Audens - Automatic De Novo Sequencing

Table of Content

Audens - Automatic De Novo Sequencing

Preamble

When Jonas Grossmann approached me for help in a project he was involved in, little did I know that I would continue to work on it for more than a year, eventually doing a semestre work on it. I was always rather interested in biology, so when I got the chance to work in a team doing research in that field, I jumped at it – even though my knowledge in this particular topic was and is only superficial at best. I will admit up-front that a lot of the more biology oriented aspects of this projects have always seemed a bit obscure to me, and this paper will concentrate on the computational aspects of the problem, although in Chapter 2, I will give a short introduction to the biological aspects of this project.

Audens has been in development by various people before I joined the project and took over development of the application. Audens is a team effort. My own rather limited knowledge of proteomics would have been of little help in improving Audens. My improvements to Audens would have never been possible without the work of Sacha Baginsky, Mark Cieliebak, Jonas Grossmann, Matthias Müller and others who have worked on Audens before. And this work is not done by a long shot, either. Others will have to take over work and continue development of Audens to make it even better.

I would like to thank Jonas Grossmann for entrusting me with this task, and I'd like to thank Mark Cieliebak and Sacha Baginsky for accepting me as a member of the team, even though my input on biological questions was certainly minuscule.

Audens - Automatic De Novo Sequencing

1. Abstract

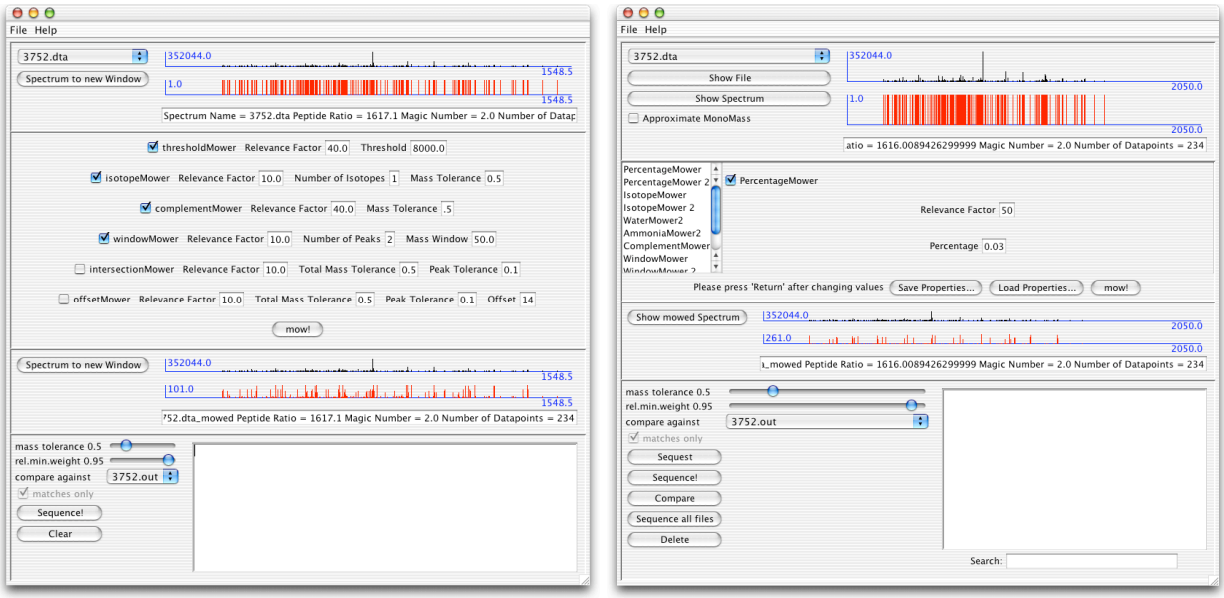
Audens is a de novo sequencing tool for peptides. The name is a shorthand for "Automatic De Novo Sequencing". Audens aims to improve on existing de novo sequencing algorithms. De novo sequencing applications can sequence a peptide from mass spectrometry data without using a database. During this semestre work, several new heuristics for pre-processing mass spectrometry data were implemented. Audens' existing infrastructure was improved and its interface enhanced. Additionally, several separate tools that support Audens or helped us improve Audens were implemented and used.

1.1 General Introduction

Audens is an application written in Java that tries to sequence a peptide from mass spectrometry data. This means that it tries to find out which peptide was actualy being measured by the mass spectrometer, based on the mass spectrometer's output. It does not use a database to do this, but instead only relies on the mass spectrometry data, e.g. the output from the mass spectrometer, and biological rules which it applies to this data.

Since we do not expect the reader of this document to have deep biological knowledge, but rather to have a background in computer science (much like the writer of this paper), a lot of the biological concepts will only be explained in a rather superficial manner. It is our aim to explain the biological concepts in a way which makes sense inside the scope of this project, i.e. the reader should be able to learn enough about the biological concepts to be able to understand the influence they have on the computer science side. To this effect, we have added a chapter explaining some of the terminology of this project, Chapter 7.

Audens - Automatic De Novo Sequencing

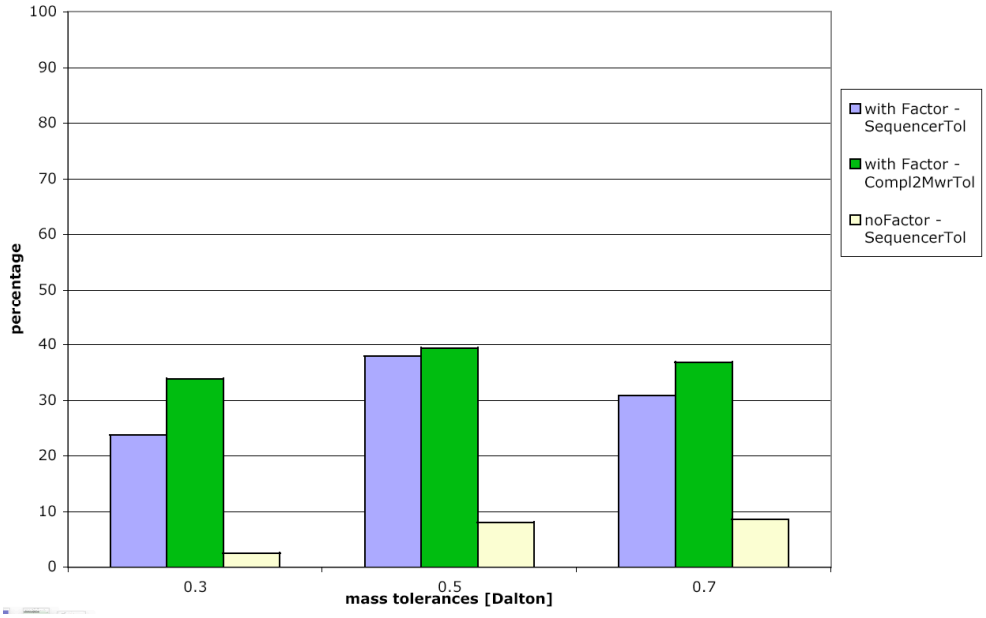


(Pictures of Audens before and after my semestre work)

When I took over development of Audens, it was already in a fairly mature state. Although the results were not as good as they are now, things like the sequencing algorithm, explained in Chapter 3, were already in place and didn't have to be changed for this version of Audens. How much were we able to improve Audens? Since Audens wasn't able to evaluate its own results when I started working on it, I can't give clear numbers comparing the two versions. In Jonas Grossmann's diploma thesis, there's a chapter on Audens' performance[1]. The main point is that especially the second generation mowers (mowers are small application components that try to remove data which decreases the quality of the sequencing, thus making the actual sequencing process produce better results, as explained in Chapter 5.2), and the approximation factor, explained in Chapter 5.1.1, have brought great progress. According to Jonas Grossmann's evaluations of representative spectra[2], there's almost a 40% chance Audens will output the correct solution as one of the first three results. I will talk about Audens' performance from a biology perspective in Chapter 2, but for an in-depth treatise, one needs to look at Jonas Grossmann's diploma thesis.

The following table, taken from Jonas Grossmann's work, shows the result of a test with 200 representative data files. The "percentage" bar shows the number of files whose true solution appeared in Audens' results list. The blue and the green bars, which are from sequencing passes with the approximation factor set, show considerably better results than the yellow bars, where the

approximation factor is not set, independent from the value we set for the mass tolerance that the mowers use. All computations use a certain mass tolerance to offset measurement errors.



During the course of this project, we also created several smaller applications to help the development of Audens. Some of these applications create test data, others analyse existing data or do similar tasks. Some of these applications will be described in Chapter 5.3.

Our main goal, however, was to improve Audens' results. To achieve this, we mainly created better mowers. The better the mowers are, the better the sequencing results are. A lot of the credits here go to Jonas Grossmann who came up with some new ideas for better mowers. Less important to us was to improve the interface. While we did add several new interface elements, some of which will be described in Chapter 5.1, it was never our goal to completely redo the interface. To this day, it remains in a state of flux. The general consensus is that since we do not yet know how Audens' "guts" will evolve, it does not make a lot of sense to create a better interface at this stage – it would only become obsolete again in short time.

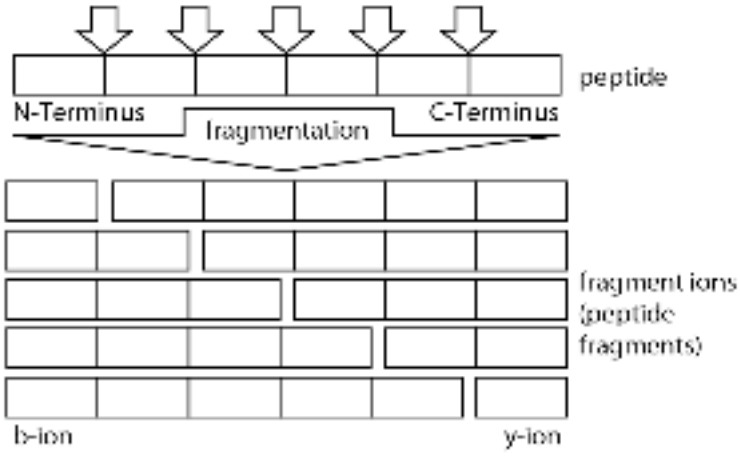
2. Introduction to Mass Spectrometry and Peptide Sequencing

DNA, short for deoxyribonucleic acid, is the information blue-print of a cell. The cell uses this information to create proteins. Proteins themselves are made of amino acids. There are 20 different kinds of amino acids. It is our aim to find the protein's sequence information in order to identify and characterize proteins of interest. Unfortunately, it's very expensive to find this information directly. Instead, we split or "digest" a protein into several peptides, whose abundance can then be measured using a mass spectrometer and sequenced using a sequencing application. This allows us to eventually find the structural information we are searching, the protein's amino acid sequence.

2.1 Mass Spectrometry

A mass spectrometer measures the mass distribution of the ions inside a sample. Ions are molecules that have been electrically charged. The mass is being measured using "Daltons". A Dalton, or "Da" for short, is defined as 1/12th of the mass/charge of a single atom of the monoisotope of carbon-12.

As mentioned earlier, the proteins are digested into peptides. These peptides are then fragmented into peptide fragments. Since this fact will become of interest later, when we discuss mowers, I will quickly explain what this means. When a doubly charged peptide is being fragmented and if the fragmentation is done optimally, most peptides will break into exactly two peptide fragments. Those are called fragment ions. The peptide has an N-Terminus and a C-Terminus. One of the fragment ions will have the N-Terminus. This fragment ion is called the b-ion. The other one will have the C-Terminus. This one is called the y-ion. The same peptide will not always create the same two fragment ions. In fact, one kind of peptide is capable of creating a limited amount of different fragement ions. The following illustration shows how this works. One single peptide which has five "attack points", e.g. any two adjacent amino acids which can break apart, can be broken in five different ways:



The mass of the resulting fragment ions is then being measured by the mass spectrometer. This implies that in the mass spectrometer data file, each peak has a corresponding peak whose mass, added up with the first peak's mass, results in some mass that is specific to each peptide - the parent mass. This mass is the actual mass of the tryptic peptide plus 2, i.e. the sum of the amino masses plus 19 on the C terminus and 17 on the N terminus (due to the way the peptide is built). This sum is called the parent mass. As an example, and ignoring biological aspects like the just mentioned offset of 2, if a peptide has a parent mass of 1000 and we find a peak with a mass of 400, it results that we must also find a corresponding peak (called a "complementary peak") with a mass of 600, of course withing a certain tolerance which is determined by factors such as the mass spectrometer.

This fact is being used by the complement mower, explained in Chapter 5.2.1.

However, even though we now know the mass of the fragment ions, we don't know what the original peptide sequence was. This is where sequencing comes in.

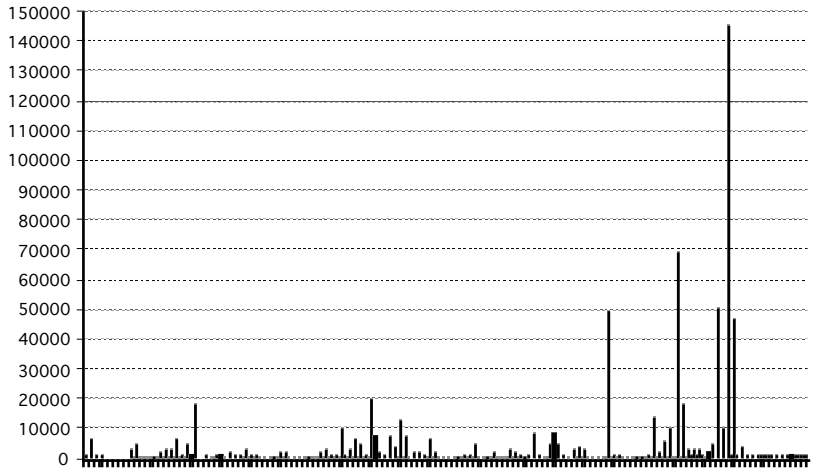
2.2 Peptide Sequencing

Basically, mass spectrometry data is a set of data points. Each point consists of a pair of values, namely a mass and an abundance. A file containing mass spectrometry data may look something like this:

```
222.9 2010.0
227.9 7240.0
231.1 1262.0
254.0 1946.0
267.0 6172.0
276.0 20159.0
```

Of course, in reality, there are hundreds of such value data pairs in a mass spectrometry data file. However, for the moment, this is all we need to know in order to understand how Audens works. The first value in each pair is called the mass, the second value is called the abundance.

On a graph, these pairs may look something like this:



(x-Axis: mass, y-Axis: abundance)

Some of the value pairs in this file are from the actual peptide fragments. We call these "true peaks". Others, which we call "grass peaks", are not. They may be random noise, they may be there because the sample was contaminated, they may be there due to measurement errors or they may be there for other reasons.

Now that we have this data, we somehow need to find out which peptide created it. There are two approaches to this problem. Up to now, the most common solution to sequencing has been to rely on existing data. Programs like Sequest[3] take mass spectrometry data and compare it to a database such as Swiss-Prot[4]. This approach works well in many cases, however, it does have some serious problems which render it useless for others.

First of all, for correctly sequencing a peptide, said peptide has to be in the database. Programs relying on databases naturally can't find anything that is not in this database. Due to the amount of data and the difficulties in obtaining this data, it is unreasonable to expect such a database to be complete. Hence, there will always be peptides that can't be correctly identified by such a solution.

Second, the contents of the database may be erroneous. For example, while man's genome is now

completely decoded[5], there are still errors in this database[6]. Even a very good database will most likely contain at least some errors. These errors may make it impossible to find some proteins.

Third, the mass spectrometry data may be of a protein that is mutated. Peptides from such a protein are unlikely to be in the database. This applies, for example, to cancerous cells. Cancer changes the DNA of a cell which means that the peptides this cell's proteins produce are different from most other peptides. Therefore it is hard to get information about such peptides using the database scheme.

Fourth, there may be other issues with the original sample that make sequencing problematic. For example, the protein may be a product of alternate splicing, meaning that the same protein can occur in different forms called genotypes.

In many cases, sequencing using a database works almost perfectly. In others, it doesn't work at all. This brings us to the second approach to sequencing: de novo sequencing. A de novo sequencing program like Audens will compute the original peptide simply by looking at the mass spectrometry data, thereby avoiding the use of a database. To get a protein sequence, several peptides need to be sequenced by Audens and afterwards assembled by hand.

In most cases, the grass peaks[7] make up the majority of a given tandem mass spectrometry data file, which leads to a problem: Which peaks does the sequencing algorithm assume to be part of the solution? A perfect solution would only use the true peaks, ignoring the grass peaks. Hence, one important task in Audens is to discern between true peaks and grass peaks.

A more in-depth explanation of mass spectrometry, peptide sequencing and fragmentation and Audens' approach to sequencing can be found in Jonas Grossmann's diploma thesis[8].

3. Introduction to Audens

As established earlier in Chapter 2.2, erroneous data makes up the majority of a data file, and in order to correctly sequence such a file, we need to eliminate these grass peaks. Hence, we need to find a way to correctly identify the true peaks. Since we can't say with absolute certainty that a given peak belongs to the true peaks or to the grass peaks, we assign it a rank. The higher a given peak's rank, the higher the likelihood that it is a true peak. We call this rank the "relevancy" of a peak.

We calculate the relevancy of each peak by applying certain heuristics. These heuristics are implemented in the aptly named mowers, which are explained in Chapter 5.2. After the mowers have run over the data, we calculate the most likely sequences based on this ranking using an algorithm proposed by Ting Chen et al in their paper[9].

This algorithm is based on dynamic programming. It computes the n most relevant paths through the dataset, based on each peak's relevancy and based on the possible masses of each peptide segment. The relevancy of a path is defined as the sum of the relevancies of the peaks that said path includes. This means that if a path is relevant, it contains many peaks which have high relevancy, e.g. have a high likelyhood of being true peaks. The algorithm is not based on any biological rules save for some basic static values that it uses[10]. It is purely mathematical, and as such, will most likely not be changed anymore. It has not been changed during the course of this semester work.

Since the sequencing component of Audens doesn't know anything about biology and just computes the most likely sequence from the mower's output, the progress must come from the mowers.

As mentioned, the sequencing algorithm takes the mowers' output and computes the n most likely solutions. Specifically, the algorithm walks along some of the peaks[11] using specific rules. The sum of the relevancies of all the peaks that are being used is being maximized[12], and the mass difference between two subsequent peaks in the result must be biologically plausible, e.g. it can only be a residue mass of one or several amino acids or amino acid modifications.

As a simplified example, imagine that the mass of a peptide segment could either be 2 or 3. Let's assume that we have the following data set:

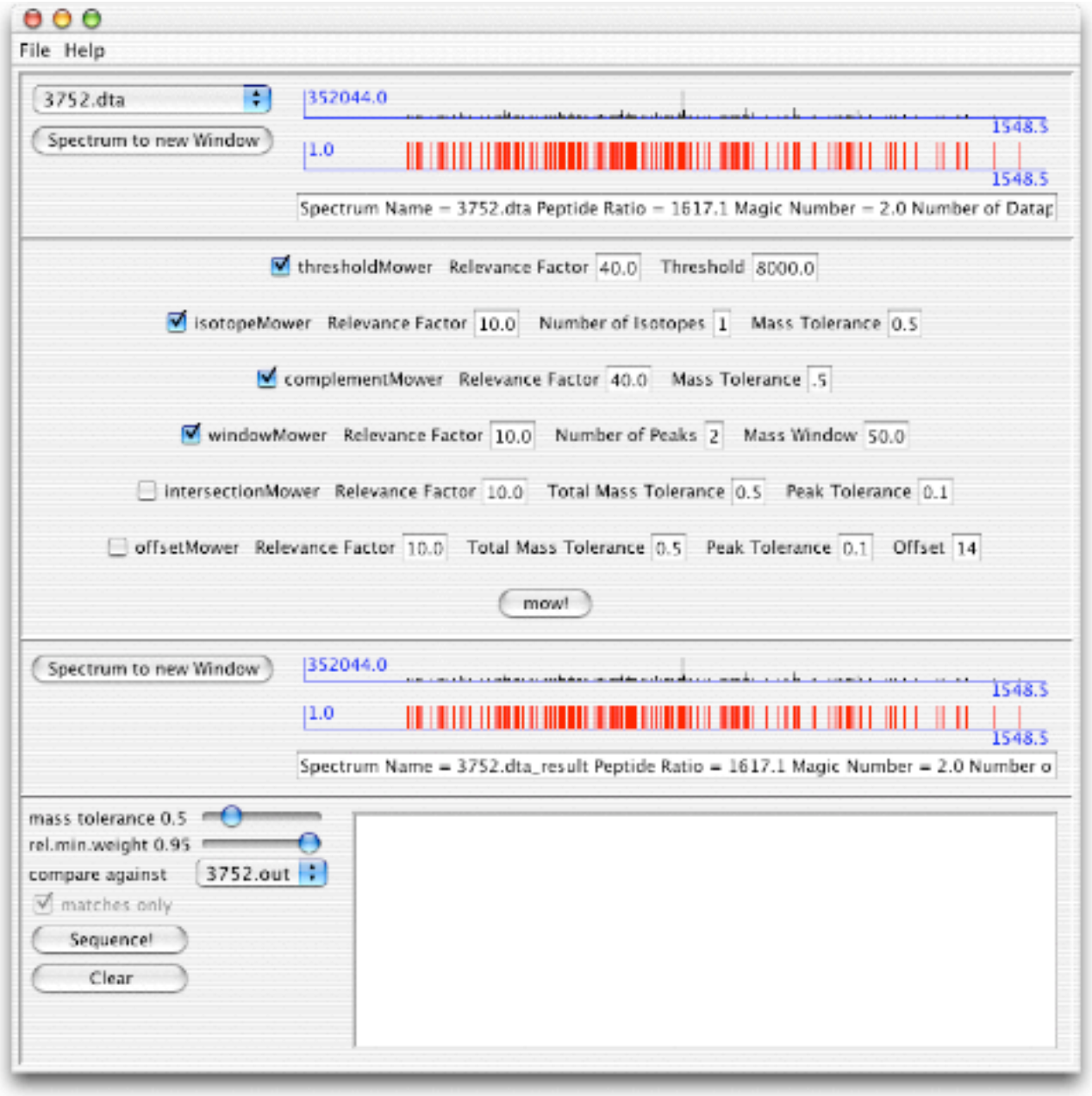
mass	relevancy
1	2
2	1
3	1
4	6
5	1
6	2
7	2

There are two possible paths through this data set: We either assume there are two peptide segments of mass 3 or three peptide segments of mass 2. In the first case, we include the peaks with masses 1, 4 and 7, which gives a relevancy sum of 10. In the second case, we include the peaks 1, 3, 5 and 7, which evaluates to a relevancy sum of 6. The first path is more likely to be true since the sum of the relevancies it includes is higher.

We are not entirely positive that the sequencing algorithm gives the best possible ranking, and an enhanced ranking of the algorithm's output could be implemented in Auden's future. This will help to get better results to always come out at the top.

4. The "Original" Version of Audens

Using screenshots, I will quickly introduce Audens the way it was before I took over the task of developing it further.



This is the main window of the "old" Audens. It is divided into four separate panes. In the top

pane, the data file that should be sequenced can be selected. Two small graphs of the contents of the file will then be shown just to the right of it. The top graph shows the actual contents of the file, while the bottom graph shows each peak's relevancy. Since the data has not yet been mowed, all relevancies are the same. These graphs will be discussed later in this chapter.

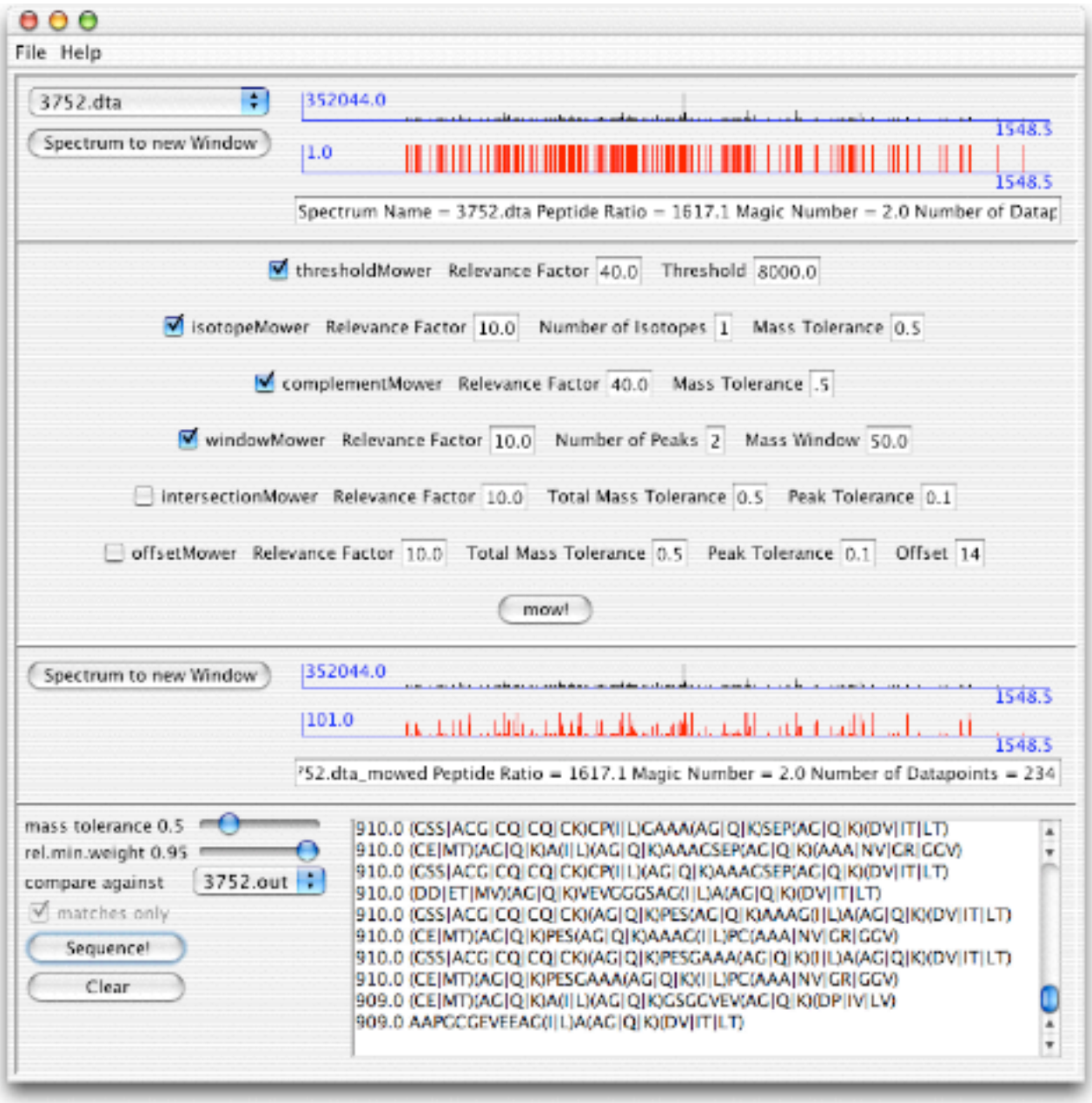
In the second pane there is a list of mowers. Each mower can be turned on or off individually. Also, each mower has a "Relevance Factor", which is the mass of this particular mower compared to the other mowers. If one mower has a higher relevance factor, it will have a larger influence on the result. Mowers can also have individual mower-specific settings. We will discuss some of these settings when we discuss the mowers themselves.

The third pane shows the result of the mowers. Since we haven't run the mowers yet in this screenshot, it looks exactly like the graph in the upper part of the window.

In the fourth and last pane, the actual sequencing takes place. There are some settings for sequencing, a button to start sequencing and a text field for the results of the sequencing. There is also a "Clear"-button which empties the text field. Again, since we have not sequenced anything, the text field is empty.

After we mow and sequence an input file, the second graph will reflect the new relevancies and the result text field will contain several possible results, ranked according to their likelyhood. These results were the top results based on their likelyhood, computed by the sequencing algorithm. The window now looks like this:

Audens - Automatic De Novo Sequencing



The sequencing results are shown in a kind of regular expression, where (AC|DE) means any of the following amino acid sequences:

AC

CA

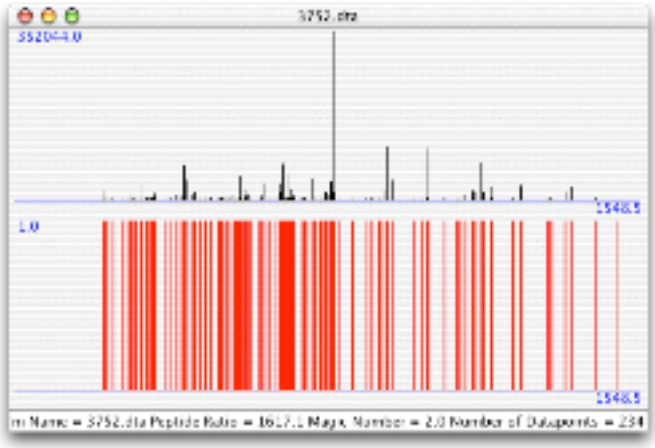
DE

ED

I will simply refer to these encoded results as "regular expressions" even though they don't conform to any existing regular expression standard such as Perl compatible regular expressions (PCRE). From a programmer's standpoint, they are just that: A simple form of regular expressions. In other works on Audens, they may be referred to as "multisequences" instead, because from a biology standpoint, they encode multiple sequences.

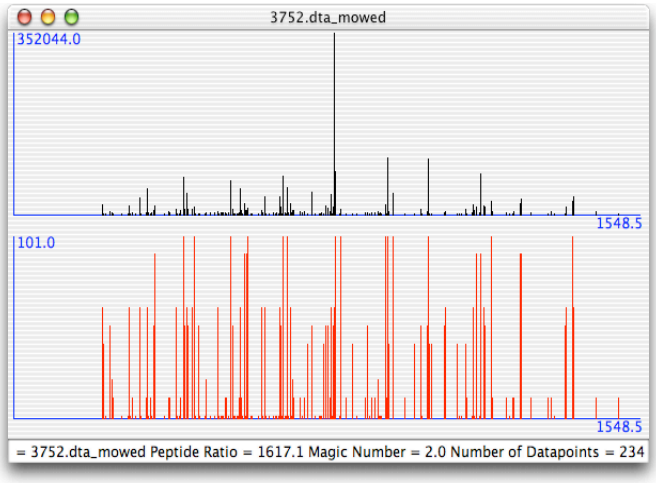
For comparison, Audens also puts the results from SEQUEST into this text field. SEQUEST is a sequencing program that uses a database approach to sequencing.

There are two buttons called "Spectrum to new Window". These will open a new window with larger versions of the small graphs shown in the main window.



Again, on the top we have the actual output from the mass spectrometer while on the bottom, we have each peak's relevancy. Since this particular file has not been mowed yet, all relevancies are equal.

To the right is a version of this window after mowing. Large peaks in the original file are slightly more likely to be relevant as the abundance of a peak is used in the window mower.



5. Our Contribution to Audens

Working together with Jonas Grossmann, Mark Cieliebak and the rest of the Audens team, I added several new features to Audens and improved existing functionality. We can roughly divide these changes into three categories:

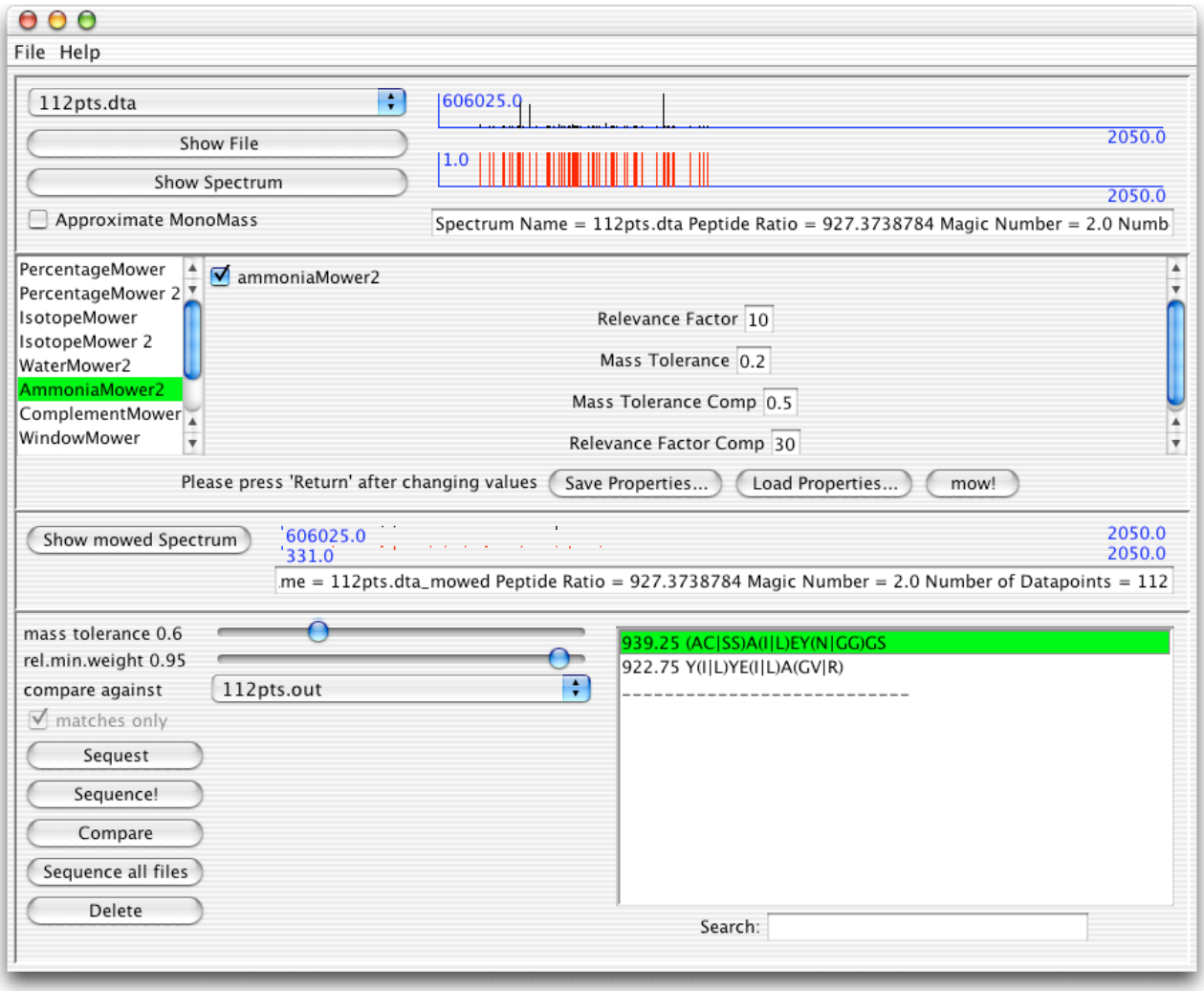
1. Interface
2. Mowers
3. Additional changes

Furthermore, we developed several complimentary tools that help with evaluating our test data.

5.1 Interface

The most obvious changes have been introduced to the interface. I will first show a screenshot and then explain some of the changes in detail.

5.1.1 The Main Window



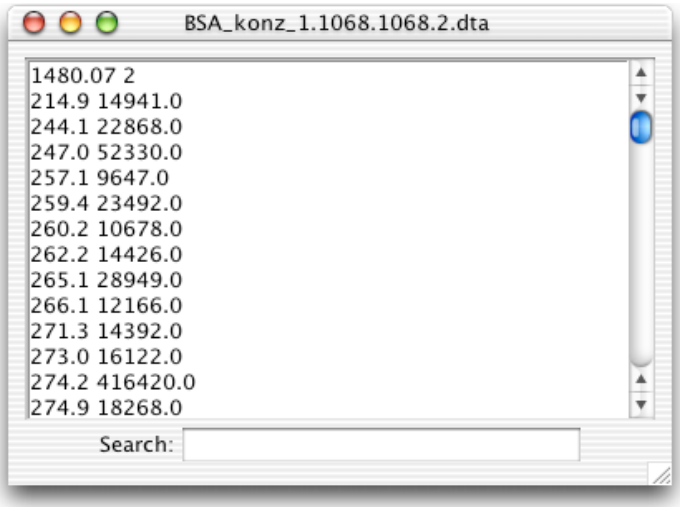
Again, I will quickly go through all four panes and explain what we changed.

In the top panel, little has changed. The way spectra are displayed has been updated. These changes are described in Chapter 5.1.2. There is a new "Approximate MonoMass"-button. This button was one of the two factors that helped a great deal in improving Audens' output quality. Audens uses the peptide's parent mass for several of its calculations. Especially for larger peptides, the average parent mass is approximately measured by the mass spectrometer. The algorithm, on the other hand, expects monoisotopic masses. Therefore, it makes sense to try to change the measured mass to the monoisotopic mass. The reason for this is that the sequencing algorithm walks over the monoisotopic peaks to find the correct sequence. Since the monoisotopic mass is not known, we

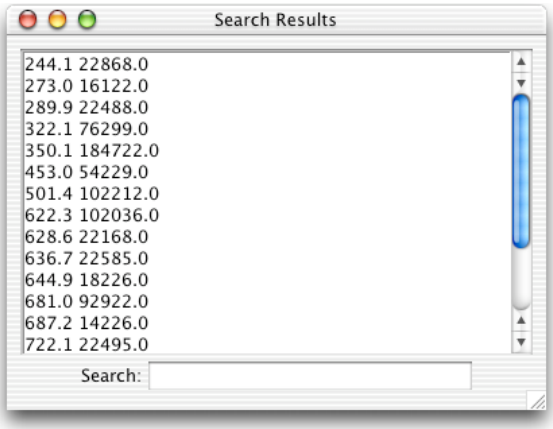
use a formula based on the amount of isotopes in amino acids in real life to calculate the approximative monoisotopic mass from the average parent mass[13]. Turning this button on forces Audens to use the approximative monoisotopic mass for all of its calculations.

Using the approximative monoisotopic mass greatly improved the sequencing results, as shown in Chapter 1.1. In a test sequencing 103 data files[14], Audens found at least one meaningful result[15] for only 6 data files when running without the approximative monoisotopic mass, but when running with the approximative monoisotopic mass turned on, it found at least one meaningful result for 34 data files, a huge improvement. For a documentation of this test, see the included CD.

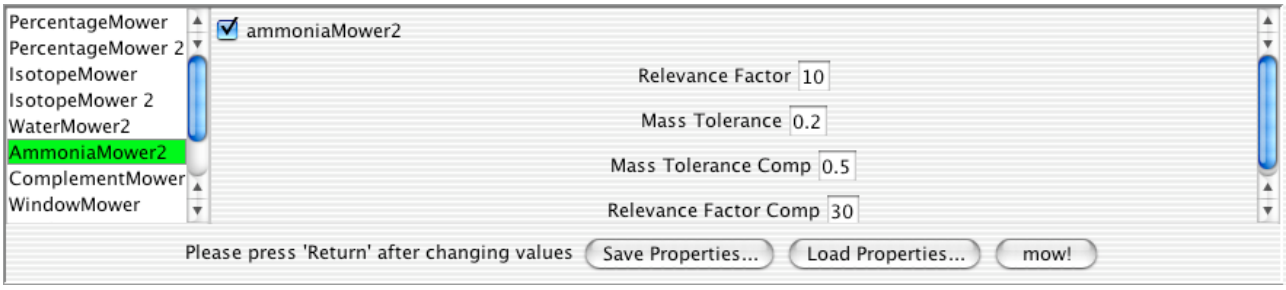
The only other thing that looks different from the old version is the "Show File"-button. It shows the contents of the currently selected spectrum data in textual form. It uses a standard Audens text window, as shown in the next picture:



What's special about the standard Audens text window is that it contains a quicksearch field. If one enters some text into this field and hit return, it will open a new text field that only contains the lines where said text occurs. For example, if we were to enter "22", it would only show the lines that contain the number "22", as shown in the picture to the right.



There are some more changes in the second panel. This is the panel that contains the mowers:

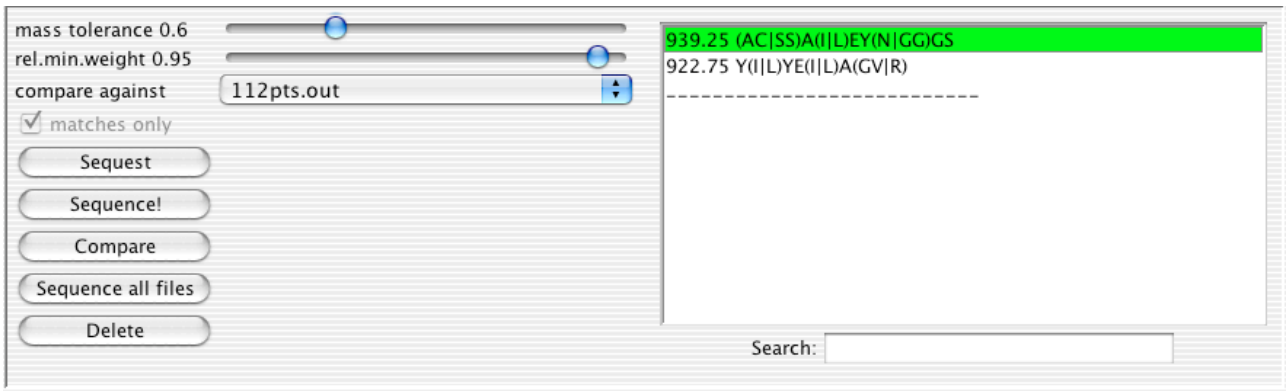


Since we have added several new mowers, we had to change organization of this panel. On the left hand side, there is a list of mowers. The currently selected mower's settings can be changed on the right hand side. Also new is the option to save and re-open the current mower settings. If one has found a combination of mower settings that produces good results, it is possible to save those settings into a text file and re-open them at a later point.

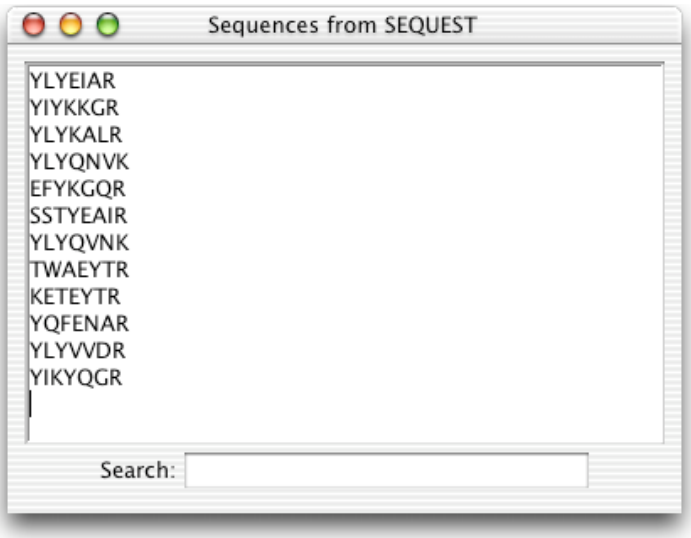
Other than the label of the button, the third pane has not been changed at all.

In the fourth pane there are some interesting new options. In the old version of Audens, we only had two buttons: "Sequence!", which sequenced the currently selected file with the current mower settings and put the results into the text field on the right hand side, and "Clear", which cleared this text field. The output in the text field would contain the sequencing results as well as results from Sequest. This was done in order to compare results.

The new panel is a bit more elaborate:



Sequest now has its own button. Pushing the button will display Sequest's output in its own Audens text window:



This means that the output window will only contain Audens results. If we were to compare the two sequencing results, we could do so using the new "Compare"-button. It does the comparison for us. This is not quite as simple as it sounds, since Audens produces its own flavour of regular expressions that can't be readily compared to Sequest's plain text output. Additionally, since Audens can produce quite a host of results, it would be too slow and take too much memory to expand all possible results hidden in Audens' regular expressions. For this reason, Audens compares the part of the regular expression that it has already expanded to Sequest's output while generating said regular expression results. As soon as the current partial result string can't possibly match any of the Sequest results anymore, recursion is halted and the expander backtracks, looking for the next possible expansion. This can be observed in framework/Expander.java.

In simplified Java, this looks as follows:

```
while(read_next_reg_exp()) {
      while(possible_result(current_result) &&
            not_finished(current_result))
      {
            compute_next_part_of_regular_expression();
```

```
        }
        if(possible_result(current_result)) {
              add_to_results(current_result);
        }
  }
```

The following is the piece of java code which will check to see the currently expanded text is actually a possible result, i.e. if it is a substring of one of the "real" results, e.g. results from Sequest:

```
// check to see if the current result can still be a "true" result
private boolean possible_result(String current_result) {
      if (seqresult != null) {
            for (int i = 0; i < seqresult.numPeptides(); i++) {
                  if(current_result.length() <=
                        seqresult.peptideNr(i).length()
                        && current_result.equalsIgnoreCase(
                        seqresult.peptideNr(i).substring(0,
                        current_result.length()))
                  ) {
                        return(true);

                  }
            }
      }
      return(false);
}
```

The result of the whole comparison process will be a text window containing zero or more found hits. As an example, it may look like this:

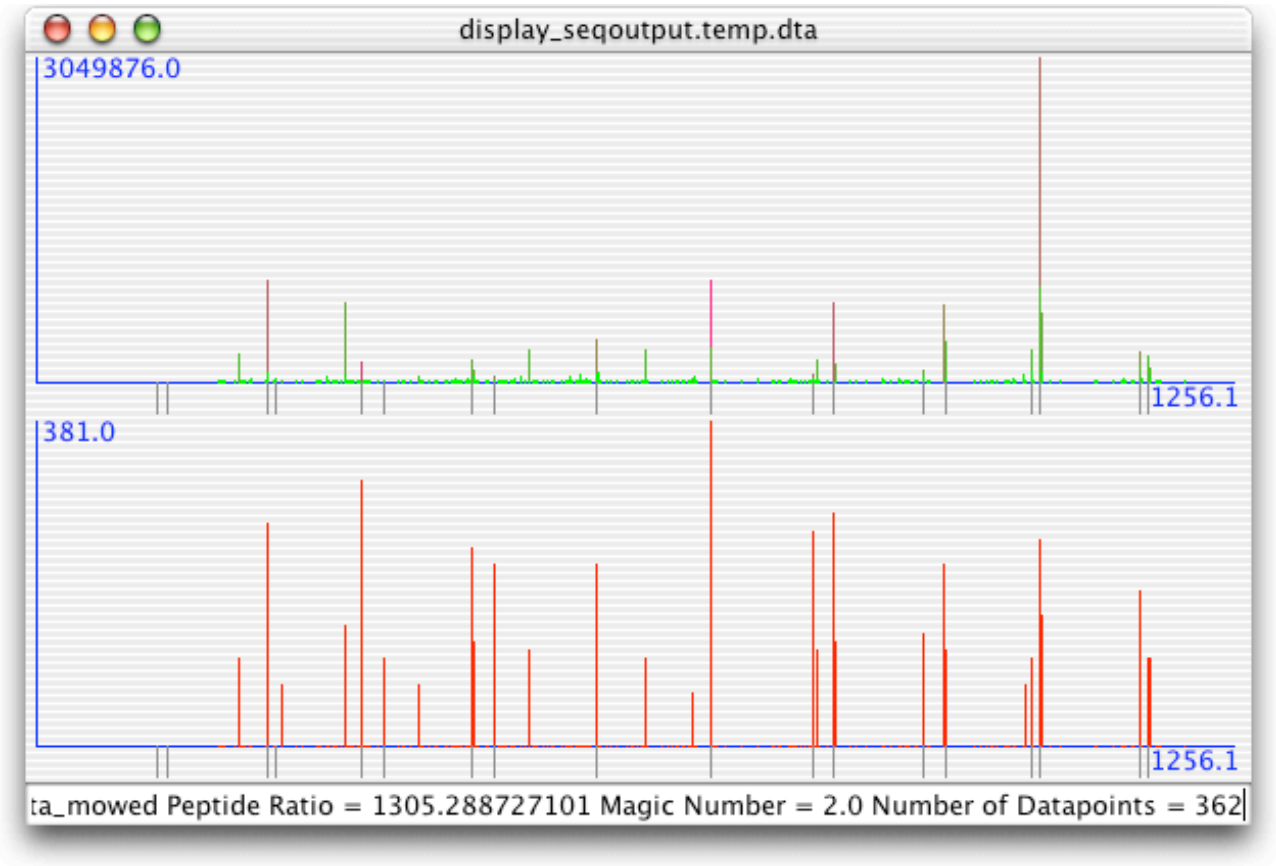
```
Found Hits:
Y(I|L)YE(I|L)A(GV|R)
expands to YLYEIAR
```

Going back to the fourth pane in Audens' main window, we see another new button called "Sequence all files". This will sequence and compare all currently available files (i.e. all files inside the directory where Audens looks for files, as specified in its config file) using the specified mower settings. It will display a list of files, the results found for each file and at which rank Sequest found these results. Sequest ranks its results based on likelyhood[16], hence if Audens has a lot of high-

ranking sequences that Sequest also ranks highly, we can assume that the current mower settings work well. Here is an example output:

```
112pts.dta
Y(I|L)YE(I|L)A(GV|R)
expands to YLYEIAR
 (score:        922.75  , rank: 2

3752.dta
Correct sequence not found

badSpectra.dta
Correct sequence not found

bsa_cov2_1pm.0955.0957.2.dta
Correct sequence not found

bsa_cov3_100fmol.0874.0876.2.dta
Correct sequence not found

bsa_cov3_100fmol.0935.0935.2.dta
Correct sequence not found

more300pts.dta
H(I|L)VDE(PQ|AGP|KP)(N|GG)(I|L)(I|L)(Q|AG|K)
expands to HLVDEPQNLIK
 (score:        1674.125        , rank: 1
H(I|L)VDEP(Q|AG|K)(N|GG)(I|L)(I|L)(Q|AG|K)
expands to HLVDEPQNLIK
 (score:        1672.125        , rank: 3
H(I|L)VDE(PQ|AGP|KP)(N|GG)(EP|II|IL|LL)(Q|AG|K)
expands to HLVDEPQNLIK
 (score:        1624.625        , rank: 6
H(I|L)VDEP(Q|AG|K)(N|GG)(EP|II|IL|LL)(Q|AG|K)
expands to HLVDEPQNLIK
 (score:        1622.625        , rank: 9
```

On the right hand side of the fourth panel, there are two changes: First, the output is no longer a simple text field. Clicking on one of the Audens results produces a spectrum with a twist. It marks the peaks that Audens used for its result with small grey lines. If everything works the way it should, these are peaks with high relevancies:



This change makes it easier to determine what effect changes to the mower settings actually have. I will explain the spectrum window in greater detail in the next section.

Second, the result list now has a search field like the Audens text windows do. If there are lots of results, this will make it easier to find all results containing a certain string.

5.1.2 The Spectrum Window

Going back to the spectrum window, we have imlemented several changes here to show more detail. Here is a picture of the old and the new version of the spectrum window right next to each other (please note that due to the fact that these are two different versions of Audens, different mower settings were used):

The graph on top shows the actual test data, the graph below shows mower results. In the new version, the graph on top also contains the information about the relevancy of peaks: the more relevant a peak is, the redder it is. Less relevant peaks are marked green.

Below each graph, there are small grey lines. If one clickes on an Audens result, these grey lines will

mark the peaks that Audens used for the result (i.e. that Audens believes to be true peaks for the selected result). If one uses the "Show Spectrum" button, the grey lines will mark peaks that Sequest thinks are true peaks (i.e. the ones it used for its highest-ranking result).

This sums up the most important changes to Audens' interface.

5.2 Mowers

As described earlier, each peak is a pair of values. The first value is the mass, the second value the abundance. Now, one might expect that the abundance is equal to the importance of a peak – the higher the peak, the more likely it belongs to the true peaks. This, unfortunately, is not necessarily the case. The abundance of a peak is one of the elements that we use to determine the likelyhood that a peak is, in fact, a true peak. The window mower uses this property. However, it is not the only one and probably not the most important one.

In order to determine the relevancy of a peak, we have to run the whole set of peaks through several filters. Those filters are based on biological heuristics. There are certain rules describing how true peaks look and how they relate to each other. We look at each peak and determine how well it fits those rules. The better it does, the more likely it is that the peak originates in fact from the actual peptide we're trying to measure.

Each of these rules is packaged in a so-called mower. A mower, in that sense, is a filter that looks at each peak and applies the little amount of knowledge it has. While each mower for itself may not be of great help in determining the true peaks, we hope that the sum of all mowers will result in a fairly good output that will help in sequencing the spectrum as described in the next Chapter.

In order to explain how exactly these mowers work, we will introduce a few of them.

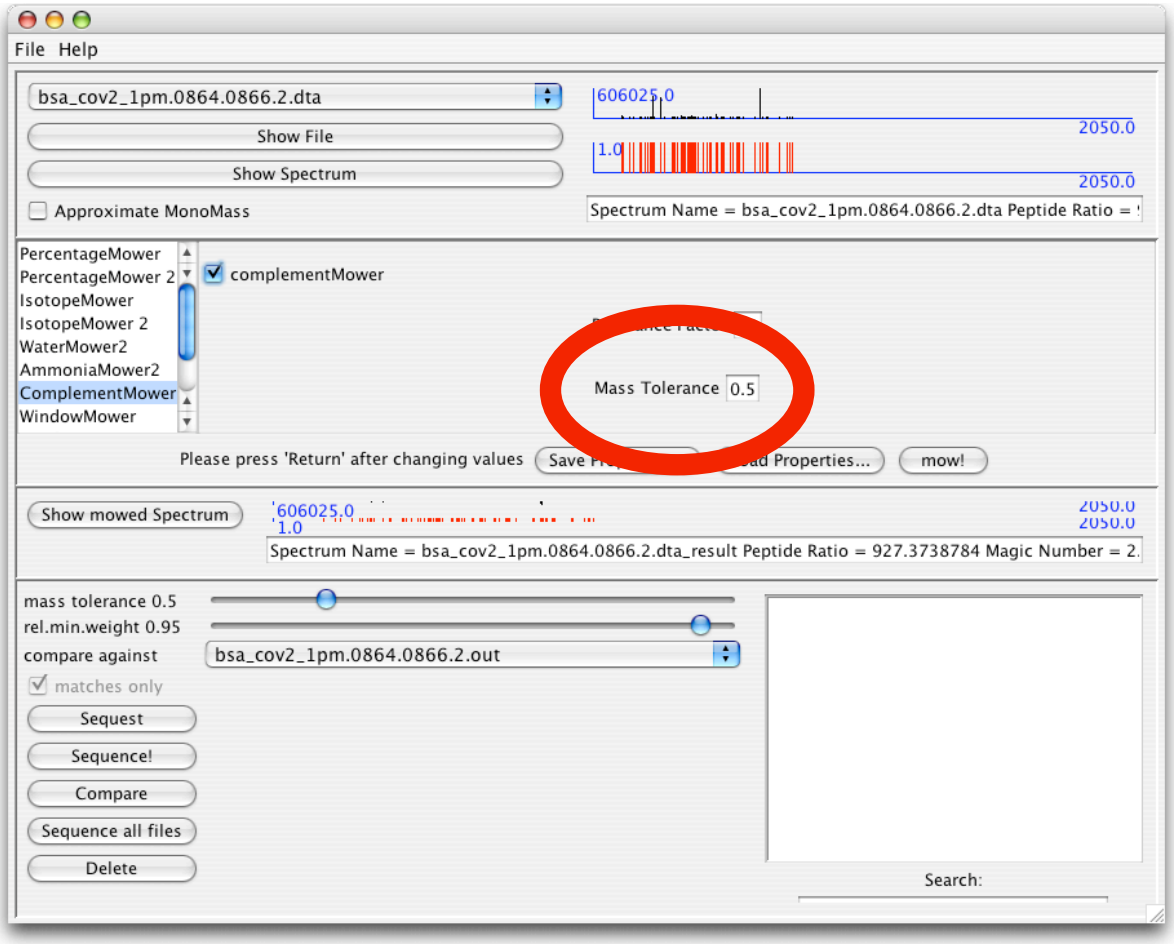
Currently, eleven individual mowers are implemented, up from seven when I started work on Audens. In the future, evaluating ideas for new mowers will be an important task in bringing Audens forward.

I will describe the complement mower in greater details than the other mower as I will use it to introduce some of the concepts that apply to all mowers.

5.2.1 Complement Mower

From the explanation of complementary peaks in Chapter 2.1, it follows that peaks that have a corresponding peak are more likely to be "true" peaks than peaks without a corresponding peak. There are certain rules that complicate this matter. For example, not all peaks are actually being measured by mass spectrometry, but for the understanding of how mowers work, this is of no importance. The complement mower knows all those rules and takes them into consideration when evaluating the data peaks, increasing relevances of peaks which fit these rules.

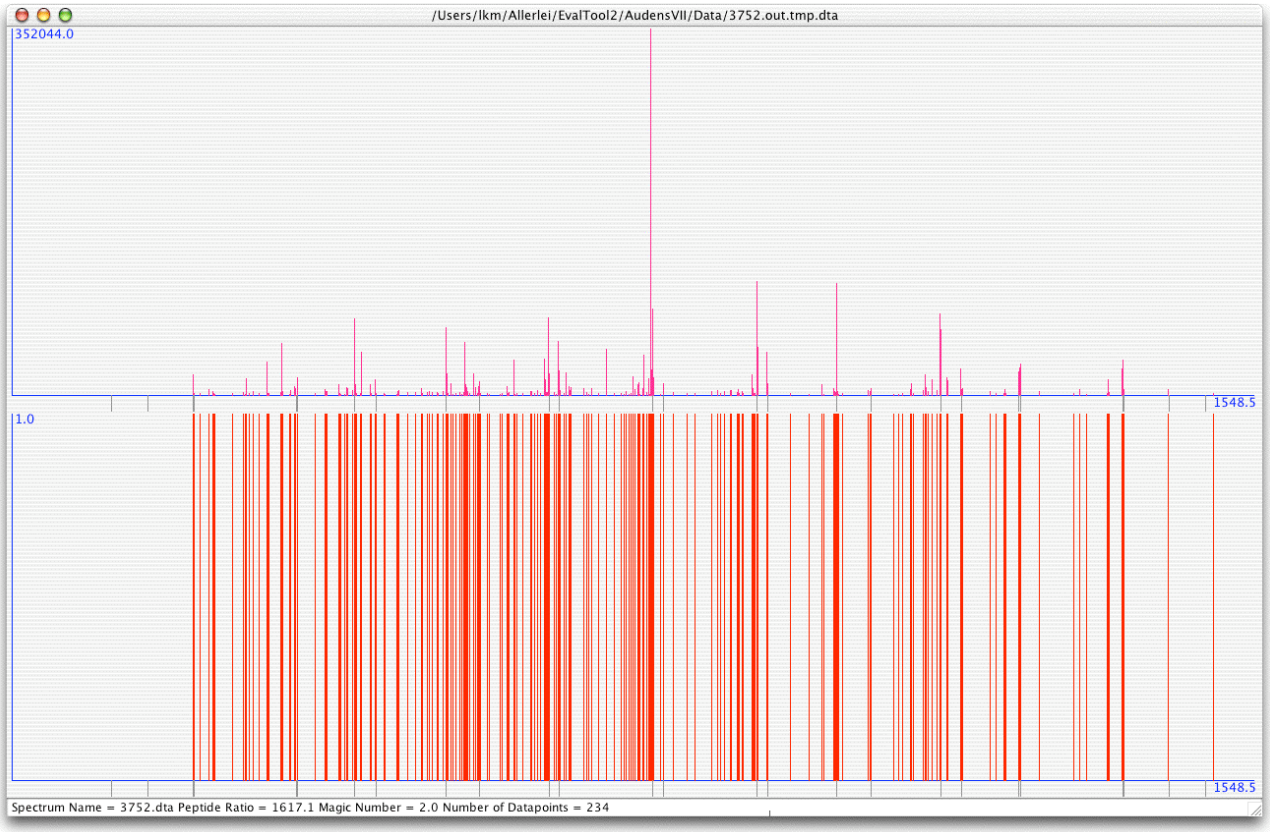
The tolerance that this mower uses to determine whether a peak has a corresponding peak is variable, hence one can change it in Audens' interface, as shown in this screenshot:



One can also see that each mower has a relevance factor. As explained earlier, this is to weight the

importance of the mowers against each other.

This is an un-mowed spectrum as shown by Audens:



In the upper part, we can see the original peaks. The x-axis signifies mass, the y-axis shows a pair's abundance. On the bottom, we see each peak's relevancy. As we have not yet run any mowers on the data, Audens treats each peak as equal: They all have the same relevancy. This graph is explained in Chapter 5.1.2.

How well does Audens work? Running the complement mower on this data set, with the default tolerance of 0.5, gives the following result:

You can see that while there are a few peaks that have complementing peaks even though they are not "true" peaks, the mower works fairly well. We know that it works well since the large peaks in the bottom graph (the ones Audens assumes to be true) often correspond to the small grey peaks below them (the ones Sequest assumes to be true). It catches all of the true peaks save for a few peaks at the end of the spectrum. The latter are not caught due to the problem with some peaks not being measured by the mass spectrometer.

5.2.2 Window Mower

Another example of a mower is the window mower. Due to biological constraints, it is not possible to have more than a fixed amount of "true" peaks in a given window. There's a minimum mass a peptide fragment can have. The difference between two true peeks can not be less than this minimum mass plus some measurement error. It follows that there can at most be a set amount of peaks in a given mass window. The smallest amino acid is G with a mass of approximately 57 Da. This means that no two true peaks can be less far away than 57 Da minus some tolerance. What

complicates this matter a bit is that we have b-ions and y-ions. Since b-ions and y-ions are in different "chains", it's possible to have a true peak belonging to the b-ions and a true peak belonging to the y-ions with both ions being closer than 57 Da.

In the end, the result is that we can not have more than two true peaks inside a mass window with a size of 57 Da plus some tolerance. The window mower increases the relevancy of the n (usually 2) peaks with the biggest abundance inside each given window with size m.

5.2.3 Combined Mowers

Due to the fact that each mower has to cope with all peaks, most mowers turned out not to be too successfull in ranking peaks. For example, consider the aforementioned complement mower: in a file with a sufficient amount of grass peaks, the likelyhood of finding a complement for a random peak is very large. There are so many peaks that no matter where one looks, a peak is nearby. Hence, we would have to lower the tolerance. This, however, means that there would be a good chance that if there was a complementing peak, it would be outside the tolerance. In some situations, the complement mower was in fact useless. On the other hand, it was clear that were we able to eliminate some of the more obvious grass, this mower could in fact become very powerful, especially considering that the actual sequencing algorithm relies on the fact that each true peak has a complement. Jonas Grossmann came up with the idea to combine two mowers. First, run a mower that, based on our evaluation and a sufficiently large tolerance, we're pretty sure won't mark any true peaks as grass peaks. Then, remove the peaks this mower marked as grass peaks from the spectrum (so far, their relevancy was decreased, they weren't actually removed) and then run the complement mower over the remaining peaks.

The results were better than we expected. In fact, it was one of the two factors that substantially increased the likelyhood of good sequencing results. We implemented this idea in several existing mowers, the percentage mower, the isotope mower, the window mower, the water mower and the ammonia mower.

5.3 Additional Tools written for Audens

We wrote several additional tools. These were not implemented as part of Audens, but as separate programs helping to improve Audens. Most of these tools are not very exciting and are just being used to create test data or do similar chores that aren't directly related to Audens and the use of Audens itself. Others were used to create statistics about our data, and even others were used to differentiate between "good" and "bad" spectra, i.e. between the spectra that Audens can sequence and those that it can't. In order to achieve this differentiation, we computed several properties for each dataset and looked for differences between good and bad datasets, meaning datasets that produced useful results and datasets that did not produce useful results for a variety of reasons. Basically, we wanted to find out what these reasons were. Some of these properties (i.e. possible reasons for bad results) were:

- The number of datapoints
- Total Ion Count (TIC; the abundance of the parent ion)
- The amount of peaks that had another peak at a given offset (good datasets are likely to have several peaks that have "neighbour peaks" at an offset of 1 Dalton, 2 Da, 3 Da (due to the fact that for, as an example, C, true peaks can have zero, one or even two C13 instead of C12), -17 Da (ammonia loss), -18 Da (water loss) but not at -20 Da. There's no meaningfull fragmentation that results in a loss of 20 Da, so it can be used as a negative control)

Two of these tools are PreProceTo and EvalTool:

- **PreProceTo** calculates these numbers, in addition to several others, and creates a list of input files with their respective numbers
- **EvalTool** is an advanced version of PreProceTo. In addition to reading out said numbers, it also calculates additional values itself, for example the amount of complementing peaks in a given input file

The data obtained from these tools allowed us to further tune the existing mowers and create new mowers.

6. Future

While Audens shows great promise, a lot still remains to be done. The things that remain to be done can largely be put into four groups:

1. Interface

Audens' interface is, while usable, not exactly beautiful or easy to use. It has grown along with the rest of the application, and it needs an overhaul. The Audens group intends to design it and implement a clean, easy-to-use interface.

2. Data analysis

In order to enhance the existing mowers and in order to create new mowers, existing and new mass spectrometry data has to be analyzed. We created programs to speed up this process, and more needs to be done in the future.

3. Mowers

Additional mowers have to be developed and implemented, and the existing mowers should be tuned further in order to get the best result possible.

4. Modifications

Additionally, the Audens group intends to apply certain modifications to the peptides they measure. All measured "true" peaks belong to one of two groups. Either they are part of the b-ion-group, or they are part of the y-ion-group. Each ion in the b-ion-group has a corresponding ion in the other group. The masses of these two corresponding groups add up to the parent mass, as explained in the Chapter 2.1 and Chapter 5.2.1.

The intent now is to modify each group in separate measurements. The aim is to shift all peaks in one group, and only these peaks, by a given value. Using this new data, one will be

able to better identify which peaks belong into one of the group, and therefore which peaks are part actual "true" peaks.

Implementing code to analyze this new data will be another task that will have to be done.

7. Terminology

Abundance	Amount of peptide fragments with a given mass
Complements	Two peaks whose masses add up to the parent mass of the given peptide
De Novo Sequencing	To sequence a peptide from mass spectrometry without using a database
Grass Peaks	Peaks that do not result from an actual measured peptide fragment
Mowers	Small application components that try to remove data which is useless for sequencing, thus making the actual sequencing process easier. They are called mowers because they "mow" the grass peaks
Parent Mass	The mass of a peptide, i.e. the sum of the masses of the peptide fragments (its amino acids) plus an offset of 2.
Peaks	A pair of two values: A mass and an abundance
Relevancy	A value describing the likelyhood that a given peak is a true peak
True Peaks	Peaks that do correspond to a measured peptide fragment

8. Endnotes

[1] Protein Identification using Mass Spectrometry: Development of an Approach for Automated de novo sequencing, Department of Biology, Institute of Plant Science ETH Zürich, Plant Biotechnology, Jonas Grossmann, April 2003, page 54ff.
[2] This excludes obviously poor spectra. The exact results can be found in Jonas Grossmann's work on pages 54 and following.
[3] http://fields.scripps.edu/sequest/
[4] http://www.expasy.org/sprot/
[5] http://www.gene.ucl.ac.uk/hugo/
[6] "The Human Genome Project aims for a resolution of one error in 10,000 base pairs", http://www.ornl.gov/sci/techresources/Human_Genome/archive/articles/drosophila.shtml
[7] Some terms, like "grass peaks", are used before they are explained because they belong into a later chapter. For this reason, they are listed in Chapter 7, "Terminology".
[8] "Protein Identification using Mass Spectrometry: Development of an Approach for Automated de novo sequencing", Department of Biology, Institute of Plant Science ETH Zürich, Plant Biotechnology, Jonas Grossmann, April 2003.
[9] "A Dynamic Programming Approach to De Novo Peptide Sequencing via Tandem Mass Spectrometry", Journal of Computational Biology, Volume 8, Number 3, 2001, pages 325 to 337.
[10] namely the possible residue masses, e.g. the possible mass differences between any two subsequent peaks occuring in the result.
[11] Belonging to either the b- or y-ion-group as these two groups are basically a mirror of each other. For a bit more on this, see Chapter 4, "Modifications".
[12] There has been some discussion about this fact. Since the current algorithm maximises the sum of the relevancies, it prefers longer sequences over shorter ones. A way around this would be to maximise the sum of the relevancies divided by the number of peaks in the sequence, but this option is not currently implemented and it is unclear whether the results would show any improvement. This also influences the ranking, as results are ranked based on this sum.
[13] Jonas Grossmann's work in Appendix table 3 on page 74 shows how the factor was calculated from the data.
[14] The documentation to this test can be found on this work's CD inside the folder called "test". The test was done using 103 representative data files. Mower settings are also included on the CD. No large amount of mower tuning was done. The list of data files contained some bad data files, e.g. data files that are very hard to correctly sequence. This test does not measure the quality of Audens as a whole, but the improvement that the monoisotopic mass approximation brought.
[15] In this case, a "meaningful result" is a result that ranks highly in Audens and matches a Sequest result that is also high-ranking, e.g. has a reasonable likelyhood of being true.
[16] The actual way Sequest ranks its results is quite involved and uses several different metrics like Delta Correlation, a preliminary score to eliminate candidates before doing the final correlation analysis or an Ions value which shows how many experimental ions matched with theoretical ions. This is explained in detail on this web page:
http://www.enovatia.com/stories/storyReader$64