UCSD-基因组测序

1 基因组测序

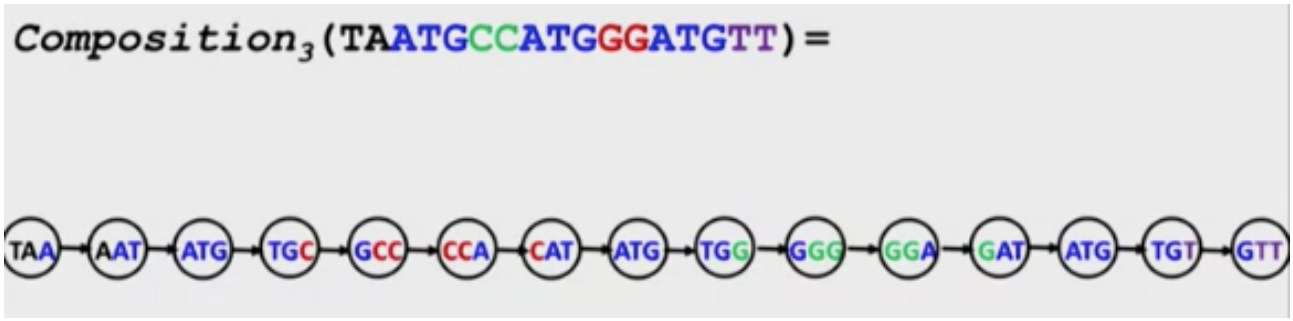
1.1 基因组测序背后的算法原理

已知测序是将待测DNA序列扩增，然后打碎成较小片段，然后将片段重新拼贴在一起。但是片段的位置信息未知，那么，我们该如何将这些DNA片段拼贴回原基因组呢?
用计算机语言描述这个问题:
输入：包含k-mers的集合
输出：一个基因组，使它的k-mers组成与输入的k-mers集合相同。

需要设计一个算法来解决如何将DNA碎片拼贴回完整基因组。

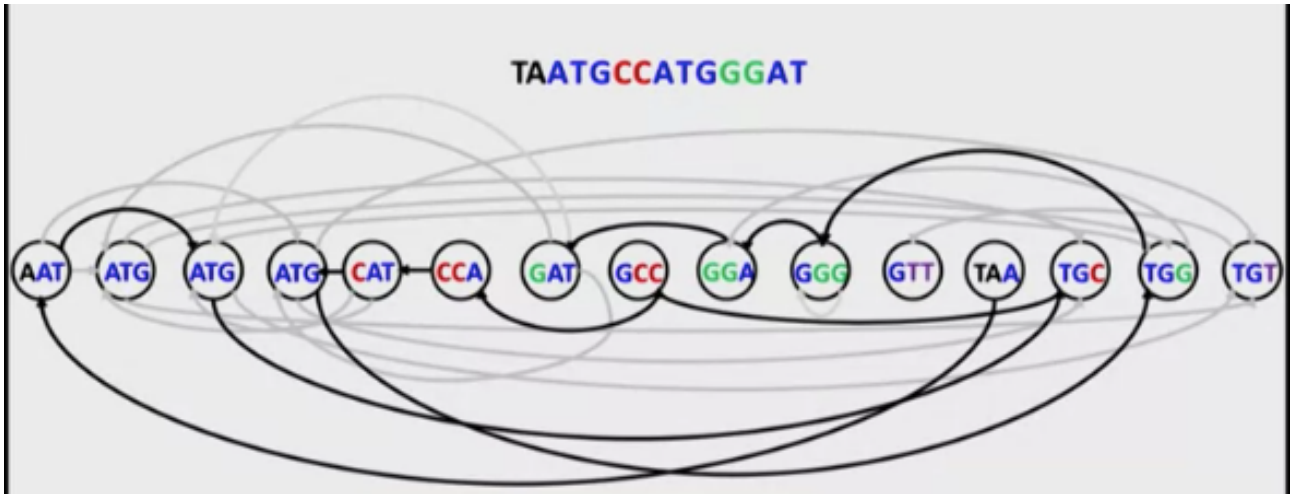
【柯尼斯堡七桥问题】
不重复地遍历城市中的7座桥，该如何设计路线?
↓
这是图论中的著名问题。欧拉解决了这一问题，将桥视为线，被桥连接的地区视为由线连接的点，这样若从某点出发后最后再回到这点，则这一点的线数必须是偶数，这样的点称为偶顶点。相对的，连有奇数条线的点称为奇顶点。欧拉论述了，由柯尼斯堡七桥问题中存在4个奇顶点，它无法实现符合题意的遍历。

1.1.1 String Reconstruction as a Hamiltonian Path Problem



将这个DNA打碎为3-mer片段，每一个3-mer片段就是一个节点，基因组就是通过每个节点的路径。
我们能否在不知道全基因组的情况下，画出这个路径?

↓
可以。把重叠(k-1)mer的节点连接在一起。
但是，可能有多个节点都可以重叠在一起！同时，这些片段在基因组上的位置是未知的，如果把节点重排一下，问题会变得非常复杂。

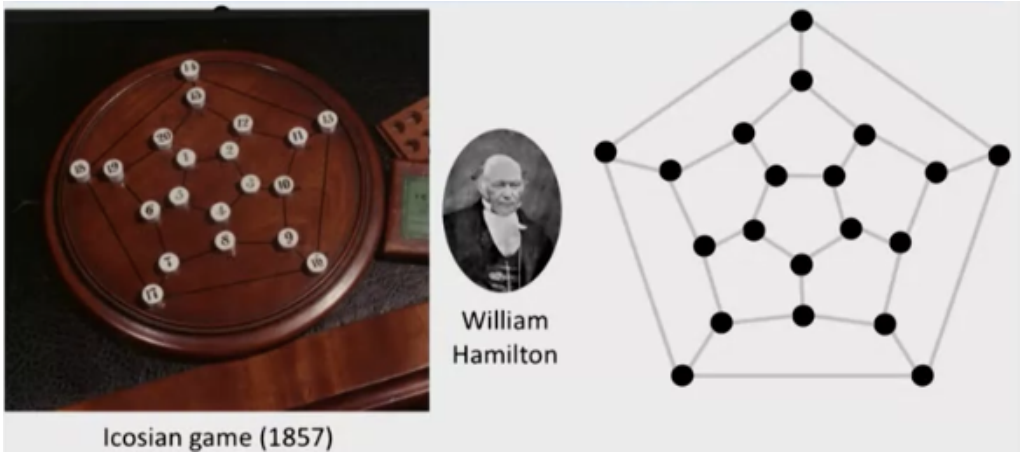


基因组实际上就是穿过这些节点的路径。
在这个预设下，我们需要结局的问题是：找到一条路径，遍历这些节点，但每个节点仅穿过一次。
我们把这个路径称为汉密尔顿路径 Hamilton Path

Hamilton Problem

Hamiltonian Path Problem. Find a Hamiltonian path in a graph.
- **Input.** A graph.
- **Output.** A path visiting every **node** in the graph exactly once.

汉密尔顿问题：给定一张图，我们从图中找出一条路径，让它有且仅有一次经过每个节点。

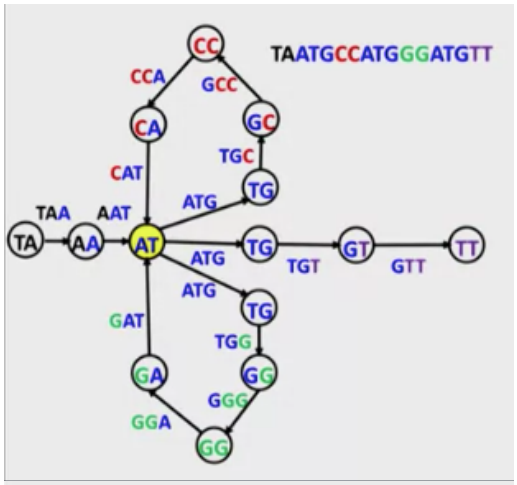


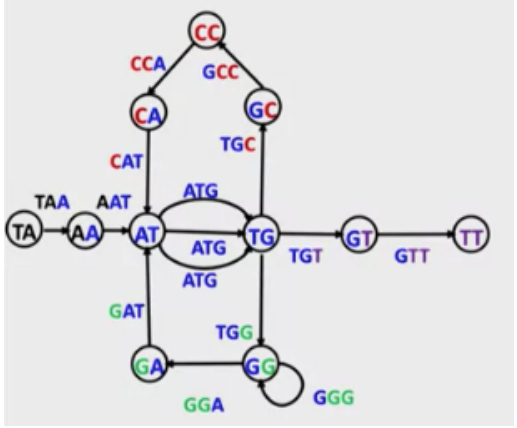
但是，汉密尔顿问题只适用于无向图，如何应用到基因组拼贴中呢?

1.1.2 String Reconstruction as an Eulerian Path Problem

改变一下节点与边的定义方法。在Hamilton Path Problem中，我们把3-mer作为节点；但在Eulerian Path Problem中，我们把3-mer作为边，节点是3-mer的前两个与后两个碱基。

然后，我们把相同的节点放在一起，使他们形成图中的定点（比如把3个AT节点放在一起）





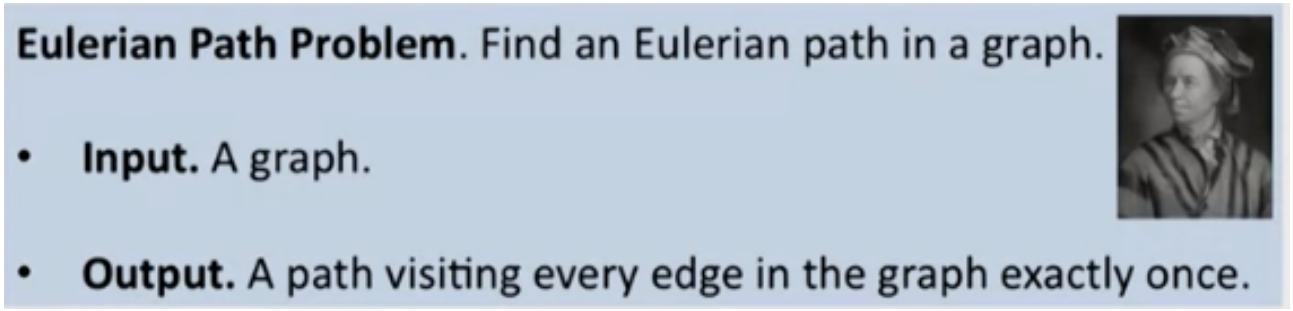
然后把3个TG节点放在一起

把2个GG节点放在一起

现在得到：**字符串的de bruijn图**

现在需要解决的问题：
找到一条路径，遍历所有边，但每条边只经过一次。
现在构建出了 欧拉路径

欧拉路径问题：经过图中每个边有且仅有一次的路径。



但是，Hamilton Path Problem是七个未解决的千禧年问题之一，所以，解决问题的重点会放在Eulerian Path Problem上。

2 Applying Euler's Theorem to Assemble Genomes

2.1 de bruijn graphs

给定一个基因组，我们可以构建出de bruijn graph。但是，在基因组测序时，基因组是未知的，已知的只有k-mers和k-mers的读序，能否在此情况下构建出de bruijn graph?

即使未知原基因组，也能通过打断的片段重新作出de bruijn graph

欧拉图——点被边连接在一起，进节点的次数等于出节点的次数，每条边经过有且仅有一次——称这张图是”平衡的“。
欧拉证明了：每张欧拉图都是平衡的；每张平衡的图都是欧拉图。
可以用这个算法来实现：找出欧拉环！（我们只可能在起始节点上被阻滞住，一旦发现被阻滞，就尝试新的节点。最终将多条路线组合为欧拉路径）

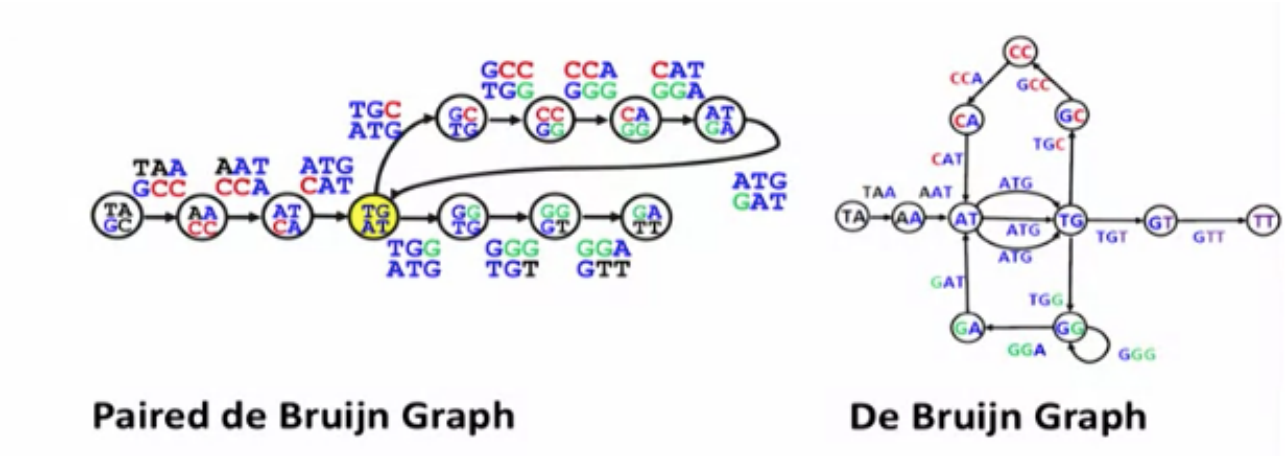
```
EulerianCycle(BalancedGraph)
  form a Cycle by randomly walking in BalancedGraph (avoiding already visited edges)
    while Cycle is not Eulerian
      select a node newStart  in Cycle with still unexplored outgoing edges
      form a Cycle' by traversing Cycle from newStart and randomly walking afterwards
      Cycle ← Cycle'
  return Cycle
```

但是！在de bruijn graph中可能有多条欧拉路径！
为了解决这一问题，需要对de bruijn graph进行改进 → 成对de bruijn graph

成对de bruijn graph（paired de bruijn graph）

1. 将成对的k-mers放在一起，每个成对的k-mers表示为一条边，这条边从成对的前缀指向后缀（成对的k-mers表示，两个k-mers之间相距为d）
2. 把同样标签的节点黏在一起

由此得到的成对的de pruijn graph里的欧拉路径会减少一些。



但实际基因组测序的情况下：①读序未知，②k-mers间距未知，③读序无法将序列完美覆盖（没有完美的基因组覆盖率）

3 sequence antibiotics

Antibiotic：由真菌或细菌产生，可以进入并杀死细菌的物质，本质是一种多肽

3.1 How do bacteria produce antibiotics?

中心法则并不是在所有情况下都适用！

一些抗生素属于“非核糖体多肽”，即使没有核糖体，这些抗生素也可以被生产出来。后来发现，它们不是由核糖体通过RNA翻译合成的，比如环状短杆菌酪肽B1（Tyrocidine B1），是经过完全不同的“NRP合成酶”过程合成出来的。

所以，在DNA上可能无法找到这些抗生素对应的序列，因为它们不是基于基因组产生的。

对蛋白质/多肽测序方法——质谱仪

分子质量——道尔顿 Dalton（Da）（一般取整数）

得到了氨基酸的道尔顿质量表：

Contains masses of all 20 amino acids

G	A	S	P	V	T	C	I	L	N	D	K	Q	E	M	H	F	R	Y	W
57	71	87	97	99	101	103	113	113	114	115	128	128	129	131	137	147	156	163	186

环肽测序问题：从质谱信息重构环状多肽

3.2 Algorithm for Cyclopeptide Sequencing

算法1：（暴力破解算法）

通常已知tyrocidine B1的总分子质量为1322

①先产生每一个可以具有这个质量的多肽分子序列

②然后产生这些多肽的分子质谱

③寻找能和实际tyrocidine B1质谱匹配的多肽分子质谱

穷举法 非常耗时！可能性极其多

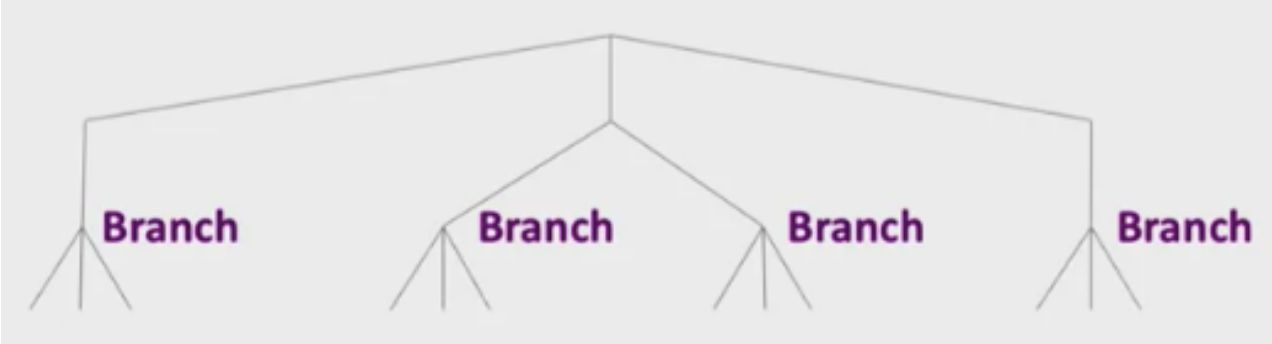
↓

Solution：避免一次性产生大的多肽序列，将大序列分解成小序列，从小一些的序列组装出候选答案；在每个阶段检查其质谱，质谱不正确的序列将被排除——由此限制可能的多肽序列的个数！

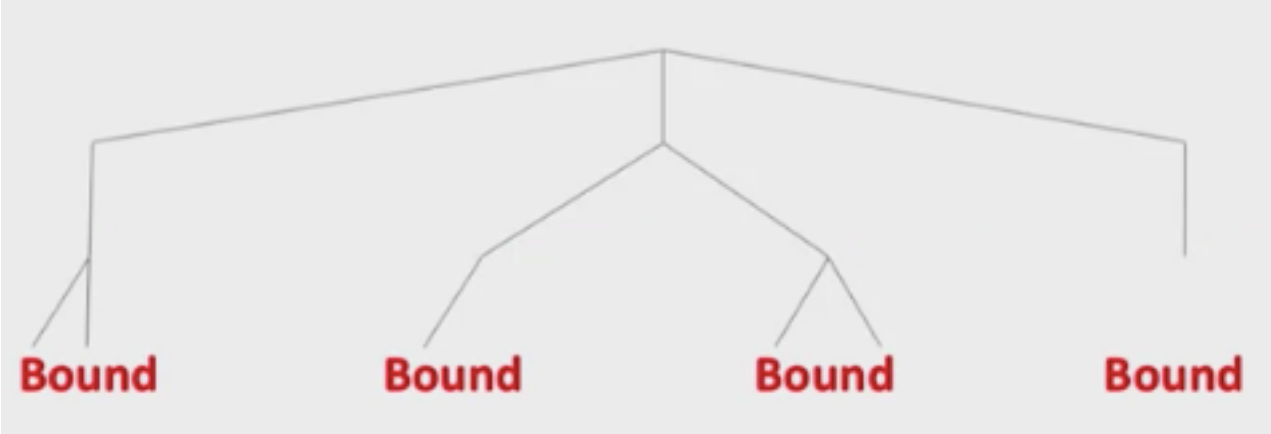
算法2：Branch and Bound 分支定界算法

Branch and band algorithm思想：

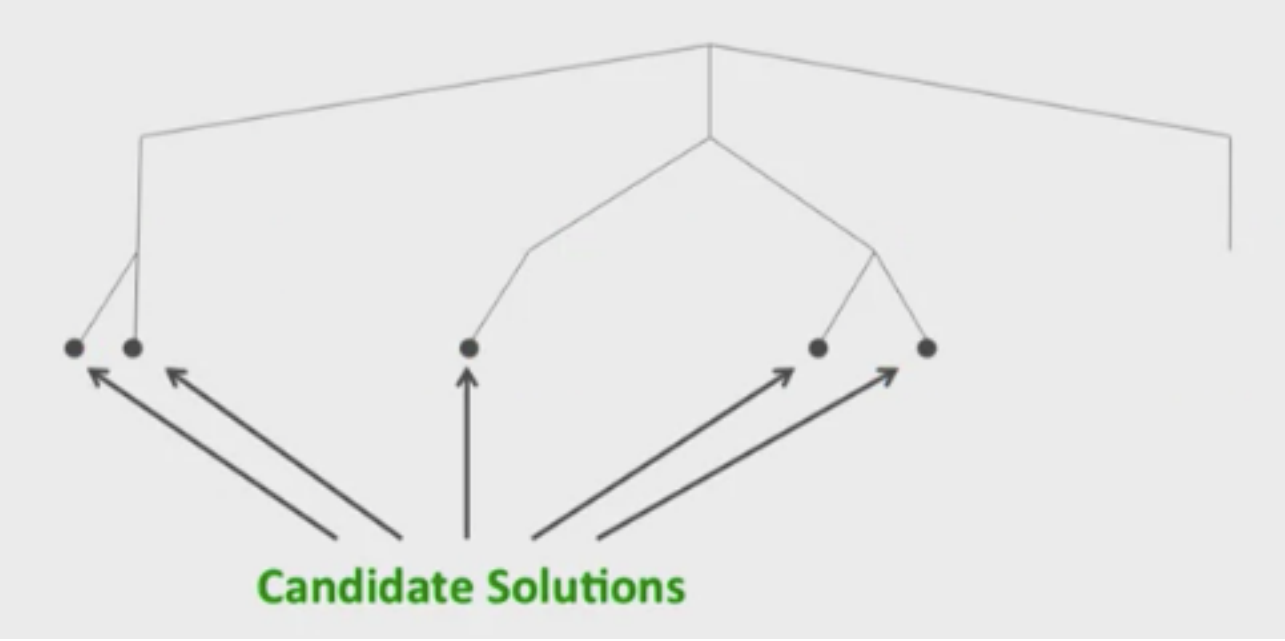
为了解决问题，我们找出了一些可能的分支方案，然后通过定界，将一些分支排除出去，剩下的一些方案还存在于这棵树的节点上。



先分支，再定界

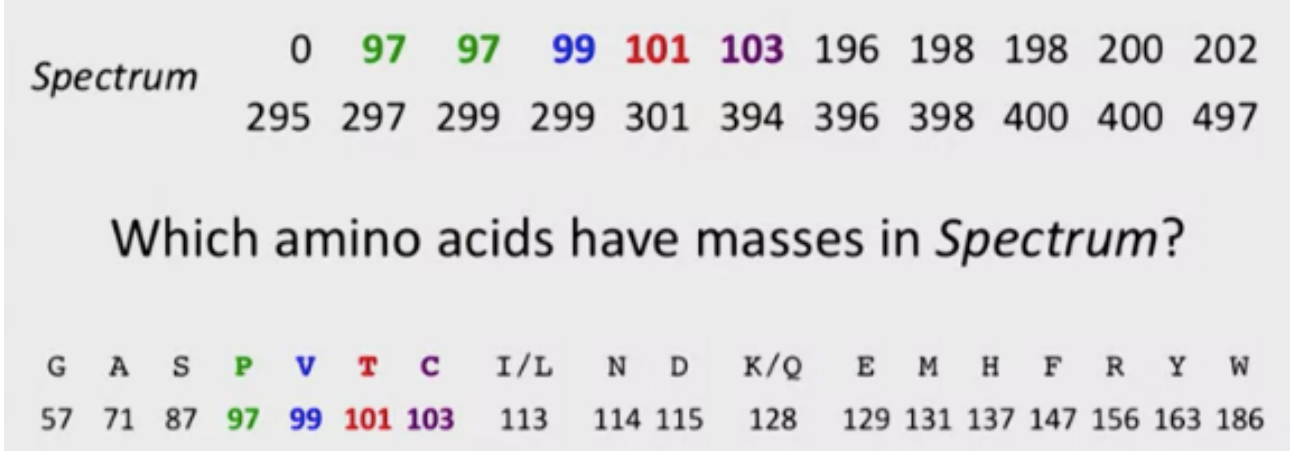


然后得到一些备选方案：



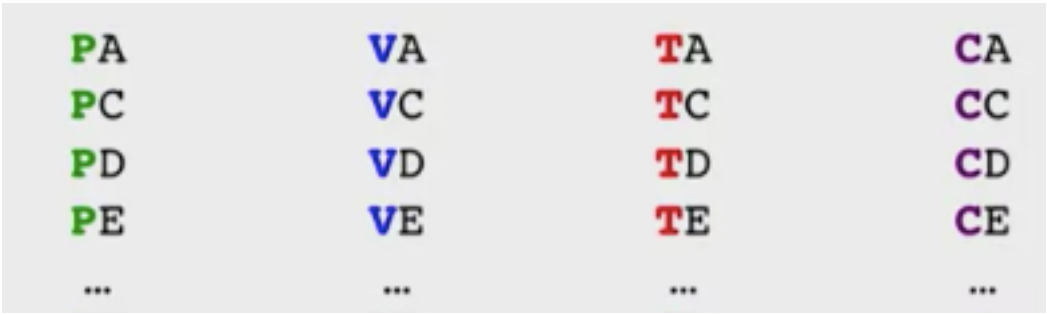
因此，我们得到了tyrocidine B1的质谱图：

Spectrum	0	97	97	99	101	103	196	198	198	200	202
	295	297	299	299	301	394	396	398	400	400	497

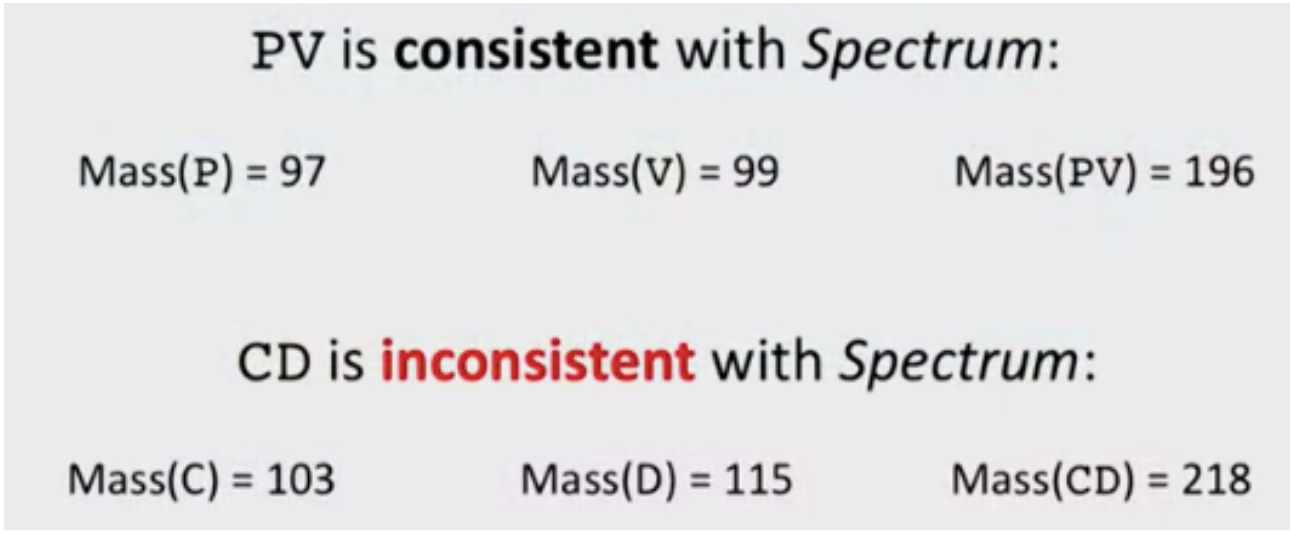


然后发现，在质谱图中出现了 P V T C这四种氨基酸的单独分子质量数据

之后，将P V C T延伸成具有两个氨基酸的短序列：



然后进行“定界”，对“分支”进行修剪：



例如，P V PV 都在质谱图中出现了；而尽管C出现在了质谱图里，但D CD并未出现，因此可以将CD序列修剪掉。（CD D不吻合质谱图）

“吻合”是指：把这个多肽剪成片段，称量每个片段的分子质量，所有的片段的分子质量在质谱图中都可以找到。

然后，如上步骤再延伸至3个氨基酸的短序列，判断是否吻合质谱图，再进行修剪。

该算法步数比较多，也会较为复杂。

问题是！实际质谱图和理论质谱图并不完全相同！理论质谱图对应着完全正确的分子质量，而实验质谱图则可能会对应着一些错误的分子质量，实验质谱图中可能还会有缺失的分子质量。

Theoretical:	0		113	114	128	129	227	242	242	257		355	356	370	371	484
Experimental:	0	99	113	114	128		227			257	299	355	356	370	371	484

所以，必须找出新的办法。我们不再要求多肽短序列与质谱图完全匹配/吻合，而是使用评分来评估多肽短序列与质谱图匹配/吻合的程度。然后保留匹配程度前n名（及并列）的序列，使它们留在列表中，进入下一轮分支定界，然后是又一轮的打分，得分高的前n名多肽（及并列）进入列表，排名更新。

算法3：排行榜环状多肽测序法 Leader Board Cyclopeptide Sequencing

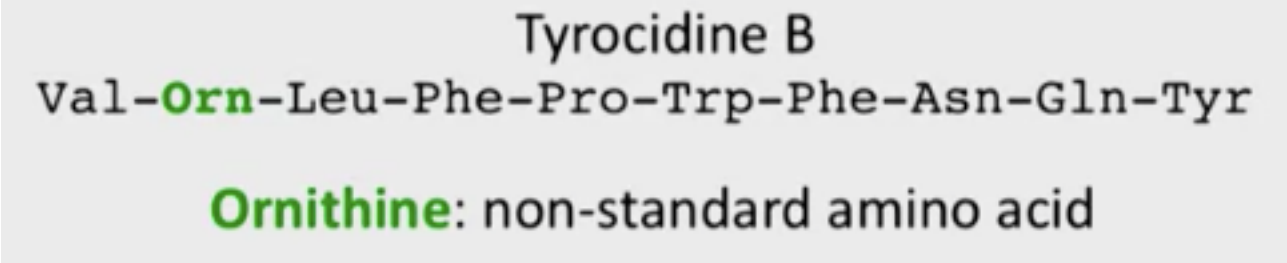
1. Add “0-peptide” to *Leaderboard* as *LeaderPeptide*.
2. **Extend** each peptide in *Leaderboard* by each of 18 different amino acid masses.
3. **Cut** low-scoring peptides from *Leaderboard*. (Keep “top N with ties”)
4. Update *LeaderPeptide* if there is a higher scoring peptide in *Leaderboard* with mass = parent mass.
5. Eliminate all peptides with mass > parent mass.
6. Iterate 2-5 until *Leaderboard* is empty.
7. Return *LeaderPeptide*.

Warning！这是一个启发算法（heuristic）！

在开始阶段，我们可能已经把正确的多肽（尽管最初得分低）淘汰掉了。

当质谱图的false/missing mass概率逐渐变大（质谱图噪音变多），更有可能得到错误的多肽序列。

这些非核糖多肽（由NRP合成酶途径生成）实际上含有非标准氨基酸，这些氨基酸在遗传密码之外。



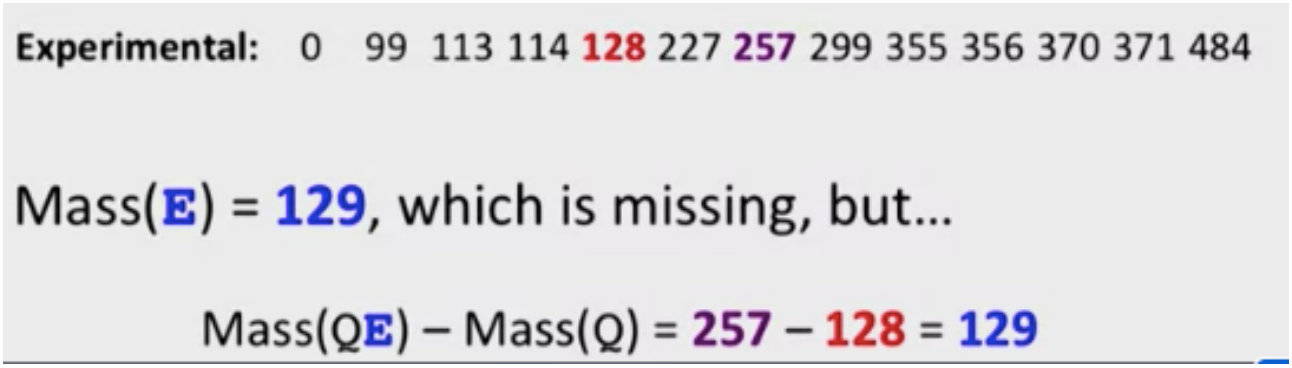
比如这里就含有一个鸟氨酸（非标准氨基酸）

由于有多种非标准氨基酸，所以我们需要扩展质谱图的分子质量表格，把非标准氨基酸及其质量也纳入表中。

↓

这就导致，算法会把非标准氨基酸纳入考虑范围，即使真实序列中没有非标准氨基酸，算法得到序列中也可能会出现非标准氨基酸！

那么，我们需要从质谱图中进一步挖掘信息。



比如说，E的分子质量并未在直观的质谱图中出现，但是，如果将QE和Q相减，则可以得到E的分子质量。

引入“质谱卷积” The Spectral Convolution

Spectral convolution: positive difference between every pair of masses in spectrum.

	""	false	L	N	Q	LN	QE	false	LNQ	ELN	QEL	NQE
	0	99	113	114	128	227	257	299	355	356	370	371
0												
99	99											
113	113	14										
114	114	15	1									
128	128	29	15	14								
227	227	128	114	113	99							
257	257	158	144	143	129	30						
299	299	200	186	185	171	72	42					
355	355	256	242	241	227	128	98	56				
356	356	257	243	242	228	129	99	57	1			
370	370	271	257	256	242	143	113	71	15	14		
371	371	272	258	257	243	144	114	72	16	15	1	
484	484	385	371	370	356	257	227	185	129	128	114	113

质谱卷积是质谱图上每一对分子质量差的绝对值。分析这份新的质谱图，找出出现频率最高的几个分子质量（57-200之间）与它们所对应的氨基酸：

99	113	114	128	129
V	L	N	Q	E

算法4：卷积环状多肽测序 Convolution Cyclopeptide Sequence

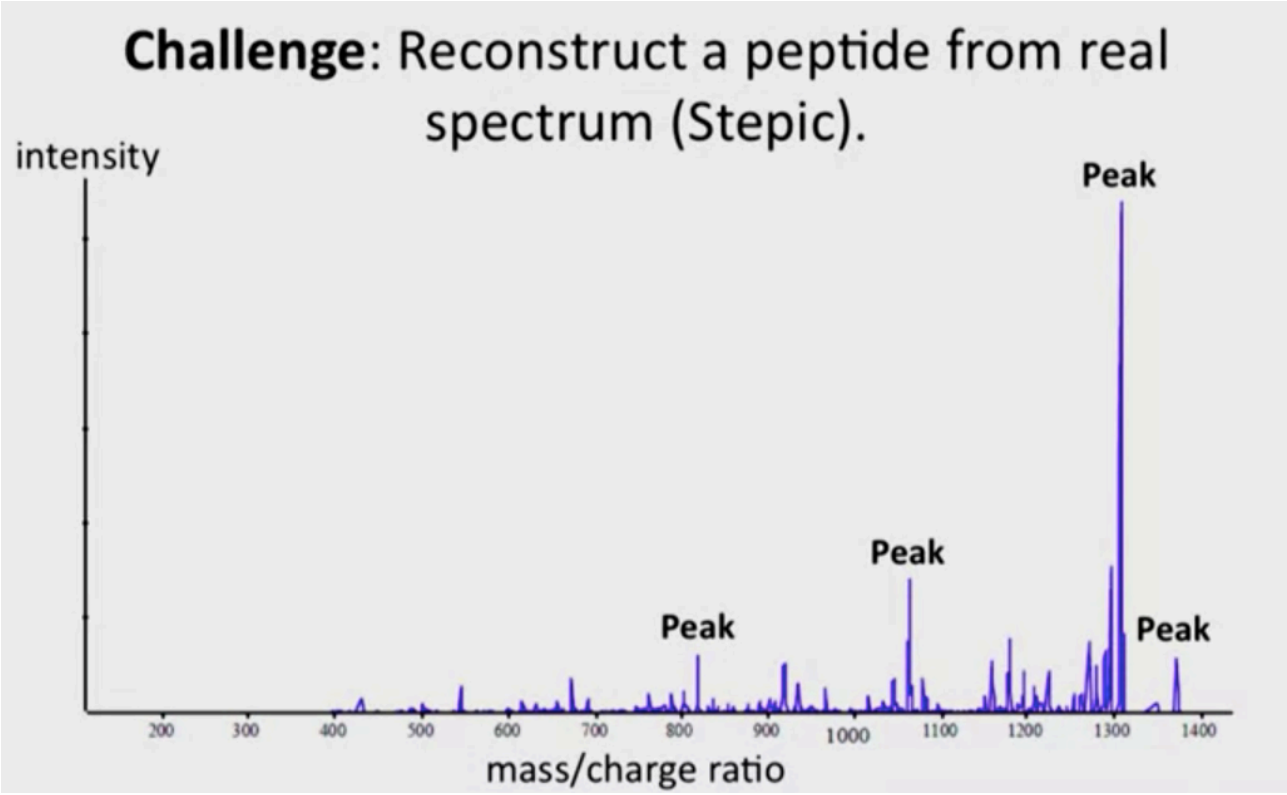
1. Form spectral convolution of spectrum.
2. Take the *M most frequent* elements in the convolution (between 57 and 200).
3. Run **LeaderboardCyclopeptideSequencing**, forming peptides only on these *M* integers.

①拿到一份实验质谱图，然后产生它的质谱卷积

②然后查看卷积质谱图，找出出现频率最高的元素，在质量为57-200之间的元素，选取出现频率最高的前M个

③运行排行榜算法

但是，我们在实际中所取得的质谱图与理论质谱图并不相同。①实际质谱图噪声更大，②实际质谱图显示的多肽片段的质荷比



我们需要将这个图像转变为质谱数据列表，在这个过程中，可能引入更多的错误/噪声。因此，算法仍然需要改进，以适应真实的实验数据。