Largest Empty Convex Polygon 用户手册

# 1 简介

Largest empty convex hull问题算法分为三个阶段：

1. 计算以每个点为kernel的右侧所有点的极角排序；
2. 生成每个点的VG（visibility graph）图;
3. 生成每个点对应的最大凸链。在生成的过程中保存最大凸链即为最终结果。

# 2 界面介绍

运行程序，首界面如图 1所示：

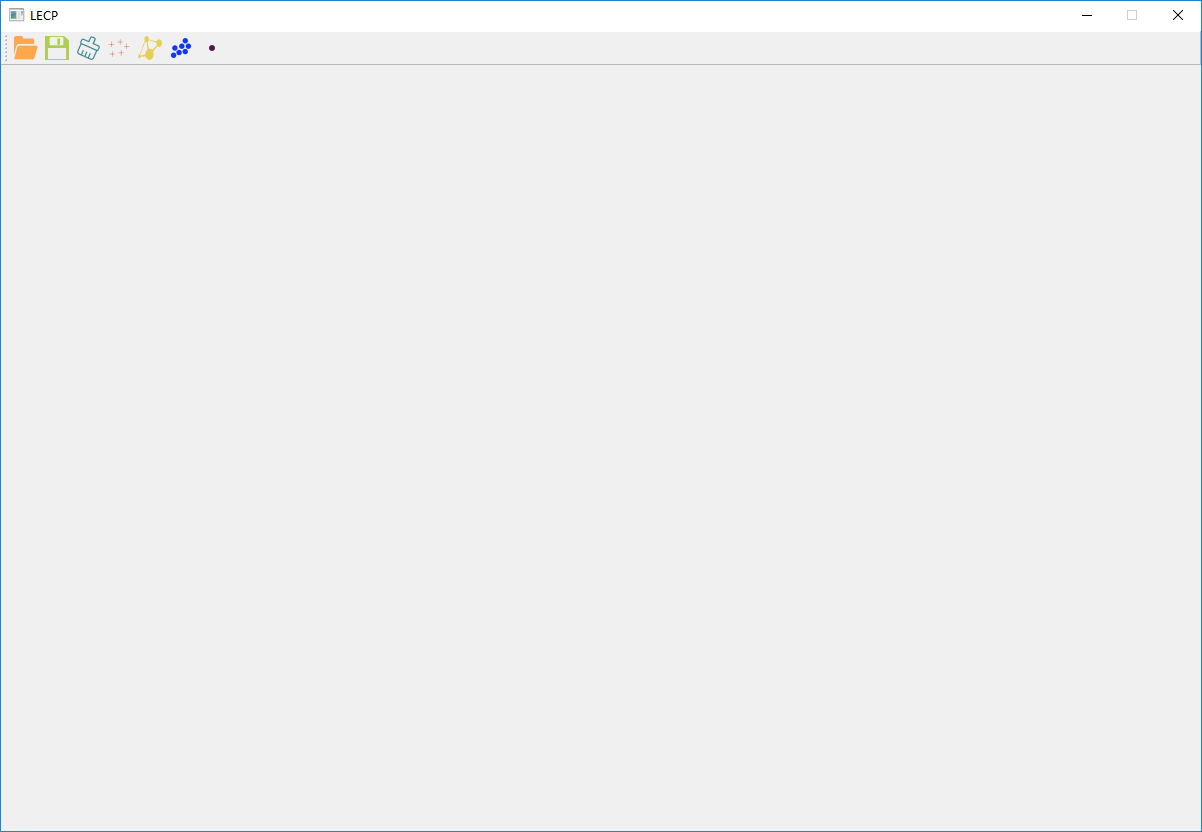


图 1

整个界面分成两大模块，最上面一栏为工具栏，下面空白区域为用户输入和显示区域。

工具栏按钮作用：

：打开并读取文件 。文件中存放点击，文件内容中第一行为点的总个数n，接下来的n行对应n个点的坐标，格式为“x y”，横纵坐标见用空格分割。

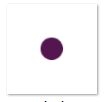
：保存文件。将当前屏幕上的所有点保存到文件中，文件内容格式同上述打开的文件。

:清空屏幕上所有点。清空当前所有点，重新输入。

:随机点生成。点击该按钮弹出对话框，用户输入要生成的点的个数n，即可在屏幕上随机生成n个点。

:显示最终的Largest Empty Convex Hull.

：所有点演示。演示动画时，每个点都演示一遍。

:单点进行演示。输入需要演示的点的label,可以仅对该点进行动画演示。

# 3 流程演示

1. 在屏幕上任意位置鼠标左键点击生成点（见图 2）：

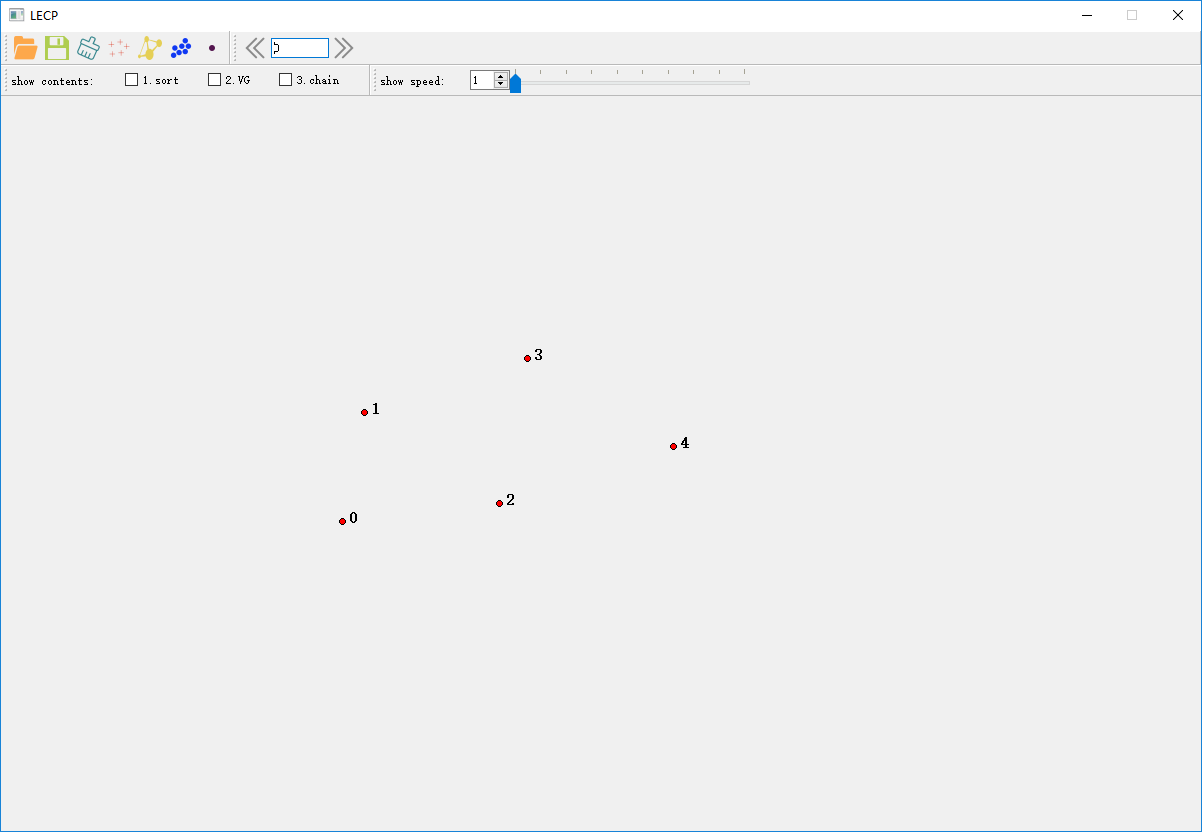


图 2

1. 选择进行动画演示的点：

当前点会变为天蓝色（见图 3）：

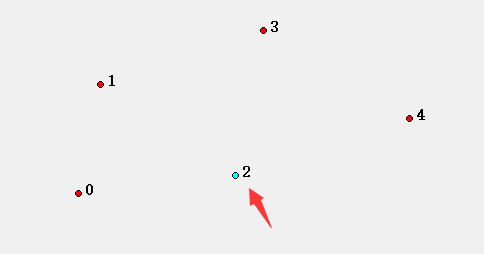


图 3

1. 选择要演示的动画：



Sort:是否演示排序动画；

VG:是否演示可视图动画；

Chain：是否演示最长凸链计算动画。

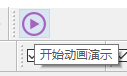
DCEL:是否显示DCEL动画；

Show Q:是否显示可视图计算过程中点的队列Q情况；

Show L:是否显示可视图中每条边为始边最长凸链长度。

1. 调整动画演示的速度：数值越大，速度越快



(5)点击开始动画演示：

# 4 DCEL动画介绍

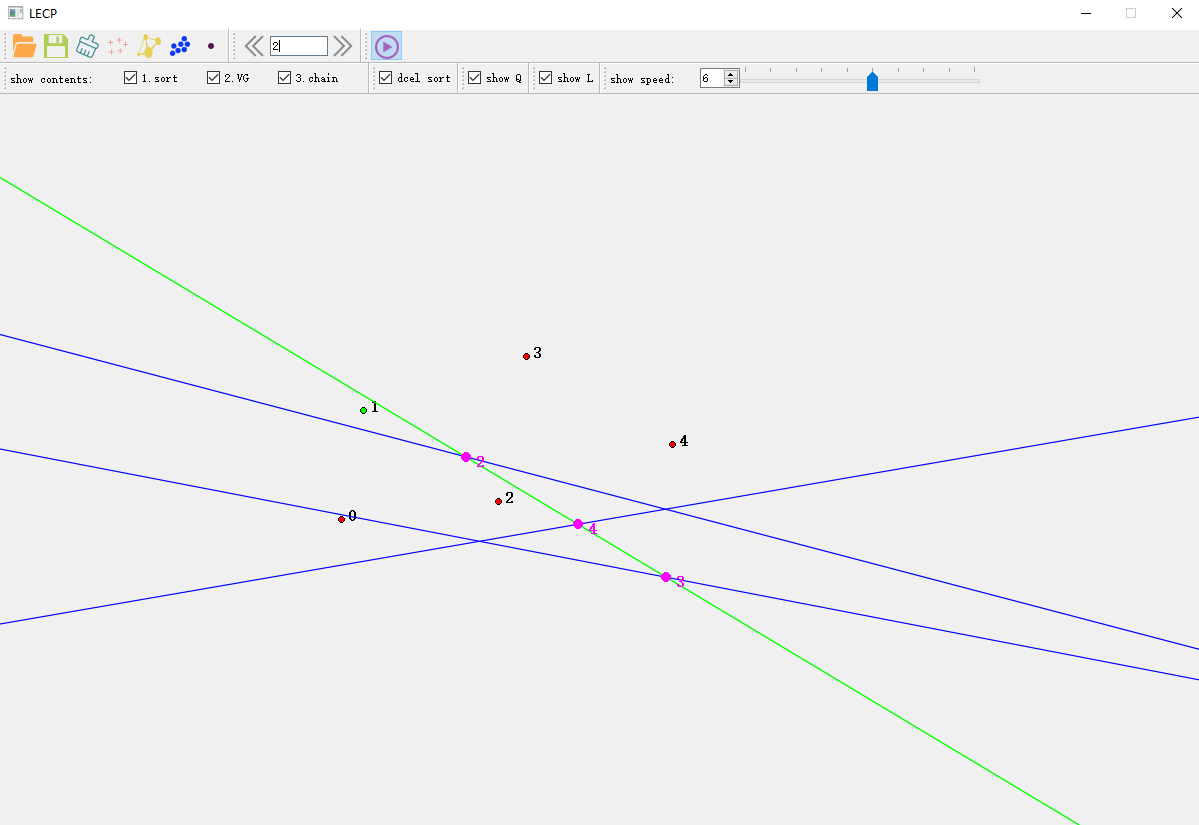


图 4 DCEL动画演示

在进行DCEL动画演示过程中，绿色点为当前点（label 1），绿色直线为对应的对偶图中直线。绿色直线（当前点的对偶直线）与蓝色直线（当前点右侧所有点的对偶直线）相交，交点以枚红色显示(2,4,3)，其对应的数字从左到右的顺序即为以当前点（p1）为kernel的极角排序结果（p2,p4,p3）。见图 4。

*注：由于屏幕有限，如果两条直线交点在屏幕之外则不进行显示。*

# 5 VG动画

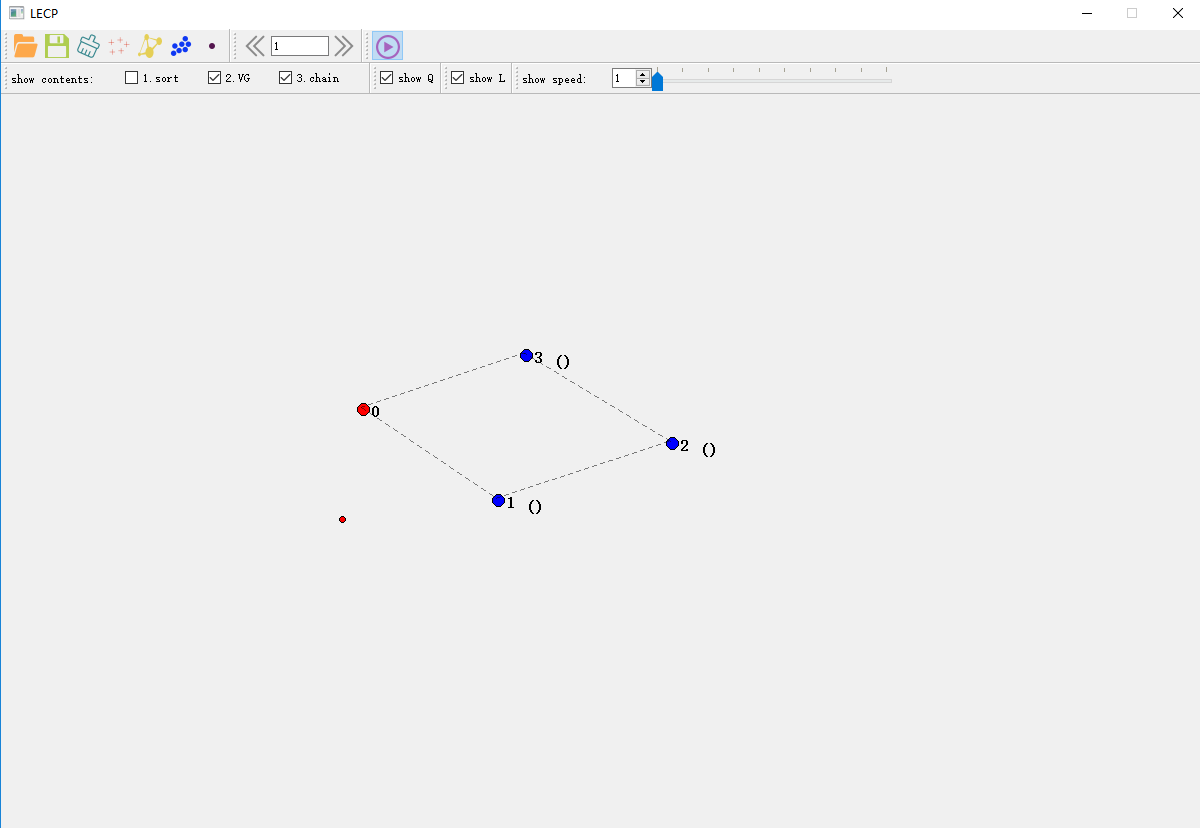


图 5

按逆时针顺序显示最终kernel右侧所有点的极角排序及生成的star-polygon结果（见图 5）。

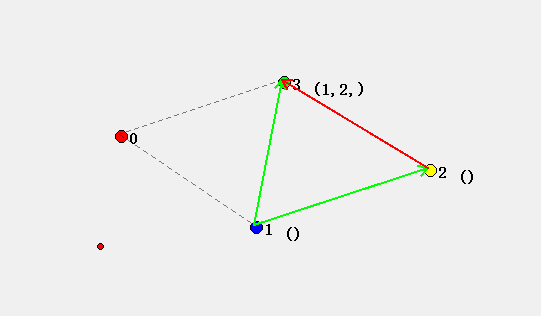


图 6

黄点和绿色点是做可视关系判断的一对点，若可见，则生成一条从low index到high index的边，当前新增的边为红色，之前的为绿色，点index处括号内是该点之前的点，被该点挡住的点留在Q(括号)中（如图 6）。

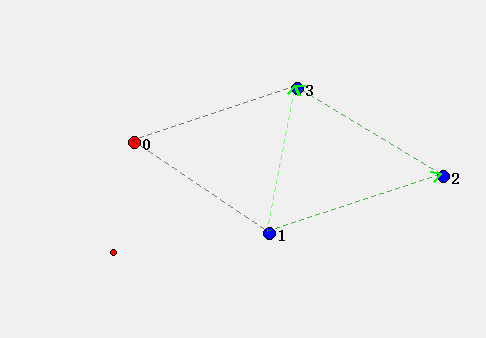


图 7 可视图(VG)结果

绿色的点线图为生成的可是图(VG)的结果。

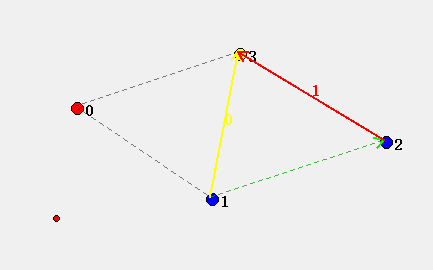


图 8

顺时针遍历每个蓝色点，在每个点处计算以当前点每一入边为始边的最长凸链长度。绿边为当前点的出边，黄边为当前点的入边，蓝边为当前点以处理的出边，红边为当前点已处理的入边，黑边为当前最长凸链的组成，边上的数字表示以该边为始边的最长凸链的长度(见图 8)。

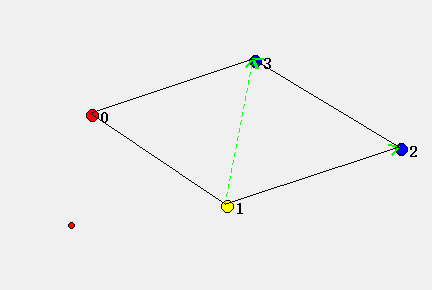
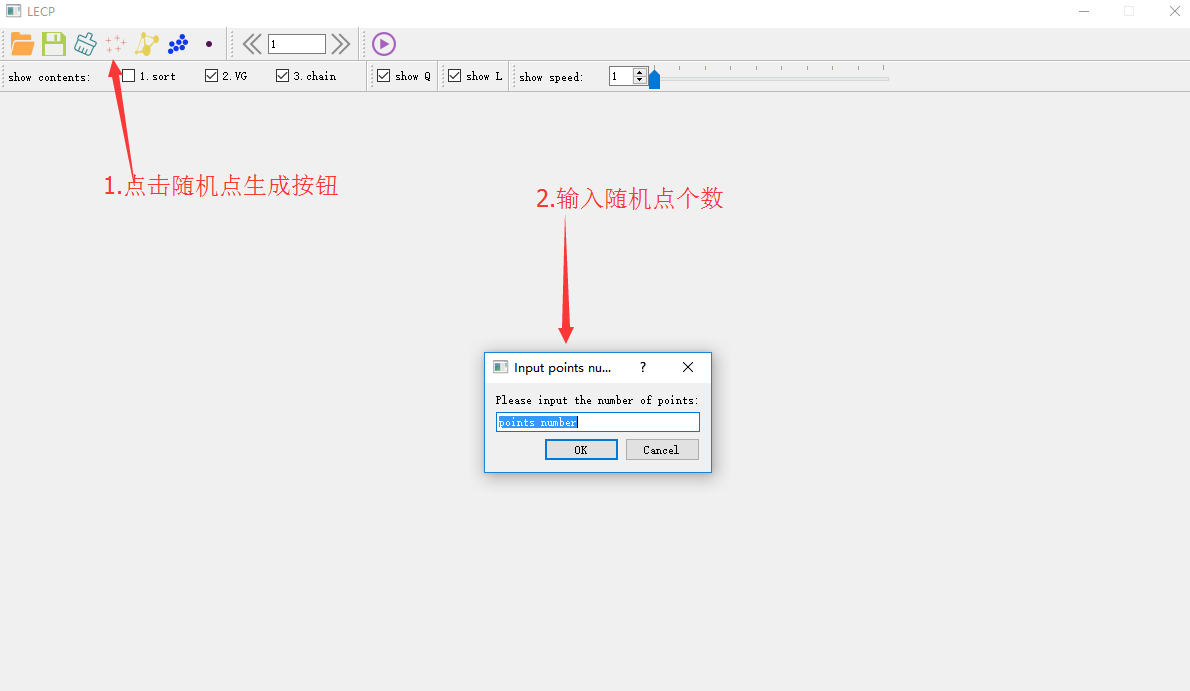


图 9

如图 9，黑色多边形为当前kernel对应的star-polygon的最大凸多边形。

# 6 其他功能

## 随机点生成





## 逐点演示

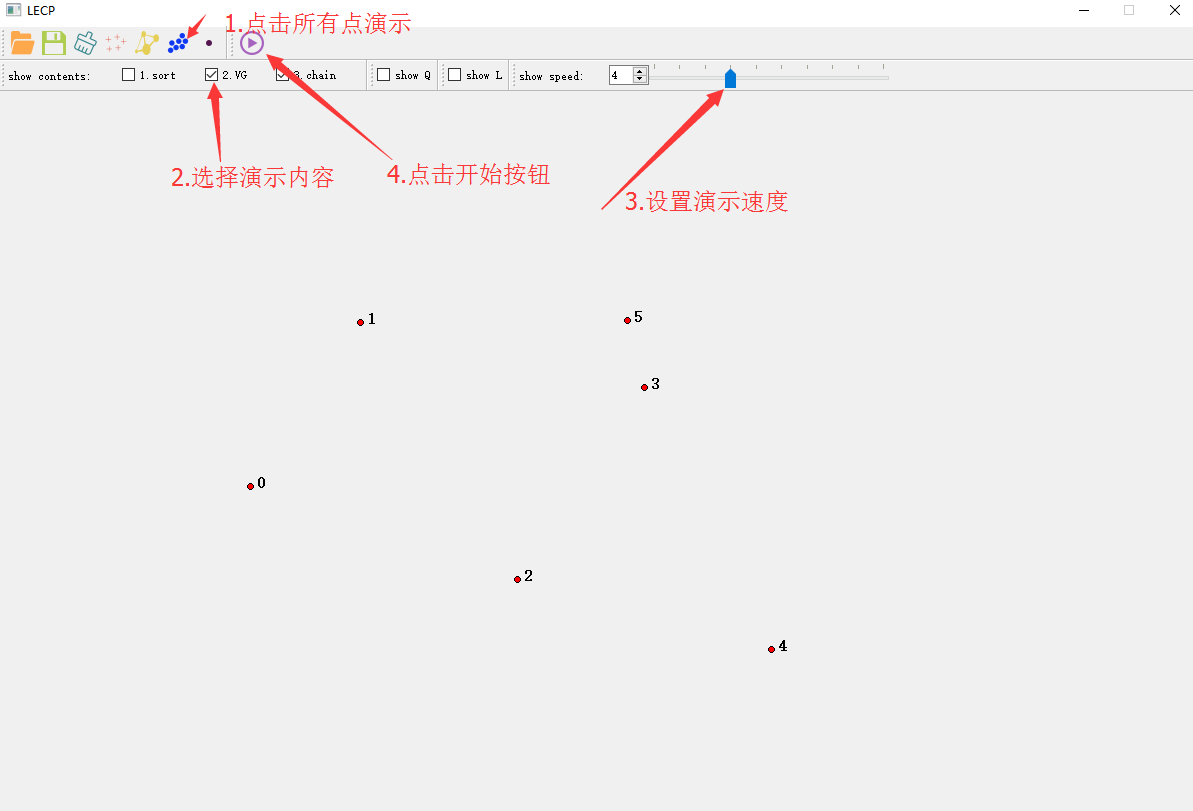


图 10 所有点演示

设置演示内容、速度，点击逐点演示按钮演示动画，从右向左，每个点分别作为kernel进行LECP（Largest Empty Convex Polygon）计算（见图 11），从而得到最终结果（见图 12）。

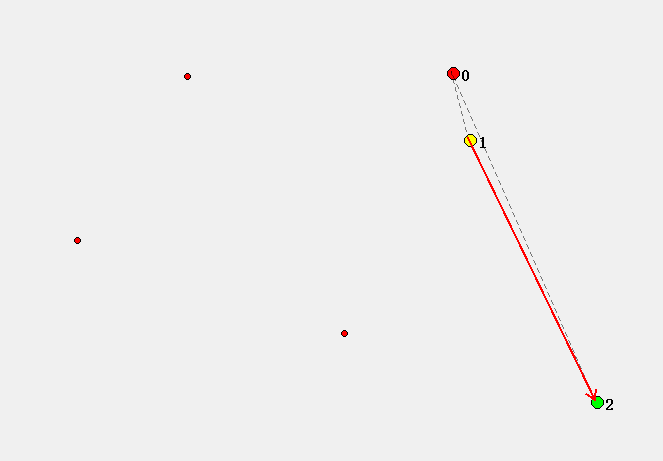


图 11 计算当前点的最大凸链

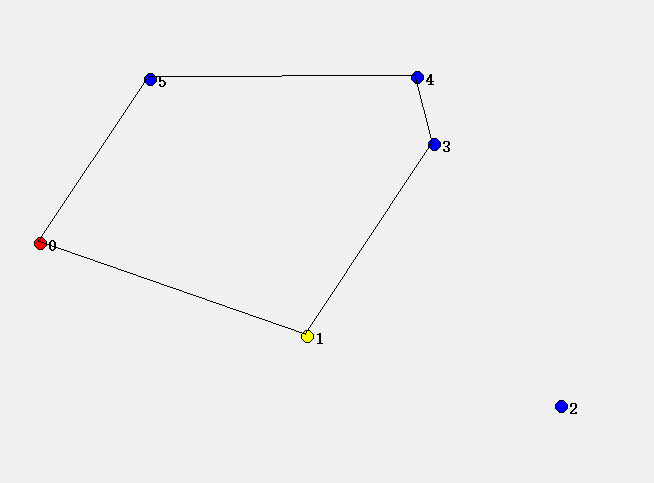


图 12 最终结果