DYNAMIC SCHEDULING

M. C. H. Wright
Radio Astronomy Laboratory, University of California, Berkeley, CA, 94720

16 December 1999

Abstract

This memo discusses scheduling for the ALMA array. We discuss two levels of dynamic schedulers: those which allocate suitable observing time, and those which adjust the observing parameters. We describe a dynamic scheduling program which determines the observing schedule in real time from a prioritized list of observations and the system status including the weather conditions. This memo takes a retrospective view of the merits and problems of automatic dynamic scheduling. To make best use of ALMA, a dynamic scheduling program should be able to adjust the observing parameters according to the weather, but must do so in a way that leaves the human observers feeling in control of the observations.

1 Introduction

There are two traditional methods for scheduling observations. 1) **interactive** observations give control of the telescope to a user. 2) **fixed queue** observations proceed in a linear fashion through a previously prepared schedule. Interactive observing is appropriate for observations where the observer looks at the data and then decides what to do next on the basis of the results. A fixed queue is more appropriate for routine observations where the observer does not need to look at the observations in real time.

Traditionally, single dish observations have been more interactive, and array observations, with the need to acquire data for several hours before sufficient *uv* coverage is available to make an image, have been fixed queue. In both cases telescope time is awarded to individual observers who then try to make best use of the allotted telescope time, sometimes with their own (approved) backup projects for less favorable weather conditions. **Dynamic scheduling** takes the observatory's view in scheduling all the approved projects to make best use of the telescope under all conditions.

In order to make best use of the instrument we need to change the observing schedule according to the system status and the weather conditions. For example, if a project needs all antennas in order to obtain the required uv-coverage and some of the antennas are not available, then it is better to observe a project that does not need all the antennas. Similarly, if the weather is not good enough for the current project to succeed, then we make better use of the array by scheduling a project which can make use of the current conditions. We might call this **level 1** dynamic scheduling, where only the time allocation can be changed, but the observing procedures are specified within the observing scripts for each project. **Level 2** dynamic scheduling allows system control of the observing procedures (calibration sources, integration times and calibration intervals, etc) depending on the weather and system parameters.

A flexible, dynamic schedule can accommodate both fixed queue and interactive observing. Time critical observations can be scheduled by giving them high priority at the appointed time. For interactive observing, the user takes control of the telescope for a time interval after which control is returned to the scheduling program to determine the best project to observe. Interactive observing can be made more productive and less stressful. Interactive users who wants more time to evaluate their data can return control to the scheduling program, and come back (hours or days) later to continue.

Flexible scheduling procedures have been implemented at several observatories and are proposed for the ALMA array (MMA memo 164). To further develop these ideas we implemented a dynamic scheduling program for BIMA array observations. The design and implementation of this program is described in BIMA memos 53, 58, and 60. Sections 2-7 summarize this development. Section 8 takes a retrospective view of the successes and failures of this implementation of dynamic scheduling. Sections 9-10 discuss the prospectives for implementing dynamic scheduling on ALMA.

2 Time Allocation

The time allocation committee makes a prioritized list of the proposed observing projects according to the scientific goals and policies of the observatory. Each project is assigned a priority and a maximum allocated telescope time. Requirements for each project such as noise level, atmospheric seeing and opacity, *uv* coverage, LST or UT time range, can be specified by the proposer. The actual observing schedule is best determined in real time from the list of projects, their requirements, and the system status including the weather conditions.

3 Scheduling Program

In this section we describe an automatic dynamic scheduling program which was used on the BIMA array. The scheduling program, **taco**, is a level 1 dynamic scheduler which determines the observing schedule from a prioritized list, the current status of the observations, system status, and the weather. The scheduling program monitors the weather and switches projects if the weather is no longer suitable for the current project. If the weather turns favorable for a higher priority project in the middle of the night, the scheduling program can stop the current project gracefully and start a new one. Clearly, one does not want to jump in and out of a project too often. The scheduling program is guided by **decision thresholds** which control when to start and stop a project. These decision thresholds form a set of parameters which can be tuned to produce the best performance. The decision thresholds really determine the behaviour of the scheduling program. Two important parameters are the minimum interval for which a project can be scheduled, and scheduling interval used to re-evaluate the schedule. If the scheduling interval is long, then high weight is given to completing the current project, and the behaviour is similar to a fixed queue; once a project is started it may continue even if the weather becomes unfavorable. If the scheduling interval is short, switching projects is more likely if conditions change. Other parameters determine the choice of the next project at each scheduling interval, e.g. how much weight do we give to filling the schedule with the next available project, what is the minimum run time for an existing project, or when starting a new project, etc.

The scheduling program maintains a list of the current status of the observations for each project, the scientific requirements, and maximum time allocation. A project can be terminated when it has met its requirements, or when it would exceed its maximum time allocation. For many projects, *uv* coverage is important. Simply repeating the same LST range during suitable weather conditions may not satisfy the scientific requirements, so the scheduling program keeps track of the LST ranges which have been observed, and matches these to each project's requirements.

4 Schedule Planning

The scheduling program, **taco**, can also be run off-line with the real system status replaced by model inputs and weather statistics. A template schedule such as is usually produced for a fixed queue might assume perfect weather. We can use a history of the atmospheric seeing and opacity to make more realistic projections for a future schedule. e.g. projects which can be self-calibrated can be observed during periods with poor seeing, such as summer afternoons. The scheduling program can thus be used for planning how much time should be scheduled for projects which require good weather.

The program can be run on different lists of possible observations. For example we can run the program on a list comprised only of 3mm projects, or a combined list of 3mm and 1mm projects, in order to see how the projected schedule is likely to work out under different assumptions of system status or weather.

5 Implementation at BIMA

The scheduling program described above was implemented at BIMA in 1997 to replace a fixed queue. The scheduling program has as its input the prioritized list of the projects, the current status of the observations, and the weather. The output from the scheduling program is a record of the completed observations and weather, and a projected schedule for future observations. At any time, the system manager and the users can see a summary of what has been done, the current status, and the projected schedule for completion of the projects. The future schedule, as in the planning phase, can assume various weather projections. Since the same scheduling program can be run off-line we can tune the decision thresholds to be used on the telescope.

We started very simply, so that a project would run to completion unless there was substantial change in the system status or weather. Analysis of the weather statistics showed that a 1-2 hour scheduling interval was appropriate for Hat Creek weather. With a scheduling interval of 1-2 hours, the median continuous run time on the same project was 2-4 hours, so that an 8-hour track would be completed with 2-4 pieces. Weather permitting, the scheduling program was very efficient at completing projects. The highest priority projects are completed first. The schedule is filled in with lower priority projects which could run in poorer weather. Using the scheduling program, no observations were acquired during unsuitable observing conditions; all the data acquired were good, in contrast to the fixed schedule where the data might require heavy editing, or be completely rejected.

The schedule is determined mainly by the project priority and atmospheric phase RMS. High priority projects are completed first if the weather is acceptable. The behaviour of the scheduling program can be modified using the minimum interval for which a project can be scheduled, and scheduling interval used to re-evaluate the schedule. As a default these have been set to 1 and 2 hours respectively. Under unstable weather conditions, more efficient use of good weather may be obtained using shorter times. This behaviour can be simulated off line using the scheduling program with an atmospheric model.

The behaviour of the scheduling program mimics our experience; as the schedule progresses, the more difficult to acquire data (e.g. afternoons) remain incomplete, and the schedule becomes inefficient as larger blocks of unscheduled time appear at times of good seeing. Projects can be added to the schedule to fill these gaps.

6 Impact on user

The scheduling program, **taco**, is only a time allocation program; it does not adjust the observations specified in the users' observing scripts. The impact of dynamic scheduling (level 1) on the user was minimized by keeping the existing structure of the observing scripts. The observer prepares an observing script specifying which sources and calibrators are to be observed. System control of the calibration sources and calibration interval depending on the weather and system parameters is also desirable, but would have more impact on the user. The user specifies certain requirements for his project to be run. Currently these are the LST range, and in some cases the UT (e.g. for solar, VLBI, comets observations). These parameters are part of the requirements for the project to run. Obviously, as the user specifies more requirements, the project is harder to schedule. The user can run the scheduling program off-line on his project with model system and weather inputs to see the effect of the specified requirements on the scheduling of his project. The requirements might then be adjusted to make the project more likely to be scheduled. In practice, we fixed the minimum acceptable RMS seeing to $\lambda/10$ for both 3mm and 1.4mm observations of non self-calibration projects.

7 Impact on data archiving

The data from dynamic scheduling are more fragmented than with a fixed schedule where each project runs to completion, good weather or bad. Instead of 8-hr *tracks*, which are evaluated after the observations are completed, with dynamic scheduling the archive should only consist of data which meet the specified quality requirements. The system and weather data are stored with the *uv* data, so that the observer can select subsets of the data based on this information. The BIMA data archive has the capability of locating and retrieving multiple data sets obtained at different times, although this is less convenient than having just one dataset.

8 A retrospective view

The scheduling program as described above was a technical success but a political failure. The program scheduled the observations, kept track of the weather and maintained a log of what had been observed 24 hours a day, as designed, but the observers were not happy. Several factors contributed to their unhappiness, but the primary reasons were sociological. Telescope time which had been *awarded* was not in their control. Note the language here: telescope time is *awarded*; this the the reward for a successful observing proposal. If the weather became good enough to allow a higher priority project to run, then the scheduling program would schedule the highest priority project for which conditions were good. Astronomers are an egotistical bunch and do not willingly give up *their* observing time. Even if the weather deteriorates, observers are reluctant to stop observing. Old timers preferred to have continuous *uv-tracks* so that they are better able to evaluate which data to discard. With dynamic scheduling and correct on-line diagnostics all the data should be good. Another problem is that **taco** is driven by a prioritized list, whereas with a fixed schedule, all scheduled projects are equal in stature - and more acceptable to more people

There were technical problems with the interfaces to the existing procedures. The dynamic scheduling program was constrained to fit in with the established proposal information, observing procedures and data archiving.

The user's requirements were extracted from the existing proposal cover sheets. The user can list the acceptable LST range, but sometimes this was wrong. The scheduling interval on a 1-2 hour time scale, was long enough for several calibrator-source cycles, but was too short to accommodate lengthy passband and flux density calibration which many users still like to do within *their 8-hr track*, even though the system passband and flux density calibration is now better than can be obtained in a reasonable fraction of 8-hours. On the output end, the data are archived first by date and then by project, whereas a more convenient order for the user is by project and then date. There are several lessons here: 1) dynamic scheduling should be developed in conjunction with the rest of the system in order to have suitable interfaces. 2) The need to educate, not only the students, but also their advisors, on the capabilities of the system.

Flexible observing at BIMA has evolved into *manual dynamic scheduling* where the resident observer, expert or not, decides which project to schedule for the next 8 hours based on the weather and satellite phase monitor. One could argue that this is appropriate for a university array, part of whose mission is to train the graduate students who comprise the observers.

9 Prospective for ALMA

Given the lack of user acceptance of dynamic scheduling at BIMA, should we then give up? I think the answer is no. An individual observer would have little hesitation in scheduling his own higher priority project, should the weather become good enough (level 1 dynamic scheduling). Dynamic scheduling takes the same view for the observatory as a whole and is on watch 24 hours a day optimizing the schedule. Likewise an expert observer would adjust the calibrators and integration times based on the weather and feedback from the data itself (level 2 dynamic scheduling). Somehow we must leave the observer feeling in control of his/her project and not in the maw of some intangible scheduling program. Similar problems with remote observing are discussed in an entertaining article by Lockman (1993).

The need for dynamic scheduling with ALMA is greater. Given the speed and power of the instrument, the potential for optimizing the observing is much greater. ALMA will be able to measure the phase coherence across the array and find the best calibrator and integration times to use on-the-fly. The difference between an optimized observation, or not, can be the difference between success and failure on long baselines at short wavelengths. Certainly, an expert observer may be able to do better than an expert program, but an expert observer will not be at the helm 24 hours a day, and it would seem to be a mistake to waste telescope time doing the wrong observation at the wrong time. So, I would argue that an expert dynamic scheduling program (level 2) should be available, and should probably become the default mode of observing, unless an expert observer wants to take interactive control of the array (and be responsible for the results). However, the expert program needs to be like the obsequious servant, always available, but waiting politely; ready, willing, and able to take care of the observing for human observers who need a break to think about the meaning of life, or their data, to eat, sleep, or whatever.

10 Observing cultures

It is not clear how to accommodate the different observing cultures of the millimeter and submillimeter wavelength observers at the university arrays, single dishes, and national facilities, not to mention the non-radio astronomers who we want to be able to use ALMA.

I think the answer is tied to having a very versatile scripting language, which observers can use to control the system to a greater or lesser degree according to their expertize and wishes.

Over the years, BIMA's basic interferometer observation program has added various calibration cycles, mosaics, polarization switching, source switching, frequency switching....It works, and efficiently too, but a table driven observation seems to be much more flexible. The table could specify the 'state' of the instrument for a basic integration cycle. The state includes the source(name, coordinates, velocity etc), polarization, frequency, pointing position, correlator configuration, integration time etc. These state parameters describe the instrument from the point of view of the observer and should not include system parameters such as delay centers etc. Table entries also specify the calibrations, as needed.

The observer can then control the instrument interactively by requesting a particular state, or can build a script which steps or loops through a table. The tables should be ascii files which the user can build with more or less help from friendly 'expert' programs which know how to sequence observations. Standard tables can be used for many observations, rather like subroutines (or macros). The tables can be read and edited, and archived as a record of the observations.

Dynamic scheduling can be implemented by allowing the system to select some of the 'state' parameters. E.g. instead of specifying the source name for a phase calibrator, the observer could specify **cal=phase RMS 5deg**, or some such thing, as the requirement for a phase calibrator, and allow the system to select the best calibrator dynamically. The users retain control of which parameters are set by the system and which they specify directly. A close analogy is taking photographs with a camera which allows either manual or automatic setting for each of its controls. For interactive observations, the human observer can search for the best calibrator, integration times, etc, and then use them. If the observation is running from a script (remote observing in some people's terminology), then the observations can still be optimized and scheduled dynamically with feedback from the data and the weather if desired.

11 Conclusions

* Implementation of a flexible observing schedule is essential to make best use of the ALMA array.

* The telescope is treated as a valuable resource. We should avoid scheduling observations which will be discarded because they do not meet their requirements.

* We implemented an automatic dynamic scheduling program, **taco**. It maintains a summary of the observations done, the weather statistics, and the future queue of observations. It can also be run off-line to produce a template schedule. **taco**, is only a time allocation program; it does not adjust the observation parameters. The behaviour of the scheduling program is determined by decision thresholds which can be tuned to obtain the desired performance.

* Although a technical success, **taco**, was a political failure. In retrospect, it was exciting to write a simple program which could control the telescope 24-hours a day, but very un-exciting to be the observer when the decisions were being made by the program.

* To make best use of ALMA, a dynamic scheduling program should be able to adjust the observing parameters according to the weather, but must do so in a way that leaves the human observers feeling in control of the observations.

* A dynamic scheduling program should be developed in conjunction with the rest of the system in order to have suitable interfaces to the ob-

serving proposals, observing procedures, and data archiving.

12 references

MMA Memo 164: MMA Computing Working Group Report, Steve Scott, Darrel Emerson, Rick Fisher, Mark Holdaway, Jill Knapp, Lee Mundy, Remo Tilanus & Melvyn Wright, 1996. http://www.mma.nrao.edu/memos

BIMA Memo 53: Dynamic Scheduling for the BIMA array. Melvyn Wright, 27-feb-97, http://bima.astro.umd.edu/memo/memo.html

BIMA Memo 58. DYNAMIC SCHEDULING: Selection criteria and evaluation M.C.H. Wright & R.L. Akeson, 19-Jun-97 http://bima.astro.umd.edu/memo/memo.html

BIMA Memo 60. DYNAMIC SCHEDULING: Implementation at Hat Creek, M.C.H. Wright, Aug 1997, http://bima.astro.umd.edu/memo/memo.html

Felix J. Lockman, 1993, in *Observing at a Distance*, proceedings of a workshop on Remote Observing, Eds. D.T.Emerson & R.G.Clowes, World Scientific Pub. Co.