目前为止，测序环肽的算法依赖于分支界限法（ “Branch and Bound Algorithm” ）来做。这些算法会从空串加薪的氨基酸。如果新的候选多肽不符数据，这些候选多肽被丢弃。所还存在的候选多肽再加新的氨基酸，再丢弃不符数据的多肽等等。如果大部分的数据是假的，也所真的数据太少，分支界限法不能测序环肽。因此，我们需要使用其他的算法来做。我这里要描述我正在研究的新算法。这个算法有几个步骤。

首先，我们需要预计全环肽的重量。这个步骤有两个部分。第一，我们可以把所有的数据点加上其他的数据点。如果这两个数据点算是全多肽的两个部分，他们综合的重量等于全环肽的质量。因为几个真的数据点确实有配对，全多肽的重量也会常出现。这个部分会提供几个候选全环肽重量。第二，我们可以使用氨基酸的重量来使标准化好的候选多肽和修剪不好的候选多肽。如果一个数据点只差一个氨基酸的重量来算候选多肽的重量，候选多肽的可能性比较高。把这些数据点和差的氨基酸重量加在一起的时候可以使标准化候选多肽的重量。因此，这个步骤最终有几个候选多肽和跟这些候选多肽有关系的氨基酸。最终结果之一应该跟真正的全多肽重量的差别不超过0.2道尔顿（Da）。

其次，分析环肽的结构。一下是一个饼图，这个饼图代表Tyrocidine B1的环肽。我有Tyrocidine B1的质谱分析数据。不同的颜色代表不同氨基酸的重量。因为有些氨基酸是重复的还是有两个同重氨基酸，同样重量的氨基酸有同样的颜色。全环的重量是1322.7Da左右，我使用了长度来代表重量。

为了好好的分析以上的环肽，我们需要总不同的角度来看这个问题。我发现重视环肽之内的链接位点很有意思。我在以下的图预算了每一个链接位点。从这个角度来看，每一个数据点代表两个链接位点。例如，数据点405.19（NQY,或者114.04+128.09+163.06）代表第7个和地10个链接位点。

两个链接位点的配对有两个有关系的数据点。在以上的NQY例子之内，NQY的相反序列（VKLFPWF，917.5）也会代表第7和第10个链接位点。幸亏预计每一个数据点的相反序列很容易。（数据点1=全环肽-数据点2）

测序环肽的算法需要分析与比较链接位点之间的长度。两个隔壁的链接位点之间长度等于一个氨基酸的重量。因此，如果之前预计的几个氨基酸正确，它们代表隔壁的链接位点。例如，V代表链接位点1和10，P代表链接位点4和5.

比较小的数据点更有可能是比较大数据点的两端之一。我们可以使用这个关系来计算其他链接位点之间的长度。例如，上面的缬氨酸（Valine, V）代表第1和第10个链接位点。在我们的数据当中，几个数据点的隔壁、数据点的一个端是V。如果一个数据点的一端是V，那个数据点的相反是V的隔壁。VKLFPWF（链接位点10、7）片段的一端是V。VKLFPWF的相反片段NQY（也是链接位点10、7）不包括V，但是它在V的旁边。此外，如果把NQY加V上，结果NQYV（链接位点1、7）也是一个真的数据点。因此，如果把VKLFPWF的片段“换”成相反的片段然后“加”上V，生物信息学家能够得到另外一个数据点（NQYV）。这两片段之间的关系很有意义。它们之间重复V，但是它们在跟V没有关系的链接位点结合（链接位点7）。

这个“换与加”的方式很有意思。如果有一个特定的氨基酸，生物信息学家们能够得到跟这个氨基酸有关系的几个“换与加”的数据点配对。这些配对包括几个假阳性，但是所有的真阳性代表一个独特的新链接位点。以上例子的配对是VKLFPWF-NQYV的配对，换言之917.4-504的配对。这个配对代表第7个链接位点。这个配对被发现的原因就是因为V的两个链接位点（1、10）都有一个数据点让他们结合在不同的链接位点 （有代表1、7和10、7的链接位点）。在我的数据里面，V的几个“换与加”配对包括代表2、6、8、9的配对。如果我们把这几个配对放在合适的方向，我们可以重新制造包括1、2、4、6、7、10链接位点的环肽。换言之，我们可以只在全10个片段环肽的7个片段。

我的算法的目的是使用每一个预测的氨基酸做一个“换与加”的分析，预测环肽的序列。我没有谈到这个理论的几个苦难。例如，如果一个氨基酸是两次出现的，两个不同类型的“换与加”配对也会出现。此外，把这些配对放在正确的方向并不容易。总的来说，我目前的代码是我解决这些问题的努力。我的几个结果有成功，但是我还在努力研究这个算法。

我的Tyrocidine B1数据，加黑的数字是真的数据点：

372.2, 397.2, 402.0, **406.3**, 415.1, **431.2**, **448.3**, 449.3, 452.2, 471.3, **486.3**, **488.2**, 500.5, **505.3**, 516.1, 536.1, **544.2**, **545.3**, 562.5, 571.3, 599.2, 614.4, 615.4, 616.4, 618.2, **632.0**, 655.5, 656.3, **672.5**, **673.3**, 677.3, **691.4**, **692.4**, 712.1, 722.3, **746.5**, 760.4, 761.6, 762.5, **771.6**, 788.4, 802.3, 803.3, 818.5, **819.4**, 831.4, **836.3**, 853.3, 875.5, **876.5**, 901.5, 915.9, 916.5, 917.8, **918.4**, **933.4,** **934.7**, **935.5**, 949.4, **966.2**, 995.4, 1015.6, 1027.5, 1029.5, 1031.5, 1044.5, 1046.5, **1061.5**, **1063.4**, **1079.2**, 1083.7, 1088.4, 1093.5, **1096.5**, 1098.4, 1158.5, 1159.5, **1176.6**, 1177.7, 1178.6, 1192.7, **1195.4**, 1207.5, **1210.4**, **1224.6**, 1252.5, 1270.5, 1271.5, 1278.6, 1279.6, 1295.6, 1305.6, 1306.5, 1307.5, 1309.6