在线二分图匹配综述

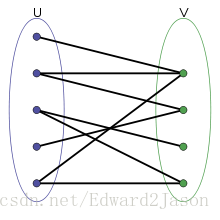
On-line Bipartite Matching

一、背景

众所周之，匹配是图论中的基本结构，在组合优化中起着不可或缺的作用。几十年来，人们一直在努力设计更有效的算法，以便根据基数(更一般地说，是它们的总权重)找到最大匹配。值得一提的是，二部图中的匹配在一些需要将实体从一个集合分配到另一个集合的情况下已经有了无数的应用(例如，匹配学生到学校，匹配医生到医院，匹配计算任务到服务器，以及在线媒体上的印象到广告商)。由于数字领域匹配市场的巨大增长，高效的在线匹配算法变得越来越重要。尤为关键的是，搜索引擎公司为在线匹配算法创造了机会，从而在数十亿美元的广告市场产生巨大影响。在这些应用程序的推动下，我们考虑了一个问题，即如何将一组一个接一个的印象与预先知道的一组广告商进行匹配。当一种印象到来时，它对广告商的优势就显露出来了，必须做出一个不可撤销的决定，即印象应该分配给哪个广告商。Karp, Vazirani和Vazirani [KVV90]给出了一个优雅的在线算法，称为排序，以找到竞争比为1−1/e的非加权二部图的匹配。他们还证明了这是可实现的最佳竞争比率。此外，Aggarwal等人[AGKM11]将其算法推广到顶点加权的在线二部图匹配问题，并证明了1−1/e的竞争比仍然是可以实现的。

二、二分图简介

维基百科中对二分图的介绍为：二分图是一类图(G,E)，其中G是顶点的集合，E为边的集合，并且G可以分成两个不相交的集合U和V，E中的任意一条边的一个顶点属于集合U，另一顶点属于集合V。一个简单的形象表示如下图：



因此，二分图的表示为：G=(U,V,E)。

如果|U|=|V|，即集合U和集合V的元素个数相等，该二分图称为“平衡二分图”(Balanced bipartite graph)。

如果U或V中顶点的度是相同的，则该二分图称为“双正则”（biregular）。

三、材料简介

1. An Optimal Algorithm for On-line Bipartite Matching （Richard M. Karp University of California at Berkeley & International Computer Science Institute Umesh V. Vazirani University of California at Berkeley Vijay V. Vazirani Cornell University）

评价一个在线算法的一种方法是比较它的性能与最好的离线算法对同一问题。因此，给定一个“利润”的度量，在线算法的性能可以通过其利润与最优离线算法的最坏情况比来衡量。这种通用的方法已经被应用在许多环境中，包括数据结构[SITa]、bin打包[CoGaJo]、图着色[GyLe]和k-服务器问题[MaMcSI]。在此，作者将其应用于二部图匹配，并证明了一个简单的随机在线算法可以达到最好的性能。

2. Understanding Zadimoghaddam’s Edge-weighted Online Matching Algorithm: Unweighted Case （Zhiyi Huang∗ Runzhou Tao Last Update: October 2019）

本文确定了Zadimoghaddam(2017)的边缘加权在线匹配算法中的一个关键算法成分，并给出了一个简化算法及其分析，以演示在未加权情况下它是如何工作的。

3. Edge-Weighted Online Bipartite Matching（Matthew Fahrbach Zhiyi Huang Runzhou Tao Morteza Zadimoghaddam May 6, 2020）

在线二分匹配及其变分是在线算法文献中最基本的问题之一。Karp, Vazirani和Vazirani (STOC 1990)为未加权问题引入了一种优雅的算法，该算法实现了1−1/e的最优竞争比。后来，Aggarwal等人(SODA 2011)将他们的算法和分析推广到顶点加权的情况。然而，除了平凡的1/2竞争贪婪算法外，关于最一般的边加权问题，我们知之甚少。在本文中，作者提出了第一个突破长期的1/2障碍的在线算法，实现了至少0.5086的竞争比，结果可以被看作是强有力的证据表明：edge-weighted双方的匹配是严格容易在网上设置子模块福利最大化。在线匹配算法的主要成分是一个叫做在线关联选择(OCS)的新子程序，它以一串顶点对作为输入，并从每对顶点中选择一个顶点。OCS不是使用一个新的随机位从每对中选择一个顶点，而是通过不同的对将决策负相关，并提供了相关性水平上的定量度量，最后将发现进一步的应用在其他在线优化问题。

四、非加权在线二分匹配

4.1问题描述

假设G (U,V,E)是一个有2n个顶点的二部图，使G包含一个完全匹配。设B是表示G (U,V,E)结构的n xn矩阵。B的行对应U中的顶点(男孩)列对应V中的顶点(女孩);每条边在适当的位置用1表示。讨论了在G (U,V,E)中在线构造大匹配的问题。假设女孩顶点按预先选择的顺序到达，并且只在顶点到达时才显示与顶点相关的边。任务是在每个女孩顶点到达时，决定与哪个男孩顶点匹配，从而使得到的匹配大小最大化。或者，我们可以将匹配看作是在逐列显示矩阵时构造的。作为惯例，我们假定列的显示顺序为n,n-1，…1。

这一随机算法的性能用p（A）表示，定义为：MIN MIN E[由A实现的匹配大小]（期望值被A内部抛硬币所取代，G：女生节点的顺序）。

4.2排序算法

分析以下随机在线匹配算法的性能，我们将其称为排序算法:初始化:对男孩的格子进行随机排列，从而给每个男孩分配一个随机优先级或排序。匹配阶段:当每个女孩到达时，将她匹配到最高级别的符合条件的男孩(如果有的话)。乍一看，分析随机算法似乎更自然，即每次有女孩到来时，从符合条件的男孩中随机选择一个男孩。然而，随机算法的性能几乎不如确定性贪婪算法;它只实现了n+o (logn)的期望大小匹配。

RANDOM在这个例子中表现不佳，因为它在第一次移动时将太多精力集中在密集的矩阵上半部分上，从而忽略了稀疏矩阵下半部分的关键边缘。排名有一种内隐的自我纠错机制，它倾向于支持那些目前符合资格但过去最不符合资格的男孩。正是这种排名的特征使它即使在考虑局部密度的图表上也能很好地表现出来。

4.3原理简介

对偶原理:排序初始化阶段结束后，男孩顶点和女孩顶点都有顺序(女孩顶点的顺序是预先选择的，男孩顶点的顺序是随机选择的)。此时，男孩顶点和女孩顶点之间是对称的:如果我们交换男孩和女孩的角色，让男孩根据他们的排名到达并选择排名最高的合适的女孩，排名的结果保持不变。

五、加权在线二分图匹配

5.1与非加权图的区别

边缘加权的情况更加模糊。这在一定程度上是因为，如果没有额外的假设，就不存在竞争算法。为了了解这一点，考虑两个边缘加权问题的例子，每个例子都有一个广告客户和两个印象。边缘的第一印象的重量是1在这两个实例,和第二个印象是0的重量在第一个实例和W在第二个例子中,对于一些任意大W .在线算法不能区分两个实例一到的第一印象,但它必须决定是否要为广告客户分配这种印象。如果不给它赋值，第一个实例中的竞争比率为0，而给它赋值，第二个实例中的竞争比率为1/W。这个问题不能被解决，除非分配给广告客户的印象是一种选择。在展示式广告中，给广告客户更多的印象只会让他们更开心。换句话说，我们可以给任何一个广告客户分配多个印象。然而，我们并没有获得分配给它的所有边的权值，而是只确认最大权值(即，目标等于分配给每个广告商的最重边权值的总和)。这相当于允许广告客户免费处理之前匹配的边，为新的更重的边腾出空间。这一假设通常被称为自由处置模型。在展示广告文献[FKM+09, KMZ13]中，自由处置假设因其自然经济解释而受到广泛接受和应用。最后，边权自由分配在线二部图匹配是单调子模福利最大化问题的一种特殊情况，在这种情况下我们可以使用已知的1/2竞争贪婪算法[FNW78, LLN06]。

尽管自Karp等人[KVV90]的重要工作以来，在线匹配方面的研究已经进行了30年，但找到一种实现竞争比大于1/2的边加权在线二部图匹配的算法仍然是一个诱人的开放问题。本文给出了一种新的在线算法，肯定地回答了这个问题，突破了长期存在的1/2障碍(在自由处理下)。定理1有一个0.5086竞争算法的边缘加权在线二部图匹配。考虑到Kapralov、Post和Vondrak [KPV13]对单调子模福利最大化的竞争1比1/2的困难结果，我们的算法表明，在在线设置中，边加权二部分匹配比子模福利最大化严格更容易。从现在起，我们将在二部图中使用更正式的离线和在线顶点术语，而不是广告主和印象。我们的主要技术贡献之一是一个名为在线关联选择(OCS)的新算法成分，这是一个在线子程序，它以一系列对顶点作为输入，并从每对顶点中选择一个顶点。OCS不是使用一个新的随机位来做每一个决策，而是询问不同对之间的决策在多大程度上是负相关的，并最终保证出现在k对中的一个顶点至少被选择一次，其概率严格大于1 - 2 - k。给定一个OCS，我们可以通过以下(勉强)随机化算法获得一个优于1/2竞争比的无加权在线二部图匹配。对于每个在线顶点，要么选择一对离线邻居并让OCS选择其中之一，要么确定性地选择一个离线邻居。更具体地说，在没有确定匹配的邻域中，找出匹配最少的邻域(即出现在最少对中的邻域)。如果至少有两个，就选两个;否则，确定地选择一个。我们在附录a中对该算法进行了分析。尽管上述算法的竞争比远不如Karp等人[KVV90]提出的最优1 - 1/e比率，但它得益于改进的可推广性。为了将此算法推广到边权问题，我们需要一个合理的“最小匹配”离线邻居的概念。假设一个邻居的最重边权值分别为1或4，概率都是1/2，另一个邻居的最重边肯定是2，它们与当前在线顶点的边权值都是3。哪一个比较不匹配?为了解决这个问题，这时，使用Devanur、Jain和Kleinberg [DJK13]的在线原对偶框架来匹配问题，以及Devanur等人[DHK+16]的边加权在线二部匹配问题的替代公式。简而言之，我们通过权值级别来考虑每个离线顶点的贡献，在每个权值级别上，我们考虑与该顶点匹配的最重边至少在该级别上拥有权值的概率。这是最大边权的互补累积分布函数(CCDF)，因此我们称其为CCDF视点。最后，对于每个离线邻居，我们利用对偶变量来计算每个权重层的报价，如果当前在线顶点与之匹配的话。在所有权重级别上聚集的净报价最大的邻居被认为是“最不匹配的”。

5.2相关文献

在线加权二部图匹配算法的文献很多，但大多是通过假设离线顶点容量大或者在线顶点的一些随机信息提前已知来实现竞争比大于1/2。我们注意到最近在更一般的环境中有几个重大的进展，包括不同的到达模型和一般(非二部图)[HKT+18, GKS19, GKM+19, HPT+19]。大的能力。离线顶点的容量是可以分配给它的在线顶点的数量。利用大容量假设到beat1/2可以追溯到20年前Kalyanasundaram和Pruhs [KP00]。Feldman等[FKM+09]给出了一种(1−1/e)的显示广告竞争算法，相当于大容量下的边加权在线二部图匹配。在类似的假设下，Ad2 Words [MSVV05, BJN07]得到了相同的竞争比，其中脱机顶点对可以分配给它们的总权重有一定的预算约束，而不是印象的数量。从理论的角度来看，在线匹配文献的主要目标之一是提供竞争比大于1/2的算法，而不对离线顶点的容量做任何假设。随机到达。如果我们知道在线顶点的到达模式，我们通常可以利用这些信息来设计更好的算法。典型的随机假设包括，假设在线顶点是从一些已知或未知的分布[FMMM09, KMT11, DJSW11, HMZ11, MGS12, MP12, JL13]，或它们以随机的顺序到达[GM08, DH09, FHK+10, MY11, MGZ12, MWZ15, HTWZ19]。这些工作达到了一个1 -方竞争比，如果大容量假设持有除了随机假设，或者至少1 - 1/e为任意容量。Korula, owkni和Zadimoghaddam [KMZ18]表明，如果在线顶点以随机顺序到达，且对容量没有任何假设，贪心算法对更普遍的子模福利最大化问题具有0.505竞争力。随机顺序假设特别合理，因为Kapralov等人[KPV13]证明，在健忘对手模型中，实现子模福利最大化意味着NP = RP。

5.3问题细化

边权在线匹配问题考虑了一个二部图G = (L, R, E)，其中L和R分别是左侧(LHS)和右侧(RHS)的顶点集，三角形数据⊆L×R是边的集合。每条边(i, j)∈E都与一个非负权值wij≥0相关，通过对缺失的边赋零权值，我们可以不失一般性地假设这是一个完全二部图，即E = L×R。LHS上的顶点是脱机的，因为它们都是预先为算法所知道的。然而，RHS上的顶点一次只到达一个在线。当一个在线顶点j∈R到达时，它的关联边及其权值将显示给算法，然后算法必须将j匹配到一个离线邻居。每个离线顶点可以被匹配任意次数，但只有其最重边的权重才会被计算到目标中。这相当于允许一个匹配的离线顶点i，比如说，到j，重新匹配到一个新的在线顶点j0，边权值为wij0 > wij，免费处理顶点j和边(i, j)。这个假设被称为自由处置模型。目标是最大化匹配的总权重。

5.4新内容

这部分介绍第三篇论文在线算法的新成分，它是广泛适用的和独立的利益。为了激励该技术，考虑以下无加权在线匹配情况下的思想实验，即对于任意i∈L和任意j∈r, wij∈{0,1}。首先我们回顾一下为什么所有将在线顶点与离线顶点匹配的确定性贪婪算法最多只能达到1/2竞争。考虑一个具有具有两个离线顶点和两个在线顶点的图的实例。第一个在线顶点与两个离线顶点相邻，算法确定地选择其中一个。然而，第二个在线顶点只与先前匹配的顶点相邻。具有独立随机比特的二选贪婪。我们可以通过随机匹配第一个在线顶点来避免上述问题，从而将期望匹配大小从1提高到1.5。本着这种精神，考虑下面的二选择贪婪算法。当一个在线顶点到达时，识别它的最不可能被匹配的邻居(在前几轮中随机)。如果有不止一个这样的邻位，选择任意两个，例如，从字典上，并与一个新的随机位匹配。否则，确定匹配最小匹配邻居。我们将前者称为随机轮，后者称为确定性轮。由于每个随机轮使用一个新的随机位，这相当于匹配在随机轮和非确定性轮中选择的最小数量的邻居。不幸的是，由于上三角图的存在，该算法也具有半竞争性。我们将这个标准例子推迟到附录B. 5的完全负相关的二选择贪婪。本思想实验的最后一种算法是两选择贪婪算法的一个假想变体，它将随机化轮数完美地负相关，使得每一个离线顶点在成为两轮随机化轮数的候选后都能进行确定性匹配。这通常是不可行的。然而，如果我们假设可行性，那么该算法是5/9竞争的[HT19]。事实上，它实际上是Kalyanasundaram和Pruhs [KP00]的2-匹配算法，每个在线顶点都有两个副本，并且允许离线顶点匹配两次。

通过引入一种名为在线相关选择(OCS)的算法成分，我们肯定地回答了这个问题，该算法允许我们量化随机轮数之间的负相关性。

5.5Edge-Weighted在线匹配

本节提出了一种在线原对偶算法来解决边加权在线二部图匹配问题。该算法使用一个soc -OCS作为黑盒，其竞争比率取决于soc的值。对于第3节中所描述的(1/16)，则为0.505-competitive，对于定理3中所描述的(1/16)，则为0.5086-competitive，这证明了我们关于边缘加权在线匹配的主要结果。

5.6在线相关选择

本节介绍我们的在线算法的新组成部分，我们认为这是广泛适用且具有独立利益。为了激发这种技术，考虑以下想法,实验在非加权在线匹配的情况下，即wi,j∈{0,1}对任意i∈L和任何j∈R。确定性贪婪。首先回顾为什么所有确定性贪婪算法都匹配每一个,在线顶点到一个不匹配的离线邻居是在大多数1/2竞争。考虑一个实例,有两个离线顶点和两个在线顶点的图。第一个联机顶点与这两个顶点相邻离线顶点，算法确定地选择其中一个。第二个在线顶点，但是只与先前匹配的顶点相邻。

具有独立随机位的二选择贪婪。我们可以通过随机匹配第一个联机顶点，将预期匹配大小从1提高到1.5。本着这种精神，考虑以下两种选择贪婪算法。当在线顶点到达时，确定最不可能匹配的邻居（在前几轮随机性的基础上）。如果有不止一个这样的邻居，选择任意两个，例如，按字典顺序，并与一个匹配新的随机位。否则，确定地匹配到最不匹配的邻居。我们指的是前者为随机轮，后者为确定性轮。因为每轮随机使用新的随机位，这相当于匹配在随机轮数最少，无确定轮数。不幸的是，这个算法由于上三角图的存在，所以也有1/2-竞争性。

完全负相关的两个选择贪婪。这个思想中的最后一个算实验是两个选择贪婪的一个假想变体，它完美地负相关随机取整，使每一个离线顶点在成为候选点后都有确定的匹配随机分两轮。这在一般情况下是不可行的。不过，如果我们假设可行,那么这个算法是5/9竞争的[HT19]。实际上，它是有效的2-匹配算法Kalyanasundaram和Pruhs[KP00]，每个在线顶点有两个副本，允许离线要匹配两次的顶点。

5.7总结

针对竞争系数为0.5086的边加权二分图匹配问题，提出了一种在线原对偶算法，解决了在线算法研究中长期存在的一个开放问题。特别地，这项工作合并和改进了Fahrbach和Zadimoghaddam [FZ19]以及Huang和Tao [HT19, Hua19]的结果，在在线原对偶框架下给出了一个更简单的算法。我们的工作开启了在线相关选择的研究，这是量化在线作业问题负相关水平的关键算法成分，相信这一技术将在其他在线问题中得到进一步的应用。使用独立的随机位进行选择会产生一个0-OCS(无负相关)，而使用一个完全负相关的假想的1-OCS会导致分配不一致。因此，我们的目标是设计一种使用部分负相关的在线匹配算法。我们首先构建a1/16 ocs，然后优化这个子程序以获得0.109927-OCS。设计一个最大可能面∈[0,1]的lu -OCS本身是一个有趣的开放问题，它将直接导致边缘加权在线二部匹配问题的竞争比的提高。然而，即使存在1- ocs，使用此方法可以实现的最佳竞争比最多为5/9，如附录b所示。因此，我们需要全新的思路，以便在未加权和顶点加权情况下更接近最佳的1- 1/e比率。一种可能的方法是考虑一个在每一轮中允许两个以上候选人的OCS，称之为多路OCS。我们将这个问题作为另一个有趣的开放问题留作以后的工作

六、自我实践

对于三篇论文的研读让我意识到必须得从最基础的内容开始学起，下面是对相关细节的学习记录。

6.1匹配的概念

1. 定义：给定一个二分图G，在G的一个子图M中，M的边集{E}中的任意两条边都不依附于同一个顶点，则称M是一个匹配。  
   匹配点：匹配边上的两点
2. 极大匹配(Maximal Matching)：是指在当前已完成的匹配下,无法再通过增加未完成匹配的边的方式来增加匹配的边数。
3. 最大匹配(maximum matching)：是所有极大匹配当中边数最大的一个匹配,设为M。选择这样的边数最大的子集称为图的最大匹配问题。
4. 完美匹配（完备匹配）：一个图中所有的顶点都是匹配点的匹配，即2|M| = |V|。完美匹配一定是最大匹配，但并非每个图都存在完美匹配。
5. 最优匹配：最优匹配又称为带权最大匹配，是指在带有权值边的二分图中，求一个匹配使得匹配边上的权值和最大。一般X和Y集合顶点个数相同，最优匹配也是一个完备匹配，即每个顶点都被匹配。如果个数不相等，可以通过补点加0边实现转化。一般使用KM算法解决该问题。（KM（Kuhn and Munkres）算法，是对匈牙利算法的一种贪心扩展。）
6. 最小覆盖  
   二分图的最小覆盖分为最小顶点覆盖和最小路径覆盖：

①最小顶点覆盖是指最少的顶点数使得二分图G中的每条边都至少与其中一个点相关联  
注：二分图的最小顶点覆盖数=二分图的最大匹配数

②最小路径覆盖也称为最小边覆盖，是指用尽量少的不相交简单路径覆盖二分图中的所有顶点。  
注：二分图的最小路径覆盖数=|V|-二分图的最大匹配数

1. 最大独立集  
   最大独立集是指寻找一个点集，使得其中任意两点在图中无对应边。对于一般图来说，最大独立集是一个NP完全问题，对于二分图来说最大独立集=|V|-二分图的最大匹配数。最大独立集S 与 最小覆盖集T 互补

如果在图G左右两边加上源汇点后，图G等价于一个网络流，二分图最大匹配问题可以转为最大流的问题。解决此问题的匈牙利算法的本质就是寻找最大流的增广路径。  
 注意:匈牙利算法，除了二分图多重匹配外在二分图匹配中都可以使用。  
 二分图匹配中还有一个hk算法，复杂度为o（sqrt（n）\*e）复杂度降低较低，代码量飙升。

交替路：从一个未匹配点出发，依次经过非匹配边、匹配边、非匹配边…形成的路径叫交替路。

增广路：从一个未匹配点出发，走交替路，如果途径另一个未匹配点（出发的点不算），则这条交替路称为增广路（agumenting path）。

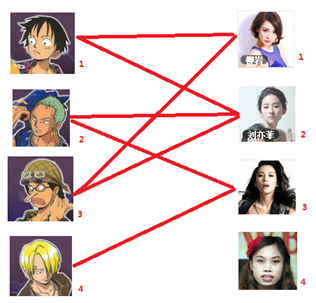
增广路有一个重要特点：非匹配边比匹配边多一条。因此，研究增广路的意义是改进匹配。只要把增广路中的匹配边和非匹配边的身份交换即可。由于中间的匹配节点不存在其他相连的匹配边，所以这样做不会破坏匹配的性质。交换后，图中的匹配边数目比原来多了 1 条。

我们可以通过不停地找增广路来增加匹配中的匹配边和匹配点。找不到增广路时，达到最大匹配（这是增广路定理）。匈牙利算法正是这么做的。

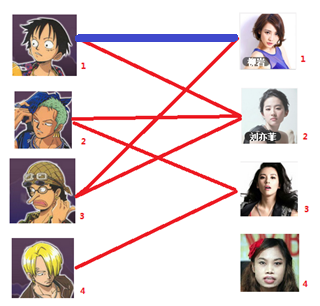
匈牙利算法解释：初始时最大匹配为空，while 找得到增广路径，do 把增广路径加入到最大匹配中去

（注：匈牙利算法虽然根本上是最大流算法，但是它不需要建网络模型，所以图中不再需要源点和汇点，仅仅是一个二分图。每条边也不需要有方向。）

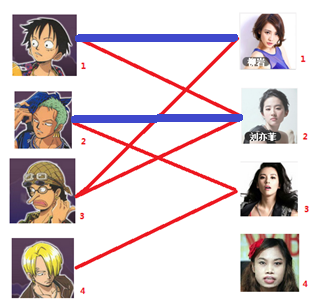
匈牙利树一般由 BFS 构造（类似于 BFS 树）。从一个未匹配点出发运行 BFS（唯一的限制是，必须走交替路），直到不能再扩展为止。（注：匈牙利树中所有叶子节点均为匹配点）

通过数代人的努力，你终于赶上了剩男剩女的大潮，假设你是一位光荣的新世纪媒人，在你的手上有N个剩男，M个剩女，每个人都可能对多名异性有好感（惊讶-\_-||暂时不考虑特殊的性取向），如果一对男女互有好感，那么你就可以把这一对撮合在一起，现在让我们无视掉所有的单相思（好忧伤的感觉快哭了），你拥有的大概就是下面这样一张关系图，每一条连线都表示互有好感。  


·想要尽可能地撮合更多的情侣，匈牙利算法的工作模式会教你这样做：

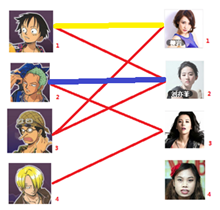
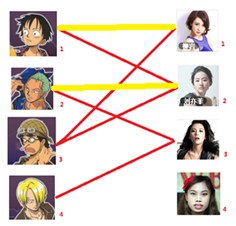
一： 先试着给1号男生找妹子，发现第一个和他相连的1号女生还名花无主，got it，连上一条蓝线  


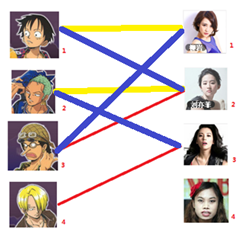
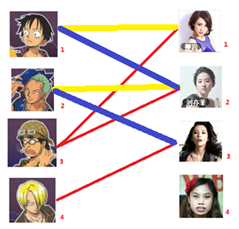
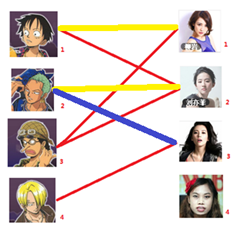
二：接着给2号男生找妹子，发现第一个和他相连的2号女生名花无主，got it

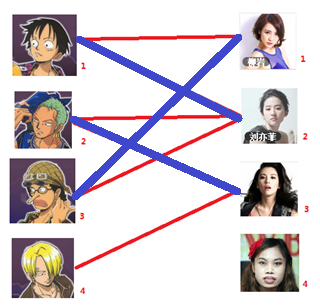


三：接下来是3号男生，很遗憾1号女生已经有主了，怎么办呢？

这时试着给之前1号女生匹配的男生（也就是1号男生）另外分配一个妹子。

(黄色表示这条边被临时拆掉)  
  
与1号男生相连的第二个女生是2号女生，但是2号女生也有主了，怎么办呢？我们再试着给2号女生的原配(发火发火)重新找个妹子(注意这个步骤和上面是一样的，这是一个递归的过程)  
  
此时发现2号男生还能找到3号女生，那么之前的问题迎刃而解了，回溯回去

2号男生可以找3号妹子1号男生可以找2号妹子了3号男生可以找1号妹子  
  
所以第三步最后的结果就是：



四： 接下来是4号男生，很遗憾，按照第三步的节奏我们没法给4号男生腾出来一个妹子。这就是匈牙利算法的流程，其中找妹子是个递归的过程，最最关键的字就是“腾”字

其原则大概是：有机会上，没机会创造机会也要上（在论文中也隐含这一算法）

**6.2Kuhn-Munkers算法流程**

(1)初始化可行顶标的值  
(2)用匈牙利算法寻找完备匹配  
(3)若未找到完备匹配则修改可行顶标的值  
(4)重复(2)(3)直到找到相等子图的完备匹配为止

Kuhn-Munkers算法的几种变形应用

1.要求最小权完备匹配只需将所有的边权值取其相反数，求最大权完备匹配，匹配的值再取相反数即可。

2.Kuhn-Munkers算法的运行要求是必须存在一个完备匹配，如果求一个最大权匹配(不一定完备)可以把不存在的边权值赋为0。

3.若要边权之积最大则每条边权取自然对数，然后求最大和权匹配，求得的结果a再算出e^a就是最大积匹配。

七、总结

上了人工智能创新设计这门课，在老师的指导下研究了在线二分匹配，花费了不少时间，还是似懂非懂，感谢张鹏老师的辛勤指导，感叹算法的巧妙，还需努力，探寻更多的知识。