

Визуализация и идентификация спектров программой *SpectraPlot*

А.В. Никитин, Р.В. Кочанов*

*Институт оптики атмосферы им. В.Е. Зуева СО РАН
634021, г. Томск, пл. Академика Зуева, 1*

Поступила в редакцию 11.07.2011 г.

Рассмотрены проблемы идентификации молекулярных спектров высокого разрешения и использования для идентификации программы *SpectraPlot*. Свободно распространяемая программа *SpectraPlot* для визуализации и идентификации молекулярных спектров может быть загружена с сайта <http://www.iao.ru/spectraplot/sp.htm>. Она позволяет изображать экспериментальные спектры, составлять и идентифицировать пик-листы, а также изображать спектры в форматах баз данных HITRAN и GEISA. Рассмотрены примеры идентификации спектров.

Ключевые слова: идентификация спектров, молекулярная спектроскопия, спектры высокого разрешения; spectra identification, molecular spectroscopy, high resolution spectroscopy.

Введение

Несмотря на большое количество программ для работы с графиками и с изображением спектров, они не всегда могут быть адаптированы к спектроскопическим задачам. Поэтому во многих спектроскопических лабораториях развивали собственные программы визуализации спектров [1–4]. Часто программы визуализации объединяют с вычислительными программами, например такими, как подгонка контура линий [5–7], с программами для нахождения пиков или программами для работы с базами данных. Кроме этого программы визуализации можно разделить на адаптированные только к определенной операционной системе и кросс-платформенные. К последним можно отнести программы, написанные в средах LabView, Maple и др.

Для идентификации спектров желательно иметь возможность быстрого просмотра большого количества спектров одновременно. Скорость изображения важна, особенно в случае, если необходимо изображать большое количество спектров одновременно. Идентификация спектра — сложный процесс, который можно условно разделить на три этапа: два подготовительных и третий непосредственно идентификация.

Первый этап — это параметризация спектров, или представление спектров в виде списка линий, для каждой из которых задаются параметры, такие как положение, интенсивность, самоуширение и, если необходимо, уширение другими газами. Обычно для параметризации спектров сначала генерируют список линий, а затем находят их параметры. Часто приходится итерационно повторять эти две операции несколько раз.

Второй этап — это создание модели спектра, например с помощью эффективного гамильтониана и эффективного дипольного момента, которой предполагается описать совокупность наблюдаемых переходов. Имеется большое количество программ, позволяющих вычислять спектры из параметров эффективного гамильтониана, например STDS [6], MIRS [7]. Программа *SpectraPlot* просто сопрягается с программами [5–7]. В последнее время в связи с увеличением точности расчетов *ab initio* можно также идентифицировать спектры непосредственно из этих расчетов.

Третий этап — непосредственно идентификация, т.е. установление соответствия между параметризованными переходами и моделированным спектром. Коротко опишем типичный процесс идентификации молекул с хорошо разрешенной колебательно-вращательной структурой. Основным источником информации являются комбинационные разности и регулярные, характерные структуры спектра. Под комбинационными разностями понимают совокупность всех переходов на определенный верхний уровень, при этом считается, что все нижние уровни известны достаточно точно. Обобщением метода комбинационных разностей можно считать алгоритм, основанный на методе Ritz.

В последнее время все чаще регистрируются серии спектров с различными температурами, что позволяет оценить значение вращательного квантового числа J нижнего уровня перехода (ниже просто J_{Low}). Даже приблизительное знание J_{Low} позволяет многократно сузить диапазон поиска. Задачи визуализации и идентификации спектров во многом взаимосвязаны. На всех этапах идентификации спектра важен визуальный контроль идентификации.

Программа *SpectraPlot* позволяет быстро просмотреть большое количество спектров одновремен-

* Андрей Владимирович Никитин; Роман Викторович Кочанов.

но, а также редактировать идентификацию. Она использовалась для идентификации спектров CH_3D , CH_3Cl , CH_4 , PH_3 вместе с программой MIRS. Заметим, что *SpectraPlot* совместима с программой MIRS <http://www.iao.ru/mirs/mirs.htm>.

В настоящей статье рассмотрены пример проекта для идентификации пик-листа с помощью программы *SpectraPlot* и некоторые детали процесса идентификации. Приведены используемые форматы входных файлов, описаны корректор идентификации и формат файла для сохранения изображения.

Пример проекта *SpectraPlot* для идентификации CH_3D

Типичный проект *SpectraPlot* для идентификации спектра приведен на рис. 1. В верхней части изображены экспериментальные спектры, в середи-

не пик-лист с идентификацией, в нижней части вычисленный спектр, часто называемый предсказанием. Принятая идентификация начинается с ключа «+» или буквы. Вместе с принятой идентификацией пик-лист может содержать предварительную идентификацию, помеченную другими ключами. Корректор позволяет редактировать идентификацию и добавлять новые линии. Изображенный на рис. 1 проект расположен в директории *SpectraPlot*\examples\example1. Хотя проект example1 содержит только три *SubScreen*, в программе *SpectraPlot* нет каких-либо ограничений на количество *SubScreen*, а также на количество и тип графиков, изображаемых в каждом *SubScreen*.

Вместе с идентификацией в отдельном *SubScreen* часто необходимо выводить дополнительную информацию, такую как вероятность идентификации. Как правило, спектры идентифицируются

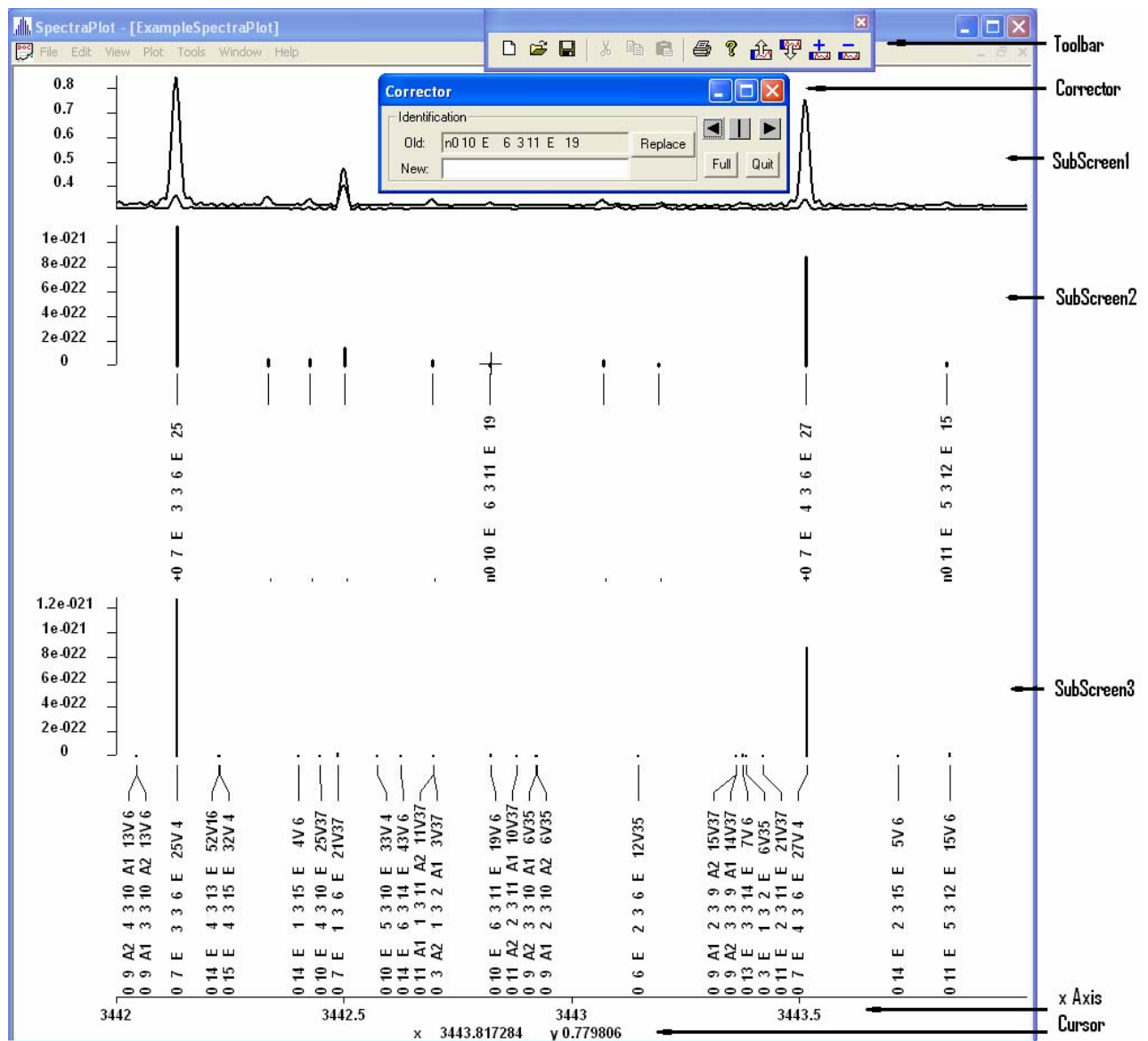


Рис. 1. Три *SubScreen* на одной оси X. Верхний экран – два спектра Фурье, второй пик – лист с частичной идентификацией, нижний – вычисленный спектр с идентификацией и дополнительной информацией о колебательной полосе (V...)

с помощью комбинационных разностей, представленных в виде списка возможных идентификаций определенного уровня либо в виде списка переходов и наиболее вероятной идентификации каждого перехода.

Вероятность идентификации верхнего уровня можно условно представить как $W = W_{posit}W_{intens}$, где W_{posit} — вероятность совпадения положений, W_{intens} — вероятность совпадения интенсивностей. При этом W_{posit} учитывает также возможность альтернативной идентификации одного или нескольких переходов, входящих в комбинационную разность. Если вероятность идентификации больше определенного порога, то уровень идентифицируется программой однозначно и ниже называется программной идентификацией (PRID). Не все уровни идентифицируются однозначно программой, и для значительного числа уровней вероятности различных идентификаций сравнимы. Кроме того, PRID может содержать ошибки и необходимо контролировать идентификацию, предложенную программой.

Процесс идентификации итерационный. Каждую итерацию можно условно разделить на несколько этапов: уточнение расчетного спектра, предварительная идентификация с помощью программ, визуальная проверка и принятие, либо частичное принятие, либо отклонение новой идентификации. На каждом этапе желательно иметь возможность отличать новую идентификацию от принятой ранее. Еще лучше сохранять номер итерации, в которой идентификация была принята. Дело в том, что некоторые итерации могут содер-

жать скрытые ошибки идентификации, которые обнаруживаются только впоследствии, например смешивание двух близко расположенных колебательных полос. Просмотр идентификации позволяет корректировать идентификацию или отклонять ее полностью, если она содержит много ошибок. В последнем случае необходимо выполнить программную идентификацию еще раз с более жесткими параметрами поиска. В примере на рис. 1 номер итерации различается по букве в первой позиции.

Часто процесс идентификации становится неэффективным, например из-за допущенных ошибок в идентификации или недостаточно качественного пик-листа. В обоих случаях желательно просмотреть переходы, вызывающие подозрение, или улучшить пик-листа. При этом на экран желательно выводить информацию, например о наиболее вероятной комбинационной разности и об альтернативной идентификации. Даже при идентификации одного и того же спектра на разных этапах желательно выводить различную информацию. Именно поэтому вывод текстовых полей в программе *SpectraPlot* достаточно общий и не привязан к какому-либо фиксированному формату.

Создание проекта

После запуска программы на экране появляется пустое окно, в котором можно изобразить один график. Для добавления спектров надо нажать правую кнопку мыши и войти в меню *properties*, после чего на экране появится диалог, показанный на рис. 2.

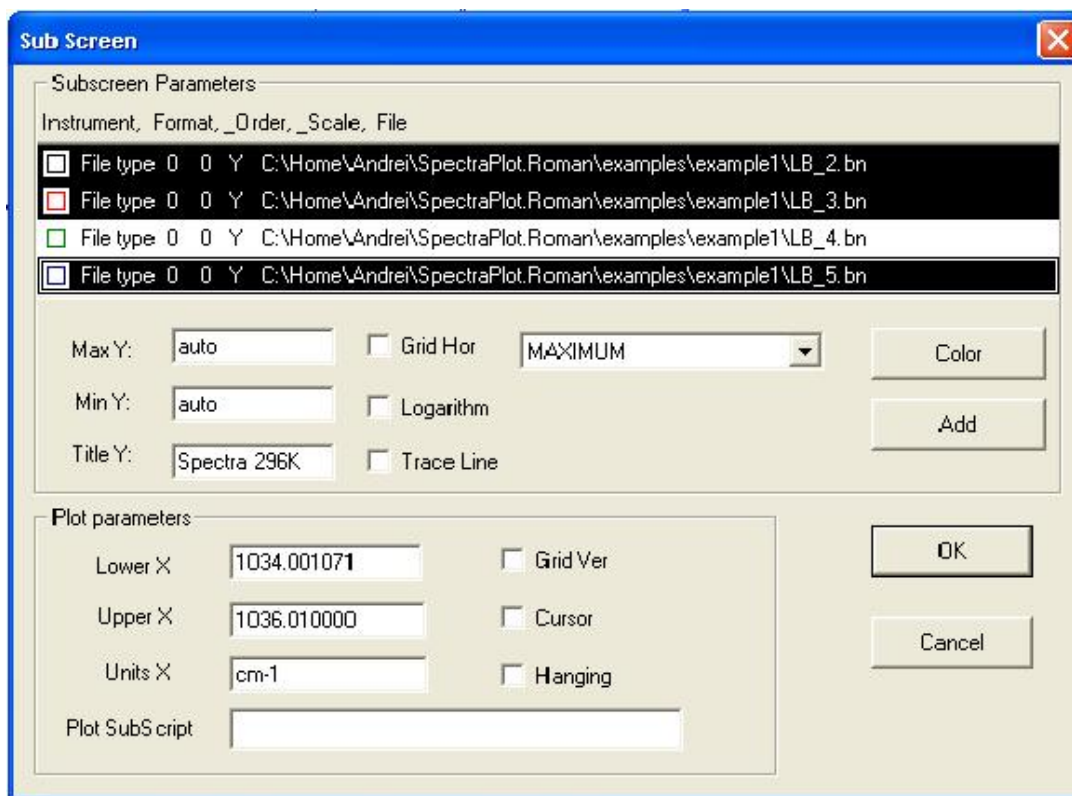


Рис. 2. Диалог *SubScreen*

Кнопкой *Add* можно добавить файлы в список в верхней части диалога *SubScreen*, а потом выделить в этом списке файлы, которые следует изображать. Здесь же можно задать диапазоны изображения и некоторые опции. Наконец, следует нажать ОК и появится изображение.

Программа *SpectraPlot* позволяет воспроизводить два типа изображений. Пример первого типа, когда вдоль оси X — волновое число (обычно в см^{-1}), показан на рис. 1. В этом случае можно изобразить спектры Фурье или дискретные частоты переходов с подписями (идентификацией) или без. По оси Y при этом могут быть либо поглощение (пропускания), либо интенсивности линий. Изображение второго типа — это диаграмма уровней энергии, ассоциированная с файлом, содержащим уровни энергии (*LEVELS*). При этом вдоль оси X откладывается дискретное вращательное число J , а вдоль оси Y — энергия уровня (например, в см^{-1}). Диаграмма уровней энергии полезна для проверки идентификации, так как для каждого идентифицированного уровня можно вывести, например, отклонение «наблюдаемого» уровня от модели. В обоих типах изображений для одной оси X можно создать несколько графиков с различными Y . Например, на рис. 1 изображены четыре графика с различными осями Y . Но если в проект уже занесены данные типа *LEVELS*, то не следует добавлять данные типа *FLOAT4* или *STICK*. После нажатия кнопки *Add* появляется диалог, предлагающий выбрать файл, который следует изображать. После выбора файла появляется окно, описанное ниже, в котором следует определить формат файла.

Параметры диалога *SubScreen* можно разделить на две группы: в верхней части диалога находятся параметры, относящиеся только к одной из осей Y , а в нижней части к оси X , единственной для данного графика.

Ниже перечислены параметры, относящиеся к оси Y .

Список файлов, которые можно изобразить в данном окне. При этом будут изображены только файлы, выделенные черным. Добавить файл в список можно кнопкой *Add*. Выделенный файл можно убрать клавишей *Delete*. Изменить выделение можно либо мышью, либо клавишей *Space*. Изменить инструмент изображения можно клавишей *T*, изменять порядок изображения объектов можно клавишей *O*. Кроме того, клавишей *S* можно изменить определение верхней и нижней границ графика по умолчанию.

Ниже приведены некоторые параметры, влияющие на изображение. Их можно разделить на параметры, относящиеся к одному *SubScreen*, и параметры, относящиеся ко всему графику целиком.

Параметры, относящиеся к одному *SubScreen* (ось Y):

Max Y , Min Y : границы графика сверху и снизу, если ввести любую букву, то границы будут назначены автоматически.

Title Y : название оси Y .

Grid Hor: если выделен, то появится горизонтальная решетка.

Logarithm: если выделен, то используется логарифмическая шкала.

Trace Line: если выделен, то при выходе за пределы графика в верхней или нижней части появится след от кривой.

Список MAXIMUM, MINIMUM, AVERAGE в правой части диалога позволяет менять способ изображения, если на один пиксель графика приходится несколько дискретных линий.

Параметры, относящиеся ко всему графику целиком (оси X), приведены ниже.

Lower, Upper X (or J): границы графика слева и справа.

Units X : название оси X .

PlotSubscript: название всего графика.

Grid Ver: если выделен, то появится вертикальная решетка.

Cursor: если выделен, то появится курсор под графиком.

Hanging: меняет направление turns the Y для *FLOAT4*-графиков.

Входные данные

В качестве входных данных используются три типа файлов.

Первый тип (FLOAT4) — бинарные файлы, состоящие из заголовка и массива четырехбайтовых чисел с плавающей точкой. Для этого типа файлов необходимо знать начальное значение частоты, шаг по частоте, число точек и количество байт (шапка), которое надо пропустить. Фурье-спектр, заданный в виде двух колонок частоты и амплитуды, можно перевести в бинарную форму с помощью меню Tools->Text to binary float4. При этом будет создан файл с форматом спектра.

Второй тип (STICK) — текстовый файл с дискретными переходами, состоящий как минимум из двух колонок: колонки с положениями линий и колонки с интенсивностями. Кроме этих двух колонок файл может содержать колонки с текстовыми переменными, которые могут также появляться в изображении в качестве подписей.

Третий тип (LEVELS) — текстовый файл, состоящий как минимум из двух колонок: колонки с числом J и колонки с энергией уровня. Этот тип может содержать также колонки с идентификацией или любой другой информацией, которая может выводиться в качестве подписей к уровням.

Поля диалога Format (рис. 3) зависят от типа формата.

Если вы используете файлы с похожими форматами, например ваш файл с пик-листом имеет определенный формат или вам необходимо ввести серию спектров с одинаковыми условиями, то разумно сохранить формат — кнопка *Save* в диалоге *Format*. Тогда не надо вводить формат в следующий раз, а просто загрузить его в диалог *Format* кнопкой *Open*.

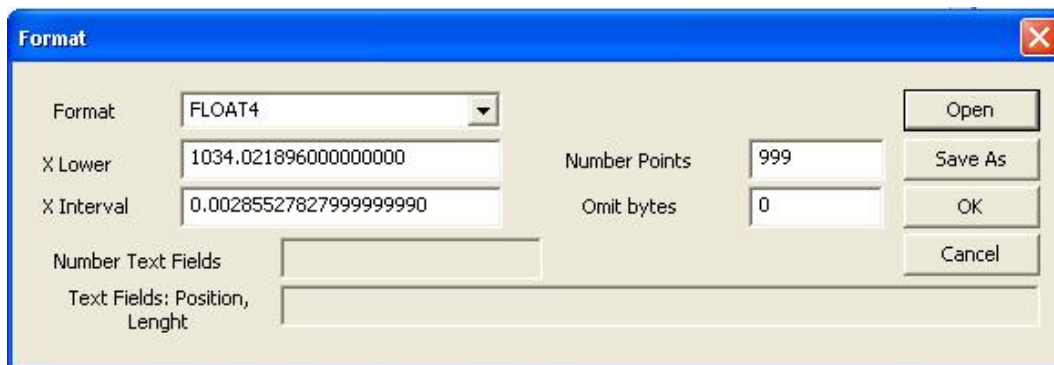


Рис. 3. Диалог *Format* – поля, которые необходимо заполнить в формате *FLOAT4*

Редактирование идентификации и составление пик-листа

Редактировать идентификацию можно только для файлов формата *STICK*. Для редактирования надо выбрать меню *Tools->Corrector*. Появится диалог *Corrector*. Изначально диалог *Corrector* появляется в упрощенном виде. Для того чтобы изобразить полный вид диалога (рис. 4), надо нажать кнопку *Full*. Кнопкой *Short* можно опять перейти к упрощенному виду. В упрощенном виде можно только изменять текстовое поле подписи и нельзя добавлять (убирать) линии. Для изменения идентификации достаточно набрать новую идентификацию в поле *New* и нажать кнопку *Replace*.

Все текстовые поля, описанные в формате, объединяются в одно текстовое поле и изображаются вместе. Поэтому в формате не следует допускать перекрывание полей. Например, в проекте на рис. 1 в нижнем *SubScreen* изображается предсказание, а в среднем *SubScreen* изображается пик-лист с идентификацией. Двойной щелчок мышки копирует идентификацию из предсказания в поле *New* корректора. Редактируя поле *New* и используя кнопку *Replace*, можно присвоить идентификацию определенному пику. Для позиционирования по пикам в упрощенном виде диалога надо использовать стрелки и квадратную кнопку *|*, а в полном виде – кнопки *Previous*, *GoTo* и *Next* соответственно.

Для редактирования пик-листа необходимо удалять и добавлять линии. Удалять линии можно

кнопкой *Delete*, а отменять удаление кнопкой *Restore*. При этом, хотя удаленные линии становятся невидимыми, а их идентификация серой, они присутствуют в списке линий и их еще можно восстановить. Для того чтобы добавить линию, надо заполнить поля *WvNm* и *Intensity* в полном диалоге *Corrector* и нажать *Add Line*. Поля *WvNm* можно также заполнить автоматически. Для этого достаточно выбрать эксперимент (один из файлов формата *FLOAT4*) и, используя кнопки *Get Max* или *Get Min*, найти максимум (минимум) предварительно выделенного правой кнопкой мыши участка спектра. Перед добавлением первой новой линии можно создать шаблон добавляемой линии – меню *Tools->Options*.

Отметим, что при добавлении новой линии необходимо задавать интенсивность приблизительно для последующей подгонки по методу наименьших квадратов. Опыт показывает, что немного завышенная начальная интенсивность подгоняется быстрее, чем заниженная, поэтому не рекомендуется занижать начальную интенсивность.

После редактирования надо сохранить отредактированный файл в другой файл, меню *Tools->Save Peak List->Save As*. После сохранения следует закрыть корректор, перезаписать новый файл на место редактируемого и загрузить заново рисунок, для того чтобы все изменения были внесены. Не следует изменять редактируемый файл какими-либо другими средствами до окончания редактирования.

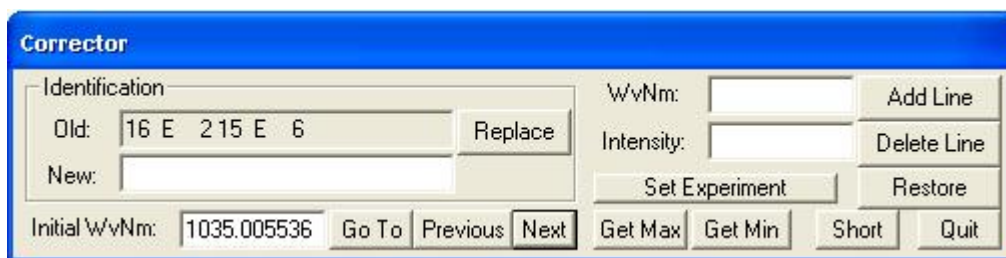


Рис. 4. Полный вид диалога *Corrector*

Заключение

Идентификация спектров — сложный итерационный процесс, который невозможен без использования программ, моделирующих спектры. Использование программы *SpectraPlot* с программой MIRS позволяет упростить идентификацию спектров. В дальнейшем предполагается улучшить взаимодействие этих программ на уровне пользовательских сообщений о завершении каких-либо действий, таких как редактирование пик-листа, завершение вычислений спектров или комбинационных разностей.

Программа *SpectraPlot* может быть использована для подгонки параметров линий одновременно к нескольким спектрам, а также для нахождения пиков из Фурье-спектров. Для этого следует запустить меню Tools->MEPL. Программа подгонки спектральных линий и программа нахождения пиков будут описаны в отдельной статье и в данной статье не рассматриваются.

Более полное описание работы с *SpectraPlot* для идентификации спектров можно найти в документации по программе. Например, вместе с идентификацией или вместо нее можно выводить полученное из температурной зависимости значение вращательного квантового числа J для нижнего уровня перехода. Часто полезно также выводить на экран отклонения от модели положений и интенсивностей, количество переходов в комбинацион-

ных разностях и вероятность идентификации. Примеры использования программы и ее последняя версия выложены на сайте <http://www.iao.ru/spectraplot/sp.htm>.

Работа поддержана грантами РФФИ № 09-05-92508, 09-05-93105 и грантом CRDF (USA) RUG1-2954-TO-09.

1. *Pickett H.* The fitting and prediction of vibration-rotation spectra with spin interactions // *J. Mol. Spectrosc.* 1991. V. 148, N 2. P. 371–377.
2. URL: <http://spec.jpl.nasa.gov/>
3. URL: <http://pgopher.chm.bris.ac.uk/ref.html>
4. *Wenger C., Boudon V., Rotger M., Sanzharov M., Champion J.-P.* XTDS and SPVIEW: Graphical tools for the analysis and simulation of high-resolution molecular spectra // *J. Mol. Spectrosc.* 2008. V. 251, N 1–2. P. 102–113.
5. *Wenger Ch. and Champion J.-P.* Spherical top data system (STDS) software for the simulation of spherical top spectra // *J. Quant. Spectrosc. and Radiat. Transfer.* 1998. V. 59. P. 471–480.
6. *Wenger Ch., Boudon V., Champion J.-P., Pierre G.* Highly-Spherical Top Data System (HTDS) Software for the Spectrum Simulation of Octahedral XY_6 Molecules // *J. Quant. Spectrosc. and Radiat. Transfer.* 2003. V. 66, N 1. P. 1–16.
7. *Nikitin A., Champion J.-P., Tyuterev V.G.* The mirs computer package for modeling the rovibrational spectra of polyatomic molecules // *J. Quant. Spectrosc. and Radiat. Transfer.* 2003. V. 82, N 1–4. P. 239–249.

A.V. Nikitin, R.V. Kochanov. **Vizualization and identification of spectra by the *SpectraPlot* program.**

The problem of high resolution spectra identification and application of program *SpectraPlot* for identification was considered. *SpectraPlot* is a freely distributed graphical program for spectra visualization and graphical assignment of high-resolution molecular spectra. Lines can be assigned graphically using the mouse and keyboard. Real time line list correction is allowed. Software can be freely downloaded at the URL: <http://www.iao.ru/spectraplot/sp.htm>. It is possible to display and manipulate experimental and simulated spectra, as well as stick spectra in any column-based text formats (including HITRAN and GEISA format). Examples of application of *SpectraPlot* for spectra identification are considered.