks 资料请参阅 [6]。可以对你提供支持的另外一个地方是 PSTricks 的邮件列表。

参考文献

- [1] PSTricks: PostScript macros for Generic \TeX , User's Guide, TIMOTHY VAN ZANDT, 1993.

- [2] PsTricks-add: additional Macros for pstricks, HERBERT VOSS, 2005.
- [3] Drawing Trees with PStricks, DOUG ARNOLD, 2002.
- [4] L^AT_EX Companion, MICHEL GOOSSENS, FRANK MITTELBACH, ALEXANDER SAMARIN, ADDISON-WESLEY, 1994.
- [5] PsTricks mailing list: pstricks@tug.org
- [6] Website <http://tug.org/PSTricks/>