

РУКОВОДСТВО ПО РАБОТЕ С ПАКЕТОМ MIDAS (версия 2002 г.)

Содержание

1	Особенности работы с операционной системой UNIX	2
1.1	Файлы и файловая система	2
1.1.1	<i>Структура файловой системы UNIX</i>	2
1.1.2	<i>Владельцы файлов</i>	3
1.2	Некоторые команды системы UNIX	4
1.3	Архивация в системе UNIX	5
1.4	Работа в режиме удаленного терминала	6
1.5	Особенности работы в режиме удаленного терминала на сервере beta Мат.-Мех факультета СПбГУ	6
2	Функциональные возможности пакета MIDAS	7
2.1	Форматы данных в системе MIDAS	7
3	Запуск системы, общие команды	9
4	Работа в различных контекстах системы	9
4.1	Эшелле-спектры	9
4.1.1	<i>Эшелле-спектрографы</i>	9
4.1.2	<i>Настройка системы MIDAS на работу с эшелле-спектрами</i>	10
4.1.3	<i>Предварительная обработка</i>	10
4.1.4	<i>Нарезка эшелле-спектра</i>	13
4.1.5	<i>Отжествление линий</i>	14
4.2	Работа в других контекстах системы	16
5	Литература	16

Данное руководство предназначено для желающих быстро освоить работу с пакетом **MIDAS (Munich Image Data Analysis System)** для обработки данных наблюдений, проводимых различными методами с помощью различных приборов. Это руководство является лишь вводной частью описания пакета и содержит описание лишь главных команд. Для более детального ознакомления следует использовать указанную в конце руководства литературу, команду **help** при работе, а также обращаться на URL:

`<http://http.hq.eso.org/midas-info/midas.html>`. Некоторые дополнения к пакету (таблицы, подсказки и т.п.) могут быть получены через **anonymous ftp** на хосте **ftphost.hq.eso.org** (IP 134.171.40.2).

1 Особенности работы с операционной системой UNIX

1.1 Файлы и файловая система

Как и во многих операционных системах, в UNIX файлы организованы в виде древовидной структуры (дерева), называемой *файловой системой*. Каждый файл имеет имя, определяющее его расположение в дереве файловой системы. Корнем этого дерева является *корневой каталог*, имеющий имя `"/`. Имена всех остальных файлов содержат *путь* – список каталогов (ветвей), которые необходимо пройти, чтобы достичь файла. В UNIX все доступное пользователям файловое пространство объединено в единое дерево каталогов, корнем которого является каталог `"/`. Таким образом, полное имя любого файла начинается с `"/` и не содержит идентификатора устройства (дискового накопителя, **CD-ROM** или удаленного компьютера в сети), на котором он фактически хранится.

1.1.1 Структура файловой системы UNIX

Корневой каталог

Корневой каталог `"/` является основой любой файловой системы UNIX. Все остальные файлы и каталоги располагаются в рамках структуры, порожденной корневым каталогом, независимо от их физического местонахождения.

`/bin`

В каталоге `/bin` находятся наиболее часто употребляемые команды и утилиты системы, как правило, общего пользования.

`/dev`

Каталог `/dev` содержит специальные файлы устройств, являющиеся интерфейсом доступа к периферийным устройствам. Он может содержать несколько подкаталогов, группирующих специальные файлы устройств одного типа. Например, каталог `/dev/dsk` содержит специальные файлы устройств для доступа к гибким и жестким дискам системы.

`/etc`

В этом каталоге находятся системные конфигурационные файлы и многие утилиты администрирования.

`/lib`

В каталоге **/lib** находятся библиотечные файлы языка **C** и других языков программирования.

/lost+found

Каталог "потерянных" файлов. Ошибки целостности файловой системы, возникающие при неправильном останове UNIX или аппаратных сбоях, могут привести к появлению т.н. "безымянных" файлов – структура и содержимое файла являются правильными, однако для него отсутствует имя в каком-либо из каталогов.

/usr, /var

В этих каталогах находятся файлы и подкаталоги различных подсистем – электронная почта, справочники, словари и т.п.

/home

Общепотребительный каталог для размещения домашних каталогов пользователей. Например, имя домашнего каталога пользователя **nick** будет, скорее всего, называться **/home/nick**.

/tmp

Каталог хранения временных файлов, необходимых для работы различных подсистем UNIX. Обычно этот каталог открыт на запись для всех пользователей системы.

1.1.2 *Владельцы файлов*

Файлы в UNIX имеют двух владельцев: пользователя и группу. Группой называется определенный список пользователей системы. Пользователь системы может быть членом нескольких групп, одна из которых является первичной, остальные – дополнительными. Важной особенностью является то, что владелец-пользователь может не являться членом группы, владеющей файлом. Это дает большую гибкость в организации доступа к файлам. Совместное пользование файлами можно организовать практически для любого состава пользователей, создав соответствующую группу и установив для нее права на требуемые файлы. При этом для того чтобы некий пользователь получил доступ к этим файлам, достаточно включить его в группу-владельца, и наоборот – исключение из группы автоматически изменяет для пользователя права доступа к файлам. Увидеть атрибуты файла можно, задав опцию **-l** в команде **ls** (см. далее). Расшифровка первых десяти символов приведена в таблице.

Тип файла	Права владельца-пользователя	Права владельца-группы	Права остальных пользователей
-	rwX	r-X	r--
Обычный файл	Чтение, запись, выполнение	Чтение и выполнение	Только чтение

1.2 Некоторые команды системы UNIX

:	Пустая команда
.runme	Текущий командный интерпретатор выполняет команды, указанные в файле runme
cd [dir]	Осуществляет переход в каталог dir. Если параметр не указан, происходит переход в домашний каталог (\$HOME)
cd ..	Осуществляет переход в родительский каталог
echo [string]	Строка <i>string</i> выводится на стандартное устройство вывода (терминал)
exit	Завершает работу текущего интерпретатора
pwd	Выводит имя текущего каталога
ls [opt]	Выводит список файлов текущего каталога
cp [opt] file1 file2	Копирование файла file1 в файл file2
cp [opt] file1 ... dir	Копирование файла (файлов) file1 ... в директорию dir
mv [opt] file1 file2	
mv [opt] file1 ... dir	Аналогично для перемещения файлов
rm [opt] file1 ...	Удаление файлов
rmdir dir1 ...	Удаление каталогов
cat [opt] file	Вывод содержимого файла на экран
cat file1 >> file2	Запись данных файла file1 в конец файла file2
more [opt] file	Постраничный вывод
pg [opt] file	Постраничный вывод
head [-n] file	Вывод первых <i>n</i> строк файла
tail [-n] file	Вывод последних <i>n</i> строк файла

<code>find dir -name file [opt]</code>	Поиск файла <code>file</code> начиная с каталога <code>dir</code>
<code>chmod [opt] file1 ...</code>	Изменение прав доступа к файлу
<code>passwd</code>	Смена пароля пользователя
<code>mc</code>	Запуск оболочки Midnight Commander (аналог Norton Commander)
<code>startx</code>	Вход в графическую оболочку X Window System
<code>netscape</code>	Запуск программы Netscape Navigator
<code>mosaic</code>	Запуск программы Mosaic
<code>man command</code>	Получение справки о команде <code>command</code>

Подробную информацию о параметрах этих и других команд можно получить с помощью команды **help** *имя команды* или в литературе (напр., [1]). Важно отметить, что в отличие, скажем, от MS-DOS, система UNIX различает заглавные и прописные буквы. Поэтому, везде где особо не оговорено, следует набирать команды именно так, как указано в руководстве.

Работа в системе X Windows аналогична работе в среде Windows (создание-удаление окон, изменение их размеров, переключения и т.п.). Подробное описание этой системы имеется в [5]. Запуск пакета **MIDAS**, а также программы Netscape Navigator, возможен только при запущенной системе X Windows (режим `xterm`). При работе в компьютерном классе мат-мех факультета СПбГУ на сервере `beta` вход в оболочку X Windows осуществляется автоматически после ввода имени и пароля.

1.3 Архивация в системе UNIX

Подобно архивации в операционной системе MS-DOS, в системе UNIX также можно архивировать файлы. Наиболее популярными являются программы **gzip** (архивация) и **gunzip** (разархивация). Архивированный файл имеет расширение `.gz`, при этом, если производится архивация одного файла с каким-либо расширением, то по умолчанию у архивированного файла будет двойное расширение (например, `xxx.bdf.gz`). Запуск программ `gzip` и `gunzip` осуществляется командами

gzip имя-файла и **gunzip** имя-файла

На выходе получаем соответственно заархивированный или разархивированный файл. Следует отметить, что исходный файл (неархивированный или архивированный) не сохраняется, поэтому, если он вам еще нужен (например, как страховочная копия, его следует предварительно сохранить под другим именем).

1.4 Работа в режиме удаленного терминала

Операционная система UNIX позволяет удобно работать в режиме удаленного терминала, т.е. работая на своем компьютере, подключенным к сети через какой-либо хост-компьютер (сервер), можно фактически работать на самом сервере или на любом компьютере, подключенном к сети (если, конечно, у вас есть соответствующие права доступа), использовать имеющиеся там программы, находясь в своей директории. В частности, таким образом удобно работать с пакетом **MIDAS**.

1.5 Особенности работы в режиме удаленного терминала на сервере beta Мат.-Мех факультета СПбГУ

Сервер beta, на котором находится пакет **MIDAS** имеет полное имя beta.math.spbu.ru.

Для работе на нем в режиме удаленного терминала в командной строке следует вызвать команду

```
telnet beta.math.spbu.ru
```

В общем случае вводится либо имя сервера, либо его IP-номер. Затем вводится свой login, пароль, после чего работа на сервере происходит в обычном режиме. По окончании работы командой **bye** или **exit** возвращаемся на свой терминал. Отметим, что сервер beta доступен не с любого другого сервера. Из серверов Астрономического института доступ туда открыт только с сервера urania (urania.astro.spbu.ru). При работе на другом сервере можно сперва выйти на сервер urania (с помощью команды telnet или rlogin), а уже оттуда – на сервер beta.

При необходимости скопировать файлы с сервера к себе на компьютер или наоборот следует, выйдя из режима удаленного терминала, вызвать команду

```
ftp имя-сервера,
```

и после ввода имени пользователя login и пароля для получения текстовых файлов используем команды

```
get имя-файла или mget имя-файлов,
```

а для отправки файлов на сервер используем команды

```
put имя-файла или mput имя-файлов.
```

Если же надо послать или получить двоичные файлы, то перед запуском программ **mget** или **mput** необходимо набрать команду **bin**.

Также можно использовать обычные команды для просмотра содержимого директорий, перехода в другие директории и т.д. По окончании копирования выходим из программы ftp командой **bye**.

2 Функциональные возможности пакета MIDAS

Проводя наблюдения каких-либо объектов с помощью выбранных приборов и методик наблюдений, мы получаем спектры различных объектов. Но они сильно отличаются от реальных из-за множества искажений, вносимых при наблюдениях. Пакет **MIDAS** является универсальной программой, позволяющей максимально очистить спектры от искажений, каковы бы ни были приборы и методики наблюдений, и получить спектры, близкие к реальным, которые затем можно использовать для теоретических исследований.

В данном руководстве, в основном, мы опишем саму процедуру работы с пакетом. Для многих может представлять интерес, каким образом происходит сама обработка, какие методы для этого используются. Для этого в конце руководства приведена соответствующая литература.

2.1 Форматы данных в системе MIDAS

Существует несколько форматов данных в системе **MIDAS**. Определить, к какому типу формата относится файл можно по его расширению. Обычный файл с данными имеет расширение “.dat”. Его содержимое можно вывести на экран с помощью обычных команд операционной системы UNIX.

Другой формат представления данных – файлы с расширением “.tbl”. В системе **MIDAS** имеются каталоги линий для калибровки (находятся в директориях MID_название-инструмента) и таблицы спектров стандартных звезд (находятся в директории MID_STANDARD). При необходимости их можно скопировать в рабочую директорию обычным образом или с помощью соответствующей команды в данном контексте. Некоторые таблицы будут автоматически создаваться в ходе сеанса работы (например, в эшелле-контексте создадутся таблицы “order.tbl”, “back.tbl”, “line.tbl”). В них содержатся промежуточные данные, необходимые для дальнейшей обработки спектров. Получить информацию о типе имеющихся в таблице данных с помощью команды

show/table имя-таблицы,

а вывести данные на экран – с помощью команды

read/tab имя-таблицы.

Самые распространенные форматы представления данных в **MIDASe** – “.fits” и “.bdf”. В этих форматах представляются как исходные данные наблюдений, так и промежуточные и окончательно обработанные данные. В каждом конкретном случае мы работаем с одним из этих двух форматов; как перевод из одного такого формата в другой будет осуществляется командой

indisk имя-файла.bdf имя-файла.fits

indisk имя-файла.fits имя-файла.bdf

В первом случае из исходного файла формата “.bdf” создается файл формата “.fits” (исходный файл сохраняется), во втором – наоборот.

Следует отметить, что файлы формата “.bdf” могут быть несовместимы при работе на различных серверах или на персоналках. Поэтому при переносе файла данного формата с одной машины на другую следует сперва перевести файл в формат “.fits”, а после переноса уже на новой машине перевести файл обратно в формат “.bdf”. В дальнейшем мы будем говорить только о “.bdf”-файлах, имея в виду то же про “.fits”-файлы, если работа происходит с ними.

Для вывода содержимого этих файлов на экран, “.dat”-файл или для печати в **MIDASe** существуют команды

read/ima имя-файла [x_0 , y_0 , N_{pix}],

print/ima имя-файла [x_0 , y_0 , N_{pix}],

где x_0 , y_0 – начальные координаты (в двумерном случае), N_{pix} – число выводимых точек. В первом случае вывод осуществляется на экран, во втором – осуществляется печать на установленный принтер. Если есть необходимость преобразовать файл в обыкновенный файл данных (“.dat”-файл), то перед вызовом этой команды следует задать имя выходного файла командой

assign/print file (имя-файла).

Эта же команда используется и для возвращения к печати на принтер (вместо file следует ввести "lprint"). Аналогично командой

print/tab имя-файла [@N1 @N2]

можно перевести файл-таблицу в “.dat”-файл (также установленный командой

assign/print, N1 и N2 – номера первой и последней печатаемой строки).

Перевод из формата “.bdf” в формат “.tbl” осуществляется командой

copy/it имя-bdf-файла имя-tbl-файла,

а из формата “.tbl” в формат “.bdf”–командой

copy/ti имя-tbl-файла имя-bdf-файла.

Дополнительные возможности вывода предоставляет команда

extract/ctrace (/cursor,/image,/line...).

Возможности и параметры этой команды в определенном контексте можно увидеть, вызвав команду help.

3 Запуск системы, общие команды

Вход в среду **MIDAS** производится командой

inmidas или **inmidas -p n**,

где $n = 01..99$ – номер пользователя. Второй вариант необходимо использовать при одновременной работе в **MIDAS** нескольких пользователей, у каждого из них номер должен быть свой. Выход из среды **MIDAS** производится командой **exit**. При работе в среде **MIDAS** для вызова команды UNIXа (например, для просмотра имен файлов каталога) следует перед командой поставить знак \$ (например, \$ls).

4 Работа в различных контекстах системы

4.1 Эшелле-спектры

4.1.1 Эшелле-спектрографы

Спектрографы с эшелле – это дифракционные решетки с небольшим числом штрихов, употребляемые в очень высоких порядках – тридцатом, сороковом и пр. Если длина заштрихованной части решетки L , а общее число штрихов N , то формула дисперсии будет [3]:

$$2 \left(\frac{L}{N} \right) \sin \alpha = m\lambda, \quad (1)$$

а разрешающая сила R определяется по формуле

$$R = \frac{\lambda}{\Delta\lambda} = mN. \quad (2)$$

Угловая дисперсия для эшелле равна [4]

$$\frac{d\varphi}{d\lambda} = \frac{k}{b \cos \varphi} = \frac{2 \operatorname{tg} \varphi}{\lambda}. \quad (3)$$

При умеренном числе штрихов N благодаря большому m получается высокое значение R . Если m – большое, то в одном и том же направлении α в двух соседних порядках m и $m+1$ будут идти близкие длины волн: $\lambda_m : \lambda_{m+1} = (m+1) : m$. Если на пути образования спектра поставить призму с небольшим преломляющим углом, дисперсия которой перпендикулярна к дисперсии эшелле, то λ_m и λ_{m+1} , ранее совпадавшие по направлению, будут разделены поперек к направлению основного спектра. В итоге получается возможность всю желательную область спектра разбить на несколько десятков небольших отрезков, расположенных один под другим, в соответствии с размерами фотопластинки. При работе в контексте эшелле спектры будут выводиться на экран именно таким образом; для работы с ними необходимо знание абсолютных порядков, соответствующие интервалы длин волн можно найти в таблицах.

4.1.2 *Настройка системы MIDAS на работу с эшелле-спектрами*

Для работы с эшелле-спектрами в контексте Echelle системы *MIDAS* необходим ряд процедур, которые нужно разместить в каталоге `/midwork` пользовательской директории.

Переписать каталог `/lss` в каталог `/midas`. Эта директория содержит, в частности, тексты процедур на языке C, необходимых для работы части *MIDAS*-процедур.

Найти в `/lss` файл `makefile`, поменять в этом файле путь на файл `default.mk`. Запустить `makefile` из директории `/lss`, откомпилированные *.exe файлы запишутся в директорию `/midas.../stdred/exec`.

Процедуры `nectwice.prg` и `suntwice.prg` расположить в области стандартных эшелле-команд `/midas.../echelle/proc/`. Сделать ссылку на `nectwice.prg`:

```
ln -s .../echelle/proc/nectwice.prg nectwice.prg
```

4.1.3 *Предварительная обработка*

Как и большинство спектров, исходные эшелле спектры находятся в файлах с расширениями “.bdf” или “.fits”. Для загрузки пакета эшелле следует вызвать команду

set/context echelle.

В контексте эшелле существует программа, демонстрирующая пример работы с эшелле-спектрами. Ее вызов осуществляется командой

tutorial/echelle.

Описание процедуры обработки эшелле-спектров находится в [2]. Здесь мы приведем последовательность команд, осуществляющих данный алгоритм.

Первый шаг – создание дисплея для вывода изображения на экран. Это осуществляется командой

cre/disp N,xsize, ysize

Здесь N – номер окна (0,1,2,...), $xsize$ и $ysize$ – размеры окна по осям x и y соответственно. Пример вызова команды:

```
cre/disp 0,100,100.
```

Размеры дисплея в дальнейшем легко могут быть изменены с помощью мыши. Образовавшееся окно следует поместить в правый верхний угол экрана. В дальнейшем при запуске соответствующей команды изображение объекта будет загружаться в последнее созданное окно.

Для начала работы с изображениями, полученными при наблюдениях с системой **MIDAS**, т.е. в формате “.bdf”, “.bdf.gz”, необходимо запустить команду:

```
@@ biss файл E;
```

Для начала работы с изображениями, полученными при наблюдениях с системой **LIMA**, т.е. в формате “.fits”, “.fits.gz”, необходимо запустить команду:

```
@@ fiss файл E;
```

Эти команды аналогичны запуску следующей последовательности команд:

gunzip файл – если файл заархивирован;

fromlima файл – преобразование данных в формат float;

indisk файл – перевод в формат “.bdf” (для файлов формата “.fits”);

loa/ima файл – загрузка изображения в окно дисплея.

Информация о типе данных в файле, наблюдаемом объекте и т.п. содержится

в соответствующих дескрипторах “.bdf” или “.fits”-файла.

Для получения темновых изображений используется команда

```
aver/win dark = ln1,ln2,...lnn med,
```

где **dark** – получаемый усредненный темновой кадр. Вычитание среднего темнового кадра производится командой

```
@@ subdark lnn dark.
```

На выходе имеем файлы с именами “lnnd.bdf”. Эта процедура выполняется также с изображением калибровочной лампы (Th+Ar).

Построение маски для определения положения порядков. Используется команда

```
define/hough lnnd,
```

где lnnd – имя нужного файла (например, “l01d.bdf”), перед запуском которой следует установить параметры сессии командой

```
set/ech slit=5. wlcmtd=method wlc = file_ThAr width2=5 thres2=30,
```

file_ThAr – имя файла со спектром Th+Ar лампы: lincat=thar33.tbl (или thar50.tbl или thar100.tbl). Ширина порядка width2 и минимальное (пороговое) значение интенсивности линии в спектре Th-Ar лампы подбираются пользователем таким образом, чтобы получить оптимальное количество необходимых для отождествления линий.

Возможные методы – PAIR (используется при наличии накладывающихся линий в соседних порядках), ANGLE (при отсутствии наложений), TWO-D (более точный метод, но требует более точного отождествления), GUESS (при наличии нескольких наблюдений с одинаковыми или почти одинаковыми инструментальными параметрами), RESTART и ORDER (дополнительные методы). Значения параметров width2 (ширина линий) и thres2 (порог относительно фона) подбираются в каждом случае для выбора оптимального количества линий и таблицы стандартных линий: (thar33.tbl), (thar50.tbl) или (thar100.tbl). В результате выполнения данной команды имеем на выходе файлы “order.tbl” и “back.tbl”, используемые в дальнейшей работе со спектрами.

4.1.4 Нарезка эшелле-спектра

Для выполнения нарезки спектра нужно установить следующий параметр:

```
set/echelle BKGRAD=20,6,
```

последняя цифра – это расстояние между порядками по оси u в целых пикселях. Далее, записываем целочисленное значение nn в `keyword` с именем `NOBJ`:

```
write/key NOBJ/i/1/1 nn,
```

где nn – номер кадра с объектом.

Перед запуском процедуры нарезки можно “надеть” на изображение объекта маску с положениями порядков:

```
load/table order :x :yfit :order -1 ? 4
```

Если маска “сидит” не очень хорошо, ее можно подвигать вверх-вниз командой

```
@@ sh_ord,
```

указывая на запрос положительный или отрицательный сдвиг в целых пикселях. Прежде чем надеть на изображение новую (сдвинутую) маску, нужно очистить его от старой:

```
clear/chan over
```

Запускаем процедуру нарезки `redima.prg`:

```
@@ redima p5=nn
```

Эта процедура является интерактивной, поэтому нужно внимательно следить за ее выполнением.

На запрос программы “Enter `BKGSMO`–” обычно вводится среднее между значениями `mean` и `standart_deviation`, которые выведены на экране выше.

На запрос о переопределении высоты щели “Redefine slits? (y/n)” вводим u . Далее следует предложение программы сделать вертикальный разрез изображения. Его нужно сделать примерно по центру кадра так, чтобы не попасть на сильные линии. Сделанный разрез будет изображен в одном из графических окон. Нужно указать курсором левый и правый концы графика, щелкнув левой кнопкой мыши. Можно указать не сразу весь график, а разделить его на 2-3 части, тогда следующее действие нужно будет совершить соответственно 2-3

раза. Это действие состоит в следующем: в другом графическом окне появится вырезанный кусок графика, снова требуется указать его левый и правый концы.

Оба действия следует выполнять предельно аккуратно, т.к. программа “вылетит”, если вы “кликните” мышью за пределами графика, и процедуру `red-ima.prg` придется запускать заново.

Если все прошло успешно, то в одном из графических окон появится кривая, по которой можно судить о качестве переопределения высоты щели.

Наконец, в графическом окне один за одним будут выводиться “вырезанные” порядки обрабатываемого эшелле-спектра.

Последний запрос программы – ввести имя объекта. (Необходимо для дальнейшей работы с изображениями в DOS-пакете DECH20).

4.1.5 *Отождествление линий*

Для начала отождествления линий спектра звезды запускается команда

`calibr/echelle,`

в ходе выполнения которой в окно будет выведено изображение спектра сравнения. Необходимо провести процедуру привязки линий спектра сравнения к длинам волн. Для этого необходимо точно отождествить как можно больше (минимум 5–6) линий в различных порядках с имеющимися в таблице. В левом нижнем углу окна с изображением будут выдаваться координаты курсора и интенсивность излучения в данном месте. Отождествление производится путем сравнения положения линии и ее интенсивности с табличными данными. Выбранные линии помечаем с помощью мыши и запоминаем последовательность их выбора. Окончание выбора производится нажатием правой кнопки мыши. Затем необходимо в требуемой последовательности ввести длины волн данных линий (с максимальной точностью), которые берем из таблицы. После ввода длины волны первой линии вводим ее абсолютный порядок, порядки последующих линий программа вычисляет автоматически. После ввода длины волны последней линии программа осуществит процедуру привязки всех линий и будет выдана информация о проценте удачно отождествленных линий. В случае удачного отождествления на появившемся на экране рисунке будут видны ровные линии, если же они окажутся размазанными, то отождествление произошло плохо. В таком случае процедуру привязки необходимо повторить с большим количеством линий.

Следует отметить, что для отождествления необходимо выбирать только точно известные узкие резкие линии, выбор широких линий, а также неточное задание длины волны приводят к существенным ошибкам.

В результате выполнения процедуры **calibr/echelle** получаем на выходе файл "line.tbl". Все параметры сессии можно сохранить командой

save/ech имя-сессии.

В дальнейшем текущие параметры сессии можно изменять и записывать в другой файл с помощью той же команды. Для установки параметров предыдущей сессии используется команда

init/ech имя-сессии,

а с помощью той же команды без параметра имя-сессии можно установить первоначальные значения всех параметров. Текущие параметры можно посмотреть с помощью команды

show/ech.

Эту команду рекомендуется вызывать перед каждым последующим шагом работы.

Процедуру отождествления обычно повторяют, используя первое отождествление как первое приближение:

set/ech wlcmtd=GUESS GUESS= имя-сессии

calibr/echelle.

Работа с изображениями объекта. При наличии фонового изображения необходимо провести его оценку и вычитание командой

substr/back файл метод.

Методом может быть spline или poly. Выделение порядков производится командой

extract/echelle исх_файл получ_файл [метод].

Используемые методы – linear, average или optimal. От метода зависит придание веса каждому порядку (одинаковые в методе linear, обратно пропорциональные длине порядка в методе average и оптимизированные по отношению сигнал/шум в методе optimal). Здесь получают изображения в формате pixel - order, т.е. практически спектры. Чтобы связать их с полученным отождествлением, их переводят в формат wavelength-order по команде:

rebin/echelle исх_файл получ_файл шаг.

Здесь параметр **шаг** указывает на то, с каким шагом (в \AA) вычисляются интенсивности (его не следует брать большим, чем разрешающая способность инструмента, который записан в дескрипторе STEP). Для эшелле спектров обычно STEP=0.2.

Командой

merge/echelle исх_файл получ_файл

порядки можно склеить в один длинный спектр. Этот спектр – одномерный, т.е. содержит только интенсивности излучения. Соответствующие длины волн определяются с помощью дескрипторов этого файла (начальная длина волны – дескриптор START и шаг, заданный ранее в параметрах команды extract/echelle – дескриптор STEP). Значения дескрипторов можно просмотреть командой

read/descr имя-файла [имя-дескриптора].

Содержание же самого файла можно просмотреть с помощью обычных команд вывода данных. Так же можно преобразовать полученный “.bdf”-файл в “.dat”-файл или, сперва преобразовав его в “.tbl”-файл и затем осуществив преобразование в “.dat”-файл, получить на выходе таблицу (в первой колонке будут номера пикселя, во второй – интенсивности).

4.2 Работа в других контекстах системы

5 Литература

1. Робачевский А. Операционная система UNIX. ВHV–Санкт-Петербург, 1997. 528с.
2. ESO– MIDAS. Полное справочное руководство.
3. Мартынов Д.Я. Курс практической астрофизики. М.:Наука, 1977. 544с.
4. Дервиз Т.Е. Астрономические спектрографы. Л.:ЛГУ, 1985. 85с.
5. Такет Д., Гантер Д. Использование Linux. К.,М.,СПб, 1998. 576с.

ПРАКТИЧЕСКАЯ РАБОТА

Обработка эшелле спектров с помощью системы MIDAS

Задание. С помощью данного руководства провести обработку 5 последовательных эшелле спектров указанной звезды.

Спектры находятся в файлах ... в каталогах ... Дополнительно для работы понадобятся файл со спектром торий-аргоновой лампы (спектр сравнения) – в том же каталоге файл ..., таблица линий (файл thar100.tbl в каталоге ...) и несколько программных файлов. Перед началом работы эти файлы следует скопировать в рабочую директорию (желательно создать отдельную директорию для выполнения задания). Находясь в этой директории, следует войти в среду MIDAS и выполнить следующую последовательность команд:

```
set/context echelle
@@ biss file1 E
@@ biss file2 E
@@ biss file3 E
@@ biss file4 E
@@ biss file5 E
@@ biss filethar E
aver/win dark = l01,l02,l03,l04,l05 dark
@@ subdark l01 dark
@@ subdark l02 dark
@@ subdark l03 dark
@@ subdark l04 dark
@@ subdark l05 dark
@@ subdark lthar dark
set/ech slit=5. wlcmtd=ANGLE wlc=thar100.tbl width2=5 thres2=30
define/hough l01d
calibr/ech
```

В ходе выполнения этой процедуры провести отождествление.

```
extr/ech l01d l01e
rebin/ech l01e l01r 0.2
merge/ech l01r l01m
assign/print file l01.dat
print/ima <,20000 l01m
```

После этого ту же последовательность команд, начиная с `extr/ech`, выполнить с остальными четырьмя изображениями.