19211371-麻言荣-笔记3

1. 文献信息：

作者: 斯图尔特·罗素 和 彼得·诺维格

论文题目:Artificial Intelligence: A modern approach （4th）

发表途径:出版社

发表时间:2020年

1. 问题意义：

论文研究问题：

已知智能体的初始状态和目标状态,求解一个行动序列使得智能体能从初始状态转移到目标状态,称为**搜索问题**。如果所求序列可以使得总耗散最低,则间题称为**最优搜索问题**。

研究分类：

1.按是否使用启发式信息：

盲目搜索：按预定的控制策略进行搜索，在搜索过程中获得的中间信息并不改变控制策略；启发式搜索：在搜索中加入了与问题有关的启发性信息，用于指导搜索朝着最有希望的方向前进，加速问题的求解过程并找到最优解。

2.按问题的表示方式：

状态空间搜索：用状态空间法来求解问题所进行的搜索；与或树搜索：用问题归约法来求解问题时所进行的搜索

研究意义：

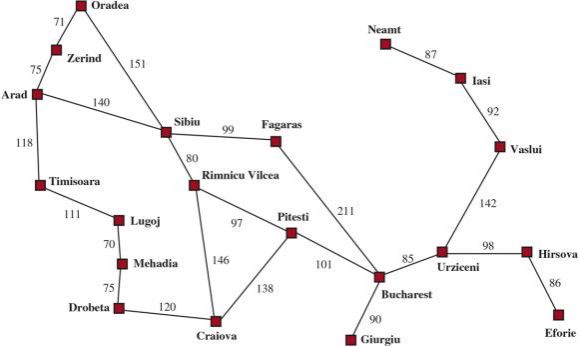
有利于减少路径的重复研究、提高最优搜索成功率。进行某一特定的搜索，都要花费大量的时间，对有关节点进行全面的调查研究，当我们有着最优搜索策略就可以借鉴、改进和部署自己的工作。只有这样方能做到有所创新、有所前进，否则容易造成重复劳动，导致人力、物力、财力的浪费。

1. 思路方法：

论文的研究思路和研究方法：

第二章节主要讲述了有关搜索的相关方法结论，从基础的节点扩展、进入搜索代理的AI到广度优先搜索、深度优先搜索，引入步骤成本，统一成本搜索，到深度有限搜索、迭代深化深度优先搜索、双向 BFS、星搜索。（这里挑较为熟知的概念进行整理）

首先引入了一张罗马尼亚的地图，如下



图一 罗马尼亚地图

假设我们现在位于最西南处的Drobeta，目标是去最西北角的Oradea。我们目前有一张地图，如上图所示，我们想设计一个能解决这个搜索问题的agent并且装载在你的2121年的自动飞行器上。那么我们该怎么解决这个问题呢？

对于我们来说，我们的任务环境是已知的，完全可观察的，确定性的。我们可以以任意方式计算如何前往我想要的目的地。一般来说我们需要遵循以下四个步骤：

1.目标公式化（GOAL FORMULATION）: 或者说明确我们的目标是什么

2.问题公式化（PROBLEM FORMULATION）: 描述我们达到目标所需的状态，以及行动的描述——世界相关部分的抽象模型

3.搜索（SERACH） 在智能体最终告诉我们这个答案是什么之前，智能体需要在其模型中模拟一系列行动，进行搜索，直到找到达到目标的一系列行动。 这样的序列叫做解（Solution） 。

4.执行（EXECUTION）: 智能体现在可以执行解决方案中的操作。

当然我们还要考虑到不重复性，使我们的系统路线是一个闭环系统，最后一系列的行动 形成了一条路径，而解就是从初始状态到目标状态的一条路径，即从Drobeta到Oradea的一条路径。一般来说， 最优解的路径成本是所有解中最低的。状态空间可以表示为一个图，其中结点是状态，它们之间的有向边是行动，就如上面的罗马尼亚地图一样。

另外我在GitHub找到一个很形象的例子，说明下面的概念，相当于简化了罗马尼亚地图，如下图。

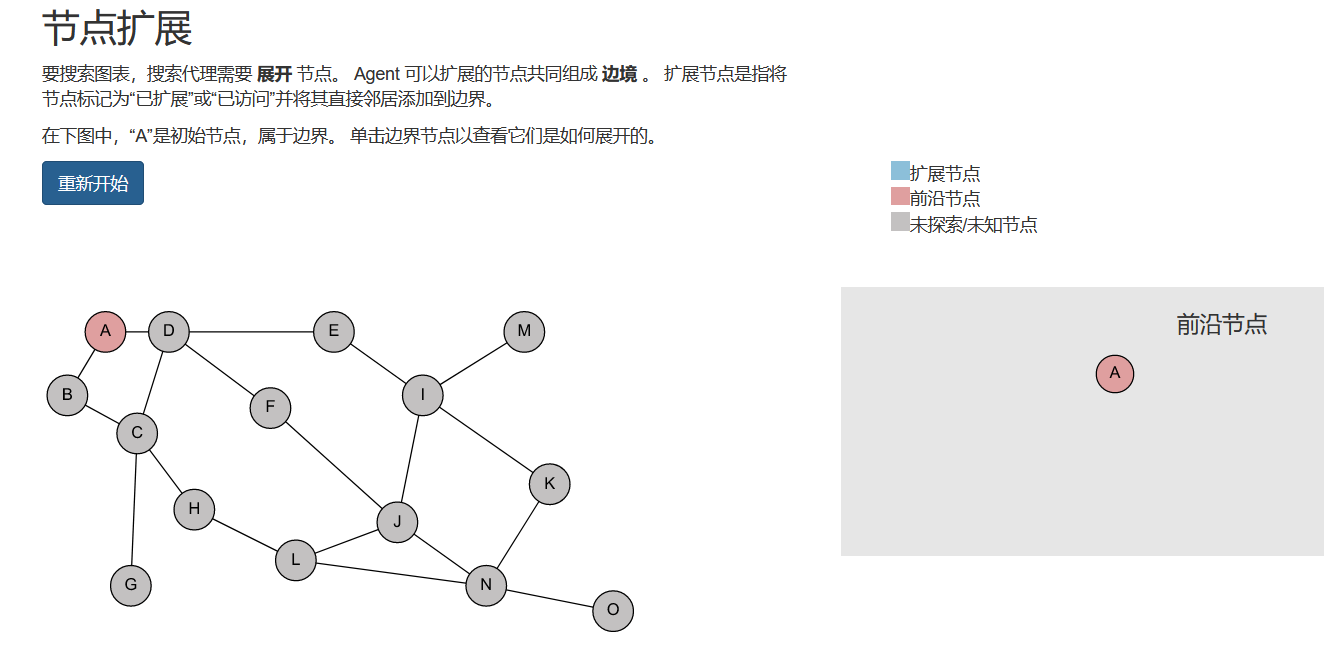


图2 全节点分布图

在广度优先搜索中，最早发现的节点被下一步扩展，即更早加入边界的节点被更早地扩展。为此，广度优先搜索算法使用 FIFO（先进先出）队列。 下图由以“A”作为初始节点的广度优先搜索算法进行探索。

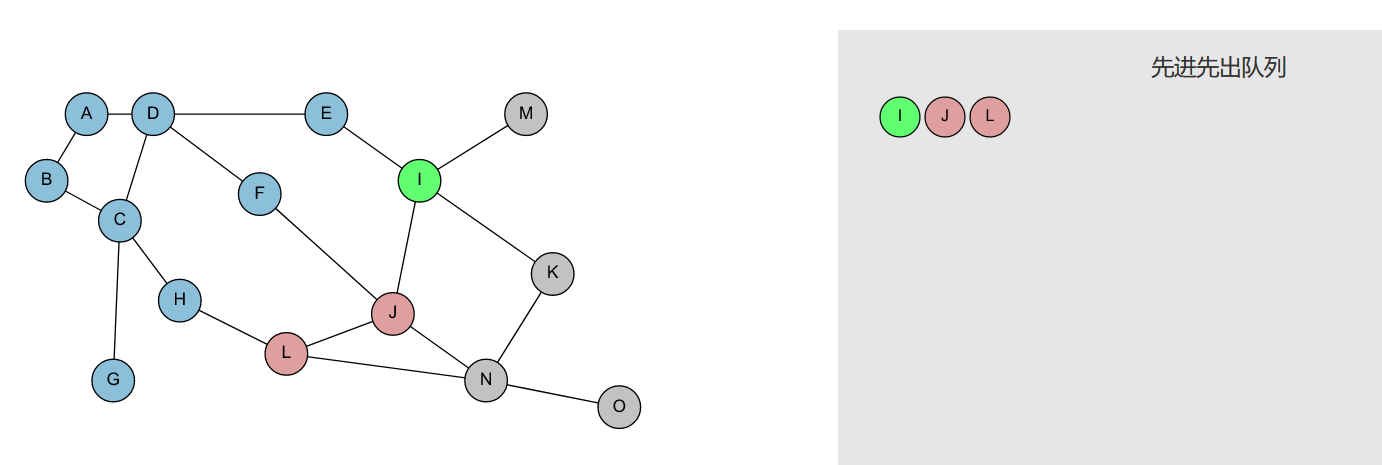


图3 广度优先

深度优先搜索：在深度优先搜索中，最晚发现的节点被下一步扩展，即后来加入边界的节点被稍后扩展。为了实现这一点，深度优先搜索算法使用 LIFO（后进先出）队列。 下图由深度优先搜索算法探索，其中“A”作为初始节点。

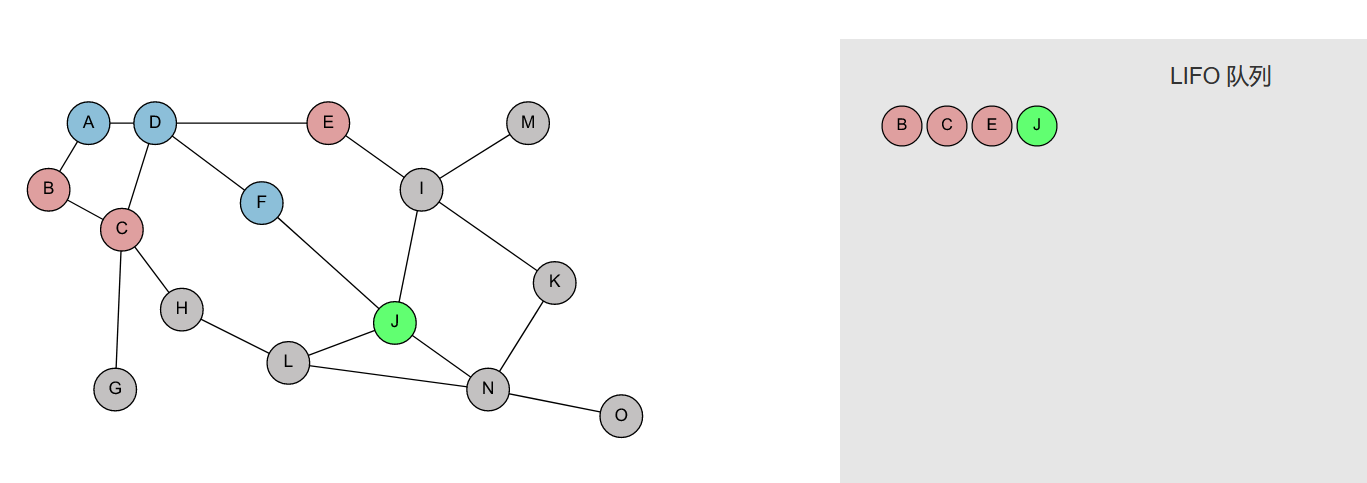
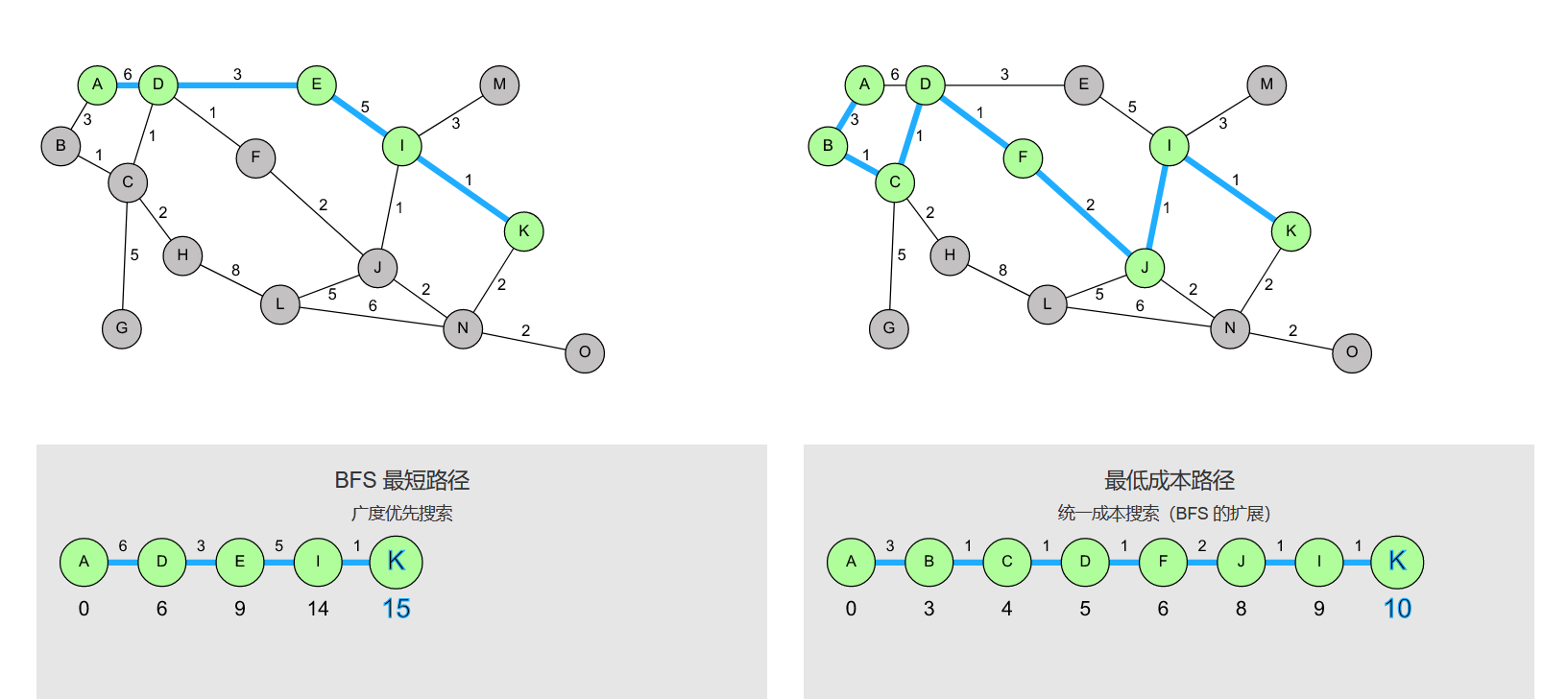


图4 深度优先

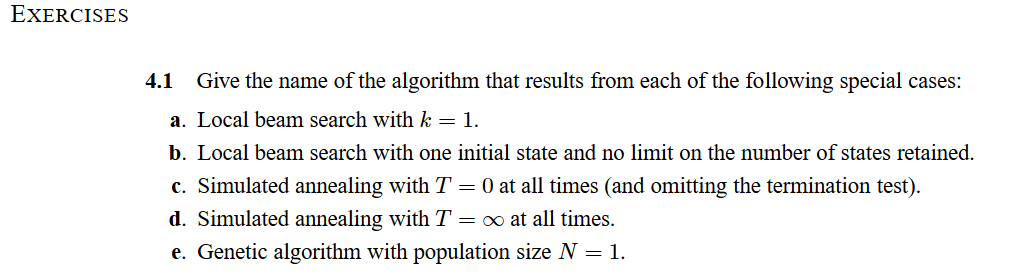
步骤成本：到目前为止，我们图中的所有边都具有相同的成本。（这就是为什么我们没有费心在图中提及成本的原因）。对于这类图，广度优先搜索是最优的，因为它总是首先弹出最浅的节点。 对于某些情况下探索节点时涉及步骤成本，需要扩展BFS算法。

图中任何节点的相关成本计算为从初始节点开始的“最低成本路径”的成本。 初始节点的成本为 0。

下面是来自同一初始节点“A”的“最低成本路径”和“广度优先搜索路径”的并排比较。BFS 无法为许多节点推导成本最低的路径。BFS的路径节点数最少，但它没有考虑步骤的成本。



1. 课后习题：



a.使用k=1进行的本地光束搜索是爬山搜索。

b.使用k=∞进行本地波束搜索：严格来说，这没有意义。这个想法是，如果每个后继者都被保留(因为k是无界的)，那么搜索就类似于宽度优先搜索，它在添加下一层之前添加一个完整的节点层。从一种状态开始，该算法将与宽度优先搜索基本相同，只是每一层都是一次生成的。

c.始终使用T=0模拟退火：忽略终止步骤将立即触发的事实，搜索将与第一选择的爬山

相同，因为每个向下的后继者都会以1的概率被拒绝。

d.始终以T=无穷大模拟退火：忽略终止步骤永远不会被触发的事实，搜索将与随机游走相

同，因为每个后继者都将以概率1被接受。在这种情况下，随机游走近似等价于深度优先搜索。

e.种群大小为N=1的遗传算法：如果种群大小为1，则被选择的两个亲本将是同一个体；交叉产生一个精确的个体副本；那么突变的机会就很小。因此，该算法在个体空间中执行随机游走。

5）启发思考：算法是一种大而一统的理论，搜索算法也处处在我们身边。教程中很多论证都看不懂，还有很多新的名词，对于笔记只能记下重要结论，希望自己学过更多的东西后，完善自己，回头再来看。也希望自己能应用到项目中去。