Hindawi Publishing Corporation
Advances in Astronomy
Volume 2010, Article ID 976890, 9 pages
doi:10.1155/2010/976890

Research Article

The REM Observing Software

Mauro Stefanon,[1] Stefano Covino,[2] Dino Fugazza,[2] Emilio Molinari,[2,3] Francesco D'Alessio,[4] Giuseppe Malaspina,[2] Luciano Nicastro,[5] Vincenzo Testa,[4] Gino Tosti,[6,7] and Fabrizio Vitali[4]

[1] *Edifici Instituts d'Invesitgació, Observatori Astronomic Universitat de Valéncia, Polígon La Coma, Paterna, 46980 Valéncia, Spain*
[2] *INAF-Osservatorio Astronomico di Brera, Via E. Bianchi 46, 23807 Merate, Italy*
[3] *INAF-Telescopio Nazionale Galileo, Fundación Galileo Galilei, INAF, Rambla José Ana Fernández Pérez, 7, 38712 Breña Baja, Spain*
[4] *INAF-Osservatorio Astronomico di Roma, Via di Frascati, 33, 00040 Monte Porzio Catone, Italy*
[5] *INAF-IASF, Via P. Gobetti 101, 40129 Bologna, Italy*
[6] *Istituto Nazionale di Fisica Nucleare, Sezione di Perugia, 06123 Perugia, Italy*
[7] *Dipartimento di Fisica, Università degli Studi di Perugia, 06123 Perugia, Italy*

Correspondence should be addressed to Mauro Stefanon, mauro.stefanon@uv.es

Received 26 June 2009; Accepted 12 January 2010

Academic Editor: Joshua S. Bloom

The Rapid Eye Mount (REM) is a 60 cm robotic telescope located at La Silla, Chile. Its Observing Software (REMOS) is constituted by a set of distributed intercommunicating processes organized around a central manager. Together they grant the system safety, automatically schedule and perform observations with two simultaneous cameras of user-defined targets, and drive fast reaction to satellite alerts. Subsequent data reduction is left to pipelines managed by each camera.

1. Introduction

The discovery in the last decades of astrophysical processes which develop on timescales of few seconds to few minutes has given rise to the necessity of instruments capable of collecting as much information as possible on those timescales. On the other side, the technological developments in informatics of the last 20–30 years have enabled the astronomical community to develop telescopes with a high degree of autonomy. The scientific needs of rapid response to astrophysical processes have thus obtained one of the required tools, that is, *robotic telescopes* which can manage to acquire photons coming from a precedent unknown source position in a completely unattended way.

The Rapid Eye Mount (hereafter REM) telescope is one of the main robotic telescopes dedicated to the study of Gamma Ray Bursts (GRBs) afterglow. Its implementation allows to automatically point towards a new GRB in 30 seconds from the reception of the alert.

The system has already been described in [1–3]. In the following paragraph we outline its overall design. The next section describes all the strategies implemented in the software, while Sections 5 to 12 report on the technical implementation issues.

2. Scientific Aim

REM was installed at La Silla in June 2004 and became fully operational in the second half of 2005. Since its commissioning, the scientific drive of the project was the study of the early phases of GRB afterglows. This concretized into the realization of a software system able to analyze in real time all the alerts coming from the satellites, like HETE II, INTEGRAL, SWIFT, and, recently, Fermi, and to rapidly and automatically point the telescope to the source location. When no alerts are active, REM in realtime draws and executes the schedule for the night based on *secondary science* targets.

Starting from October 2006 REM has also become an Italian national facility under INAF (Istituto Nazionale di Astrofisica). Within this new context, REM is now offered to the scientific community with two Calls for Proposals per year.

Since the operations involved in the observations are quite different depending on the fact that the system is reacting to satellite alerts or not, it is thus natural to divide the kind of observations in two classes:

(i) satellite alert fast reaction,

(ii) observation of known-position sources. A generalization of this point is the possibility of activating a Target of Opportunity (ToO) request. This will be discussed in Section 6.

3. General Description of the System

3.1. Telescope. The Rapid Eye Mount REM is a 60 cm robotic fast slewing telescope located at La Silla site of the ESO La Silla Paranal Observatory, in the Chilean Ands.

It has a classical Ritchey-Cretien optical scheme, with a primary mirror of 60 cm diameter at f/2.2 with a total equivalent focal ratio of f/8 on the focal plane.

The mount is a compact alt-azimuth with two Nasmyth stations. One station hosts two instruments, a high-efficiency IR camera (REMIR) and an optical slitless spectrograph (ROSS), which can operate simultaneously thanks to a beam splitting dichroic mirror; the second Nasmyth focus is currently not used by any instrument. The compact design of the mount allows the telescope to slew as fast as 10 degrees per second. Although the telescope is not equipped with a guiding systems, the high precision ETEL encoders allow to track for about five minutes keeping the stellar spot within its average 2 arcsec full width half maximum (fwhm).

The telescope enclosure is composed by a two-sided roof, which can slide apart, leaving the telescope able to point to a minimum altitude of 5 degrees above the horizon.

3.2. Cameras. The REM Infrared Camera ([4, 5]) is equipped with a Rockwell 512 × 512 pixels, 18 μm pitch HgCdTe array, and cooled by liquid nitrogen. An optical system allows to obtain a FoV of $9.9 \times 9.9\,\text{arcmin}^2$. The camera has a 10-position filterwheel, with standard J, H, Ks plus y, and H2 filters.

The REM Optical Slitless Spectrograph (ROSS [6]) is a commercial Apogee CCD camera, with a 1 K × 1 K detector. The optics allow to cover almost exacly the same field of the IR camera (namely, $9.54 \times 9.54\,\text{arcmin}^2$ FoV). Apart from standard Johnson V, R, I filters, the filterwheel also holds an Amici prism, which allows to obtain slitless spectra of all the objects in the field down to a magnitude of $R \approx 14$ for an exposure of 300 s and S/N = 5.

Since August 2006, REM is also hosting the TORTORA camera ([7, 8] this proceeding). The instrument is dedicated to the early discovery and study of optical transients; mounted in parallel to the REM optical tube, it is constituted by a wide field (20 deg × 10 deg) fast photometry photomultiplied optical camera, allowing to obtain the photometry of the observed region of sky every second down to $R \approx 9$.

4. Overall View of the Software Structure

The diverse operations performed by REM translate to a great number of algorithms necessary to their implementation. In fact, as a robotic telescope, REM does not only automatically perform a set of observations, but it also continuously monitors itself and, to some degree, it can also self-diagnose and react to malfunctions, allowing, in some cases, to automatically continue the observation although in a degraded mode.

The complexity of operations is on the other part efficiently implemented through a set of intercommunicating programs, each one dealing with one (or in the most complex cases, even only a part) of the many aspects of the REM activities.

Another advantage of the adoption of a modular organization of the tasks is that in case of software failure, only the corresponding module is affected, leaving all the other programs running, thus minimizing the risks deriving from such problems.

Figure 1 represents a general scheme of all the processes constituting the REM Observing Software. They can be divided into logic groups: management of dome, management of meteorology data, management of cameras, management of satellite alerts, management of the telescope, system logging and communication with home, and synchronization of all the operations. Each one of these units will be described in the following sections.

As a general idea, all the modules relate to a specific one, named the REM Manager (REMMNGR). This program collects the data coming from the other modules, processes all the relevant information, and takes a decision about the next operation to perform. The number of actions includes the following: observe new sources, evaluate the figure of merit for a GRB alert, and interrupt an ongoing observation as a response to a transient alert.

Figure 2 shows a block-diagram of the idle cycle of the REMMNGR process. The system is continuously listening for new alerts. While no alert is received and the environmental conditions are good enough, a target selected from the accepted scientific programs is observed. As soon as a new alert is received, any possible ongoing observation is interrupted and the telescope is pointed towards the new source.

The complete list of all the processes together with a brief description is available in Table 1.

The vast majority of the modules have been implemented with the python language, with only few exceptions (REMSCHDLR, REMOBS) which are entirely written in ANSI C.

Despite the fact that most of the modules run on the same machine, communication among them is achieved through the TCP/IP protocol; in particular the YAMI [9] library has been chosen for its portability (it is available under all the most common OS), its availability for many common programming languages (C/C++, python, Tcl), its compactness, and for the low amount of resources required for running it.

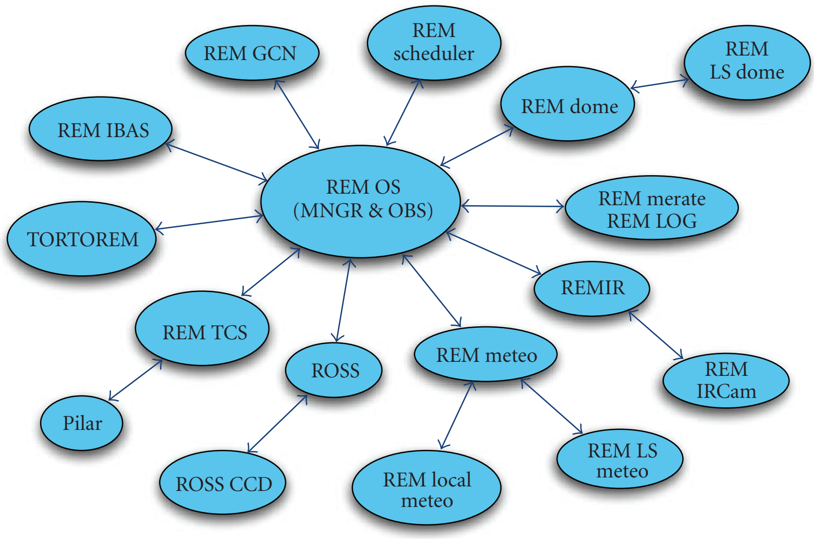


Figure 1: The set of all the REMOS processes can be organized into several groups: management of dome (REMDM, REMLSDM), management of meteorology data (REMMT,REMLSMT), management of cameras (REMROSS,REMIR), management of satellite alerts (REMGCN,REMIBAS,TORTOREM), management of the telescope (REMTCS), system logging and communication with home (REMMRT,REMLOG), and synchronization (REMMNGR).

Table 1: List of all the modules which constitutes the REMOS system.

Module name	Purpose
REMGCN	GCN network connection
REMIBAS	IBAS network connection
REMSCHDLR	Short-term scheduling of observations
REMDOME	REM dome management
REMLSDOME	Status of the La Silla domes
REMLOG	Logging management and alert routines
REMIR	NIR camera management
ROSS	Optical camera management
REMMT	Local environmental parameters management
REMLSMT	La Silla meteorological station connection
REMTCS	Connection with the Telescope Control System (TCS)
REMMNGR	General manager of operations
REMOBS	Drive ongoing observations
REMPLR	Evaluation of the status of the low-level telescope software
TORTOREM	Communication with the TORTORA camera

5. Alert Management

Since one of the main features of REM is the ability of rapid reacting to satellite alerts, great care has been put into the management of all the kinds of alerts sent by the number of high-energy satellites available to date.

The main source of alerts is the GCN system. This system is in charge of providing alerts generated by the HETE II and Swift satellites. Communication with this system is achieved by the REMGCN process. This program encapsulates the always active socket server necessary to receive the different classes of messages dispatched by the GCN central engine over a TCP/IP network connection, and to communicate the receipt of the alert to the REM central manager, which will chose its action depending on various parameters.

The other permanent source of alerts is the IBAS network, developed for the Integral satellite. The process in charge of keeping the communication with this system is REMIBAS; in this case, the socket connection follows the UDP standard and makes extensive usage of a public library developed by the Integral Team [10] and with whom REM also actively collaborated as beta tester.

Apart from merely listening for new alerts, both the modules above incorporate algorithms capable of detecting network failures and of recovering from these classes of problems, trying to reestablish the corresponding connection as soon as the conditions are restored.

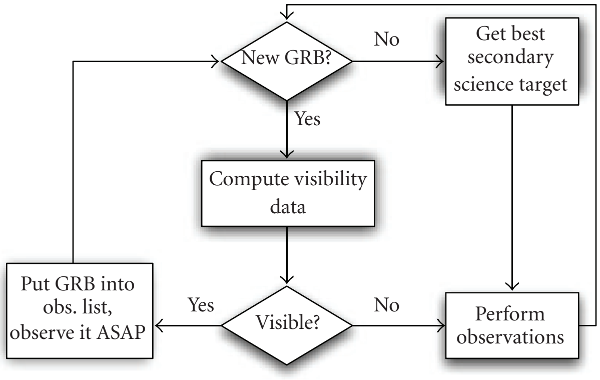

New GRB?
No
Get best secondary science target
Yes
Compute visibility data
Put GRB into obs. list, observe it ASAP
Yes
Visible?
No
Perform observations

Figure 2: Block diagram of the idle cycle of REMMNGR.

6. Source Observation

Since REM is now a national facility available to the scientific community, the common way REM operates is through the submission of a proposal for observation. The rapid reaction to satellite alerts is managed at this phase just as one of the observing modes offered by the telescope.

As a first step, the proposal submitted by the PI will be judged by the same INAF panel evaluating proposals for all the other Italian astronomical facilities, like the Telescopio Nazionale Galileo. Once the proposal has been accepted, for each required target a corresponding target is created and added to the queue. Here two processes are in charge of selecting which target to observe during a specific night at a given time: *OB manager* (REMOBS) and *short-term scheduler* (REMSCHDLR).

6.1. Target Archiving and Organization. The way the REM system manages the observation is through what is called *Observation Block* (hereafter OB). OBs are the unit of information for the observation and they can be considered as an adaptation of the ESO P2PP OB scheme. (The Phase II Proposal Preparation (P2PP) is an ESO program used to create and manage Observation Blocks for all ESO instruments.) As such, the information they carry can be organized into 5 categories: source type, telescope preset, instrument setup, observing conditions, and PI information.

The *source type* section contains data about the kind of target (e.g., if it is a star, AGN, GRB, etc.) and the origin of the target information (e.g., GRB alert, manual insertion of target, etc.). This influences the way targets are considered by the observing software; that is, they act as a first-level target priority.

Telescope preset data encapsulates all the relevant data needed, in the most general case, for deciding if pointing the telescope to the given position or not; in particular, this class of data includes information on Right Ascension and Declination, the equinox to which the coordinates refer and the error box of the coordinates. While this last parameter is generally not used for fixed, previously known objects, it is of fundamental importance for targets coming from satellite alerts. In fact, the central algorithm applies a figure of merit on each alert which excludes from following those alerts whose error box is much bigger than the field of view of the REM cameras.

The way images are collected is defined in the *instrument setup* section. Here it is possible to define which sequence of filters to use, the exposure time, and the number of desired repetitions.

The conditions under which each target should be observed are organized in the *observing conditions* parameter set. These include both information on the sky conditions (like moon fraction, sky transparency, and airmass) and constraints on the allowed epoch during which the target should be observed. In addition, it is also possible to set a periodicity of observation and to link two or more targets in order to get them observed during the same night (grouped targets). Finally, each target is assigned a priority which drives the order in which targets are observed during the night. Section 6.3 explains in more detail this process.

After the completion of each observation, REM automatically informs the PI of the program via email with a preview of the acquired (for the optical camera) and prereduced (for the IR camera) images. The data necessary for this is stored in the *PI information* fields.

Each OB is initially ingested into a general PostgreSQL database physically located on a machine in Italy and duplicated on the corresponding machine in the dome. This database thus archives all the targets foreseen for the running public proposal period (AOT). A process, called REMOBS and described in the next section, has the duty of selecting, by putting them into a shorter archive, those targets which are best observable in the following 3–5 nights and which will be used by the short-term scheduling software to draw the night of observation schedule.

6.2. OB Manager. To make the scheduling operations of REM telescope easy, the REM team has developed a password protected interface that provides the insertion of an OB through a web form. This interface allows to perform the most common activities on the OB database, namely, list all the OBs currently in the archive, insert and remove OBs, and modify existing OBs (see Figure 3), in the view of a mid to long-term scheduling.

The main activities which can be executed through the web interface are as follows.

(i) Seasonal optimization: this includes automatic selection of all and only OBs relevant for the ongoing AOT.

(ii) *Syncopal* scheduling: although the short-term scheduler, presented in the coming section, is capable of allocating targets with a fixed periodicity during any number of nights, for some targets it may be advantageous to allocate them with nonconstant (syncopal) periodicities.

(iii) The third one is the reallocation of already observed targets whose constraints were not properly satisfied during the execution.

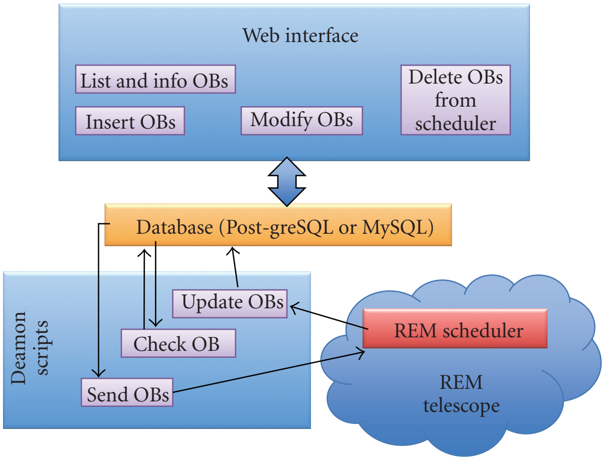


Figure 3: Schematic view of REMOBS operations.

The synchronization between the general and the local database is maintained by three processes (REMOBS_performed,REMOBS_send, and REM_ntimes), which are run daily in an automatic way through daemon scripts from the server located in Merate. The REMOBS_performed module is in charge of collecting all the information on the observations performed during the previous night, while the REMOBS_send sends the updated list of OBs to the local archive. Finally, the REM_ntimes process checks for syncopal periodicities of the OBs and acts as *garbage collector* for the general archive, removing targets already observed a preestablished number of times.

The whole system is based on different perl scripts interfaced with a PostgreSQL database and Common Gateway Interface (CGI) library. The web form is dynamic and all the options are modified depending on the selected semester Call for Proposal. After inserting the password, the REM team can choose which interface to work with and the operation to use (see Table 2).

6.3. Short-Term Scheduling. The decision about the observation of any specific target during the current night is taken by the REMSCHDLR module. This process is called by the REMMNGR at the beginning of the night, before and after the observation of each target and every time a new target is inserted into the archive. The output of this process is an ordered and time-defined list of targets for the current night.

The REMOS system is able to automatically detect camera malfunctioning and this information is also intercepted by the scheduling software which can select in real time only those targets which do not need the temporary unavailable camera. A further benefit of having a dynamic scheduling against a static one, computed once at the beginning of the night, is that it allows to schedule targets which most take advantage of the changing sky conditions.

Figure 4 shows the block diagram of the central engine of the scheduler. The basic idea is to try to allocate each target when it is best visible, and respecting all the timing and visibility constraints, as defined by the user.

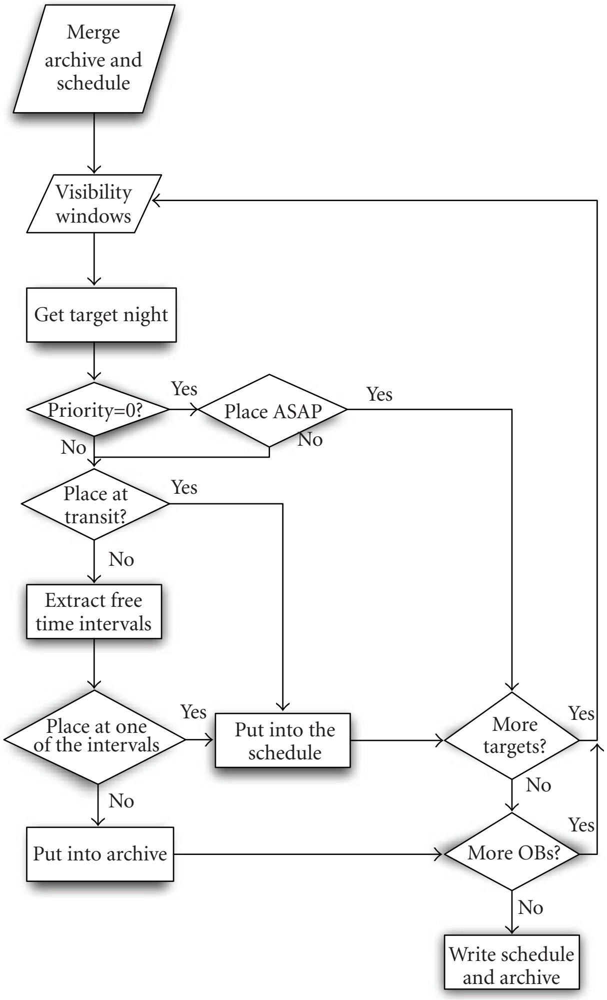


Figure 4: Block diagram of the REMSCHDLR core.

As a first step, the current short-term archive and schedule are merged and targets are sorted according to their priority. The algorithm then collects all the time intervals which are wide enough to allow the whole observation and which also satisfy all the visibility constraints. If the target has the highest priority, then the chosen time interval is the one which follows the *observe-as-soon-as-possible* strategy. This is the typical case for observations of GRBs whose location in the sky was not immediately reachable by the telescope at the time of the alert. For all the other priorities, the program sorts the collected time interval according to their distance from the time of transit through the meridian and tries to allocate the target starting from the interval closer to the transit time.

Starting from 2006 REM also accepts Target of Opportunity (ToO) requests. In the spirit of robotic operations, ToO is also completely managed in a fully unattended way. The activation of a ToO consists in an email sent by the PI to a machine located in the REM dome and containing all the information necessary to create the corresponding OB. A dedicated process then creates the OB in the archive; the

Table 2: The list of the possible operations with REMOBS interface.

Interface	Operation
List OBs	Visualize OBs status for the chosen programme.
Modify OBs	Modify OBs previously inserted for the chosen programme.
Insert OBs	Insert new OBs for the chosen programme.
Delete OBs	Delete OBs from REMSCHDLR.

schedule is computed as soon as the ongoing observation is completed and the target observed as soon as possible.

Apart from dealing with scientific targets, the scheduler is also capable of managing some classes of calibrations, namely, sky flat field, standard star fields, and focus fields. In these cases, the REMSCHDLR program only allocates the time for the observations, leaving the choice of the particular field to the REMFIELDS, REMSTD, and REMFCS, respectively, for the sky flat field, the standard star field and the focus field. In particular, flat fields are observed every morning, with the strategy left to each of the camera, although a typical session consists in taking 3 sky flats per filter; for standard stars are allocated once per night with 2 optical standards and one for the infrared and the focus field is observed once per night. The detailed strategy for focusing is explained in the next section.

6.4. Focusing. Good focus is a nontrivial task for a robotic facility; in the case of REM this is complicated by the fact that there are two cameras which observe at the same time and thus need to both have a good focus position. This is achieved through a double procedure.

(i) The infrared camera focus is directly managed by moving the M2 mirror.

(ii) Focus for the optical camera is adjusted via a series of neutral optical elements in the pupil and with different optical lengths, with the effect of shifting the focal plane.

Each night a set of 11 *focus frames* are taken, each one with a different M2 offset value, distributed around what is considered the best value. This set is then analyzed during the following morning and the focus value is obtained by fitting a parabola to the 3-σ clipped fwhm values of the star along the major and minor axis of the stellar image. The value obtained for the focus position is then archived in a database together with the environmental temperature; the full dataset contained in the database is finally used to update the parameters of the temperature-focus relation via a minimum χ^2 fit and after applying a 3-σ clip to the data.

Immediately before the beginning of each observation, the REMMNGR process sets the value of the focus position according to the environmental conditions.

As a further possibility, for the observation of very bright objects, the definition of a focus offset for source defocussing is also offered.

7. Cameras Management

Each camera is controlled by a dedicated computer (instrument machine) on which the low-level C/C++ software runs.

One of the most important features of REM is the ability to synchronize the observations with the two cameras. The synchronization is managed by the REMMNGR process which sends the relevant information to REMIR and ROSS modules, respectively. These two modules have a corresponding *twin* module on the instrument machine whose purpose is twofold.

(i) Dispatch the observation information from and to the low-level software of the camera. In general, this includes sending the beginning of the observation, waiting for its completion and, in case an alert is received, sending an abort message to the camera, in order to stop the current observation and start the sequence which best fits the kind of source of the alert.

(ii) Keep a constant check on the status of the camera. In case an erroneous state is detected, the main manager is immediately informed and subsequent operations are taken in accordance to the new status.

Being an external project, the impact on the software side of the TORTORA telescope is minimal, although necessary. The TORTOREM process is in charge of constantly informing the main TORTORA software about the ongoing observing conditions and in particular the position REM is pointing to and if a GRB alert is active.

8. Telescope Control System

The intermediate and low-level controls of the telescope are provided by the *Pilar* software, produced by the German company 4Π Systems Gmbh and running on a dedicated machine. This software provides a socket server with all the high-level commands necessary to point the telescope to a given coordinate set, start/stop the tracking of the telescope, and move the derotator to the desired angle.

Similarly to the way communication with the cameras is implemented, also for the telescope control system a dedicated process, REMPILAR, runs on the telescope machine. This python-written module both manages the communication of the commands to and from the Pilar server and it also periodically checks that Pilar is correctly running. When this is not the case, the module automatically tries to restart Pilar and, if after a number of attempts the problem still persists, it sends an email informing the team of the problem;

an analogous algorithm is implemented by REMPLR also for the detection of the general telescope safety activation which may occur when local technicians from the observatory need to access the telescope for routine mechanical and cryogenic maintenance.

9. Meteorology

The acquisition and analysis of weather conditions is one of the fundamental activities to keep the observatory in a safe status, especially when operations are managed in an autonomous way.

REM is continuously checking meteorology data coming from two distinct sources; in fact, apart from having its own meteorology station, it can also access the data of La Silla weather station. To this aim, the two processes REMLSMT and REMMT are collecting data every 5 minutes, respectively, from La Silla meteorology data and the REM local station. These processes are in charge of merely collecting the data, leaving the decision process to the central engine of the system, that is, the REMMNGR tasks.

The possibility of accessing an external, but still local, meteorology station, like the one of La Silla, has a double advantage: on one side it provides an independent estimate of the meteorological conditions and on the other side it offers the necessary level of redundancy in case of failures of the REM local station.

The local weather station is a Davis Vantage Pro device, with sensors measuring the temperature and humidity inside and outside the dome, the wind speed, and direction and the rain rate.

For weather conditions being considered safe, both REMLSMT and REMMT processes implement the following limits which must be satisfied at the same time.

(i) Rain rate must be zero.

(ii) Wind speed must stay below 15 m/s.

(iii) The relative humidity inside and outside the dome must be below 90%.

The decision algorithm implemented in the REMMNGR considers meteorological conditions as safe for operations if the above limits are entirely respected by at least one of the two sources of information.

10. Dome

Since the telescope does not implement any kind of protection for the primary mirror, the dome is the only way the REM observatory has to keep its instrumentation safe from environmental conditions. Great care has thus been put in the algorithms which control its opening and closing phases. Furthermore, the design of the dome is such that it is possible to open and close it without the need of having the telescope in a particular position, assuring the fastest reaction to sudden changes of weather conditions.

In addition to the constant check of the meteorology conditions, the REM system relies on the status of the domes of the other telescopes at La Silla. The REMLSDOME module is in fact able to gather the data about the opening status of the ESO DIMM, NTT, 3p6, 2p2, and the Swiss Heuler telescopes.

A figure of merit based on the status of the above telescopes is then applied in order to establish if the REM dome can be opened. Apart from satisfying the meteorological constraints described in the previous section, the REM dome can be opened when.

(i) the DIMM protection is open;

(ii) at least another dome appears as open.

As a last aid to the safety of the telescope, if, for any reason, the REMLSDOME process can not access the status of La Silla dome, then the REM dome is kept closed and everything is automatically resumed as soon as the status can be accessed again.

11. GPS and Time

The accuracy of time measurement is a fundamental issue for a robotic observatory like REM for two reasons. In fact, a correct timing not only is essential for an accurate pointing and tracking of the telescope but also has a scientific value.

The ability of rapidly reacting to satellite alerts for astrophysical processes whose typical times are of the order of few seconds means that the information stored together with the observations must allow to perform an as much accurate as possible timing analysis of the phenomenon itself. Hence information such as the time of the reception of the alert and the time of beginning and end of each observation must reflect this need.

To satisfy the need of accuracy, the REM observatory adopts a double source for time synchronization. On one side, the GPS antenna installed on the external wall of the dome provides an accurate time signal through the Pilar program. On the other side, all the machines in the dome are synchronized to the La Silla time server via the *Network Time Protocol*. On top of these two methods, the module REMGPS continuously checks that the time provided by the GPS antenna is compatible with the one on the computers. In case a difference greater than 8 seconds is found, the process sends an email to the team informing of the problem.

This procedure already allowed to discover, and correct, an anomaly with the GPS antenna.

12. System Data Logging

Although the software system implements many algorithms which allow to limit the operations in case of problem and, for the best cases, also to automatically recover from an error state, a detailed system logging is fundamental tool of diagnosis for all the other cases.

Each REMOS process sends its logging information to a central module, the REMLOG, which locally archives the log data and immediately also duplicates it to an external server located in Merate, Italy.

Another fundamental aspect is the supervision of all the parameters of the instruments and of the telescope, that is, the *system telemetry*. The tREMometer (see Molinari et al.,

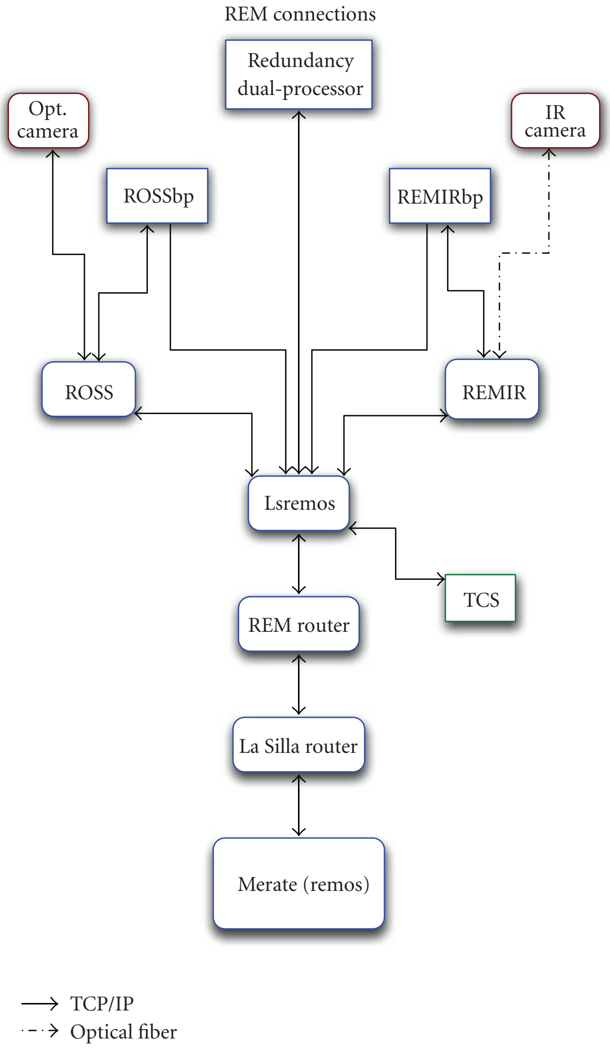


Figure 5: Organizational chart of the REM local network.

this issue [11]) is an articulated software which checks the status of the instruments and of the REM software and generates a publicly available almost real-time resume.

13. System Hardware

Figure 5 shows the organization of the running hardware inside the dome. The REMOS software has demonstrated to be capable of being easily run on commercial machines, which makes maintenance an easy task.

Most of the modules run on the same machine, an AMD i686 single processor with Linux Fedora 8. The optical and infrared camera are managed by a dedicated dual processor each, to grant the speed necessary for the real-time prereduction pipelines.

Communication with serial devices is implemented through two MOXA devices, which provide an ethernet access to serial ports.

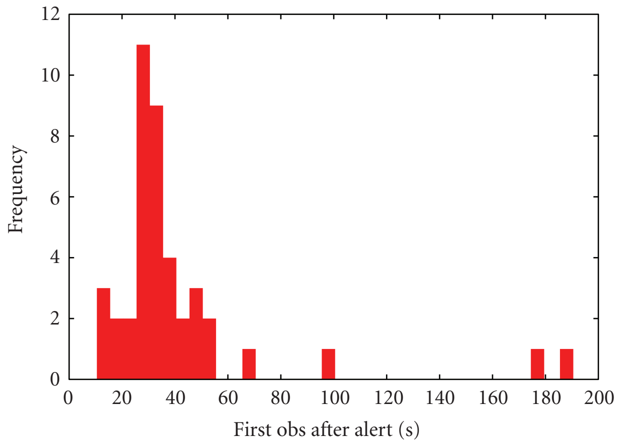


Figure 6: Histogram of the time necessary to REM to point a new GRB alert. The median time is of 30 seconds.

14. Results and Conclusions

Figure 6 shows the histogram of the pointing time for all the GRB alerts received by REM in 2007. The median value is 30 seconds; this value mostly raises during the accelerating and decelerating phases of the pointing. This result is a confirmation that, on the software side, a strict real-time implementation is not necessary, being most of the dead-time on the electromechanical part.

To date, more than 100 GCN have been published reporting prompt observation of GRB made automatically by REM. This is also made possible by the high level of stability reached by the whole software operating the telescope.

In this article the software managing the observations of the REM telescope has been presented. The requirement of autonomously reacting to satellite alert together with the necessity of synchronizing the observations of two independent cameras makes the software requirements more demanding than in other cases. The strategy for the implementation was to keep as much separate as possible the distinct kinds of operations of the telescope. The results obtained so far by REM are the clearest proof of the effectiveness of the adopted solutions.

Acknowledgment

The authors wishes to thank Maciej Sobczak for providing the YAMI library and for his continuous support to it.

References

[1] F. M. Zerbi, G. Chincarini, and G. Ghisellini, "The REM telescope: a robotic multiwavelength facility," in *Ground-Based Instrumentation for Astronomy*, A. F. M. Moorwood and M. Iye, Eds., vol. 5492 of *Proceedings of SPIE*, pp. 1590–1601, Glasgow, UK, June 2004.

[2] G. Chincarini, F. Zerbi, A. Antonelli, et al., "The last born at La Silla: REM, the rapid eye mount," *The Messenger*, vol. 113, pp. 40–44, 2003.

[3] S. Covino, M. Stefanon, and G. Sciuto, "REM: a fully robotic telescope for GRB observations," in *Ground-based Instrumentation for Astronomy*, A. F. M. Moorwood and M. Iye, Eds., vol. 5492 of *Proceedings of SPIE*, pp. 1613–1622, Glasgow, UK, June 2004.

[4] P. Conconi, R. Cunniffe, F. D'Alessio, et al., "The commissioning of the REM-IR camera at La Silla," in *Ground-Based Instrumentation for Astronomy*, A. F. M. Moorwood and M. Iye, Eds., vol. 5492 of *Proceedings of SPIE*, pp. 1602–1612, Glasgow, UK, June 2004.

[5] F. Vitali, F. M. Zerbi, G. Chincarini, et al., "The REM-IR camera: high quality near infrared imaging with a small robotic telescope," in *Instrument Design and Performance for Optical/Infrared Ground-Based Telescopes*, M. Iye and A. F. M. Moorwood, Eds., vol. 4841 of *Proceedings of SPIE*, pp. 627–638, Waikoloa, Hawaii, USA, August 2002.

[6] G. Tosti, M. Bagaglia, C. Campeggi, et al., "The REM optical slitless spectrograph (ROSS)," in *Ground-Based Instrumentation for Astronomy*, A. F. M. Moorwood and M. Iye, Eds., vol. 5492, pp. 689–700, Glasgow, UK, June 2004.

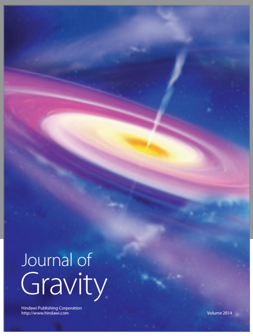
[7] Grigory Beskin, Sergey Bondar, Sergey Karpov, et al., "From TORTORA to MegaTORTORA—Results and Prospects of Search for Fast Optical Transients," *Advances in Astronomy*, vol. 2010.

[8] S. Karpov, et al., "Wide and fast monitoring the sky in subsec domain," *Advances in Astronomy*. In press.

[9] M. Sobczak, "YAMI: a simple orb library," *C/C++ Users Journal*, vol. 21, 2003.

[10] D. Cremonesi, S. Mereghetti, and J. Borkowski, "The INTEGRAL burst alert system," in *Gamma-Ray Bursts in the Afterglow Era*, F. Frontera, E. Costa, and J. Hjorth, Eds., p. 434, Springer, New York, NY, USA, 2001.

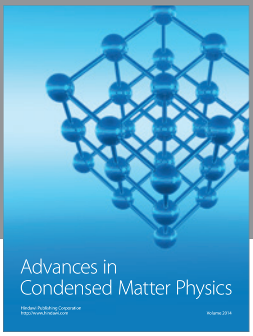
[11] E. Molinari, et al., "REM: automatic for the people," *Advances in Astronomy*. In press.

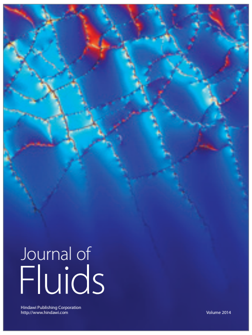

Journal of
Gravity

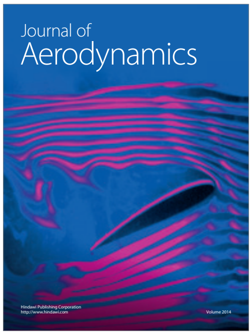
Journal of
Photonics

The Scientific
World Journal

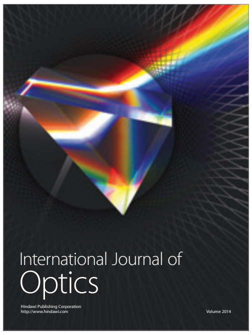
Journal of
Soft Matter

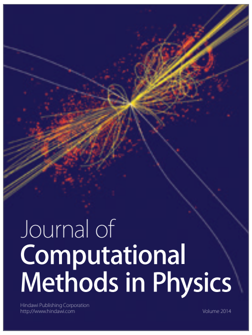

Advances in
Condensed Matter Physics

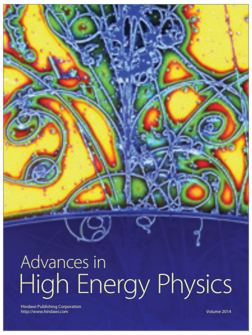

Journal of
Fluids


Journal of
Aerodynamics

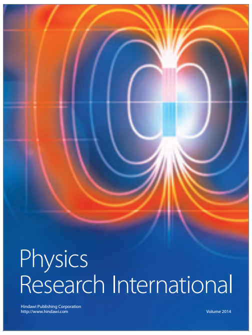
International Journal of
Statistical Mechanics

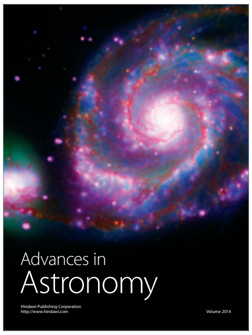

International Journal of
Optics


Journal of
Computational
Methods in Physics


Advances in
High Energy Physics

Journal of
Thermodynamics


Physics
Research International

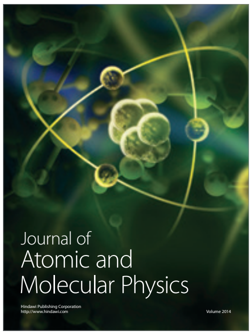

Advances in
Astronomy

Journal of
Solid State Physics

Journal of
Astrophysics

International Journal of
Superconductivity

Journal of
Biophysics


Journal of
Atomic and
Molecular Physics