

# Mantis: Optical Network Planning and Operation Tool

Aristotelis Kretsis, Panagiotis Kokkinos\*, Kostas Christodoulopoulos, Emmanouel (Manos) Varvarigos  
Computer Technology Institute "Diofantus" and University of Patras, Greece

\* Department of Electrical and Computer Engineering National Technical University of Athens, Greece  
Tel: (+30) 2610 960470, e-mail: manos@ceid.upatras.gr

## ABSTRACT

We present a network planning and operation tool, called Mantis, for designing the next generation optical networks, supporting both flex-grid and mixed line rate networks. Through Mantis, the user is able to define the network topology the traffic matrix, the CAPEX/OPEX parameters, setup basic configuration parameters, and use a library of algorithms to plan, operate, or evaluate an optical network of interest. Mantis is designed to be deployed either as a desktop application or as a cloud service. For its execution Mantis can utilize the services of the Amazon Elastic Compute Cloud (Amazon EC2), however its modular architecture allows other cloud services to be adopted in the future with minimum effort. Using the cloud services Mantis can scale based on the user demands, executing fast and efficiently the scenarios requested. The included planning and operation algorithms range from routing, to equipment placement and to wavelength and spectrum allocation and others.

**Keywords:** optical networks, algorithms, tool, planning, operation, cloud services, elastic optical networks

## 1. INTRODUCTION

The continuous growth of consumers' IP traffic [1], fed by the generalization of broadband access (through DSL and FTTH) and the emerging rich-content high-rate and bursty applications, such as video-on-demand, HDTV, Cloud Computing lead to our dependence on optical transport networks. For the future, it is expected that the traffic will not only increase in volume (traffic increase of 34% on average per year [1]) but will also exhibit high burstiness, resulting in large variations over time and direction [2][3]. The most promising technology to meet the requirements of the next generation transport networks appear to be elastic/flex-grid optical networks that will span across both the core and the metro/access segment all the way to the end user, enabling a fine-granular, cost- and power- efficient transport network able to carry a wide range of signal bandwidths that will be varying in real time in direction and magnitude. Network equipment vendors and operators are already looking into this technology. A key enabler for the introduction of flex-grid technology will be the creation of network planning and operation tools that will help plan and manage these networks.

Mantis operates in one of two modes, the network planning and the network operation mode. Network planning, which typically occurs before a network is deployed, is focused on how to better accommodate the future network traffic, assuming that any additional equipment (fiber, regenerator) required can be purchased and deployed. In network operation phase, the demands are generally processed upon their arrival, one at a time, and it is assumed that the traffic must be accommodated using whatever equipment is already deployed in the network. Therefore, the operation process must take into account any constraints posed by the current state of the deployed equipment, which, for instance, may force a demand to be routed over a sub-optimal path.

There is a large number of algorithms that have been proposed in the literature for the planning and operation of fixed-grid or flex-grid optical networks, including algorithms for the Routing and Wavelength Assignment (RWA) or the Routing and Spectrum Allocation (RSA) problem, that take or not take into account Physical Layer Impairments ("Impairment Aware" or "Impairment Unaware"), in the presence or not of regenerators (translucent or transparent networks), with or without Modulation Level adaptability. It is often difficult to compare these algorithms against each other, under a common set of assumptions and parameters, reducing the importance and relevance of the related research. Mantis can be used as a researchers' tool for developing and evaluating, under common conditions, new algorithms for the optical network's domain. Mantis can also be used by equipment vendors when evaluating their devices and by network operators when designing or extending their network, as a mean to increase clients' satisfaction and decrease CAPEX (Capital expenditure) and OPEX (Operational expenditure) related costs. Also, Mantis is the first of its kind academic or commercial tool for flex-grid and mixed line rate optical networks.

## 2. MANTIS ARCHITECTURE

Mantis is a network planning and operation tool for designing the next generation optical networks. For its design, we have adopted an architecture that provides fast execution, efficient computation resources' usage and basic fault tolerance, and enables the deployment of the tool both as a cloud service and as a desktop application.

Mantis components are organized in three layers: *the access layer*, *the application layer* and *the execution layer*. Furthermore, there are two common interfaces whose primary purpose is to provide loose coupling between the application layer and the other two layers. By using these interfaces we can use the same access and

execution layers for both versions of the tool while we can extend their functionality without breaking the implementation of the other components. Figure 1 shows Mantis architecture and its main components.

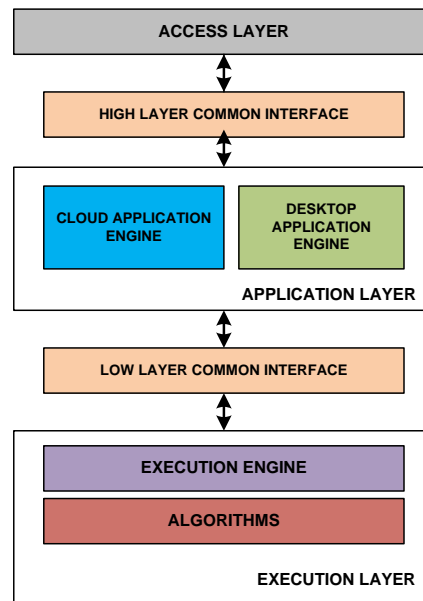


Figure 1. Mantis layers and its main components

The access layer handles the interaction with the users through a web-based user interface and its exposed RESTful API. The execution layer consists of the execution engine and the library of available network planning and operation algorithms. Execution engine receives requests, for starting or terminating algorithm executions, through the common interface from the application layer and is responsible for performing all the required actions, including the preparation of the execution environment, the monitoring of the execution progress and the handling of the final results or possible failures. In the current version, the algorithms are written either in Cython [8] or in C++ language and are accessed from the execution engine through a custom plug-in mechanism. This mechanism enables new algorithms to be added in the tool without any modification of the application layer and the execution engine.

The application layer implements the application logic and orchestrates the execution of user requests. It is the only layer that differs between desktop application and cloud service deployment as there are different requirements and operations that should be performed. In the first case, there is a server that contains the desktop application engine and the execution layer implementations. The desktop application's engine receives requests from the access layer and stores them in a local queue and a disk file that provides a simple fault tolerance mechanism, eliminating the possibility of requests getting lost or not served due to server problems. Also desktop application's engine limits the number of concurrent executions based on the capabilities of the hosting machine in order to avoid resources saturation since the algorithms are executed only in the machine where the server is deployed.

When MANTIS is deployed as cloud service, the application layer implements the cloud application's engine that handles the interaction with the cloud infrastructure. In that deployment, there is one execution engine at every computing node in the available cloud resources. Cloud engine has been designed to be modular in order to support multiple cloud service providers with minimum effort and changes. In the current version Mantis supports Amazon Web Services [6] and ~okeanos [7], GRNET's cloud service for the Greek Academic Community.

Cloud engine consists of the following components: *request and response queues*, *information provider* and *dispatcher*. Cloud engine receives requests from the access layer and stores them in the request queue and a disk file. The dispatcher reads the request queue and checks if the user's request relate to a new execution or the termination of an old one, still in progress. In both cases, the dispatcher uses the available information from the information provider in order to handle the requested operation. The information provider keeps useful details about the available cloud resources, their capabilities, their current load and the tasks that are assigned to each one for execution. Using this information the dispatcher can decide the execution node each request should be forwarded to, while it can automatically adjust the available cloud resources so that they are better aligned with the total demand. Finally, the response queue is used by the execution nodes in order to inform the cloud engine about the usage of their resources and the status of the executed jobs.

### 3. TOOL DESCRIPTION

A primary purpose of Mantis is to create a common benchmarking environment where users compare their algorithms under the same conditions, making in this way their results more usable and creating a connection between them. Users can define various parameters (e.g., network topology, traffic demands, equipment, devices monetary and energy cost) and select among a large set of algorithms for routing and wavelength assignment, routing and spectrum allocation for spectrum-flexible networks, for impairment awareness, for equipment placement, such as regenerators, transponders. Algorithms evaluate future network plans and demands and report a detailed solution including the required bandwidth to serve the demands, the number and configurations for transponders and regenerators, total monetary cost and total required energy, connections that could not be established either due to physical layer impairments or due to bandwidth unavailability.

#### 3.1 Algorithms

The current Mantis version includes network planning and operation algorithms for fixed-grid, flex-grid and mixed line rate optical networks that can be used for both transparent (without regenerators) and translucent (with regenerators) networks. The OFDM IA-RSA (IA stands for Impairment Aware) algorithm [4] considers the planning problem of a flex-grid optical network under physical layer impairments. It serves the demands for their requested rates by choosing the route, breaking the transmission in multiple connections, placing regenerators if needed, and allocating spectrum to them. The IA-RWA-MLR and IA-RWA-SLR algorithms [4] consider respectively the planning problem of mixed line rate and single line rate fixed-grid optical networks under physical layer impairments. The physical layer effects are incorporated in the definition of the feasible transmission configurations of the transponders that can be used in each case. Online versions of these algorithms are also available that take as input the output of the offline case and serve the new demands defined by the users. Furthermore, the EA-RWA algorithm [5] aims at minimizing the energy consumed by the optical layer components when planning translucent WDM optical networks.

#### 3.2 User Interface

MANTIS comes with a clean and simple web-based user interface through which the users can have access to all tool functionalities. The main Mantis' entities are the configurations, the instances and the projections. A configuration consists of a set of parameters that define the execution of a specific algorithm, while an instance is an executed configuration. The projections are parameterized versions of a configuration that result in a set of instances to be generated at the execution nodes. These instances differ only in the value of one initial parameter (e.g., number of wavelengths). In the user interface there is a separation of the different steps: network topology and traffic demands creation, algorithms selection and configuration, execution and results presentation.

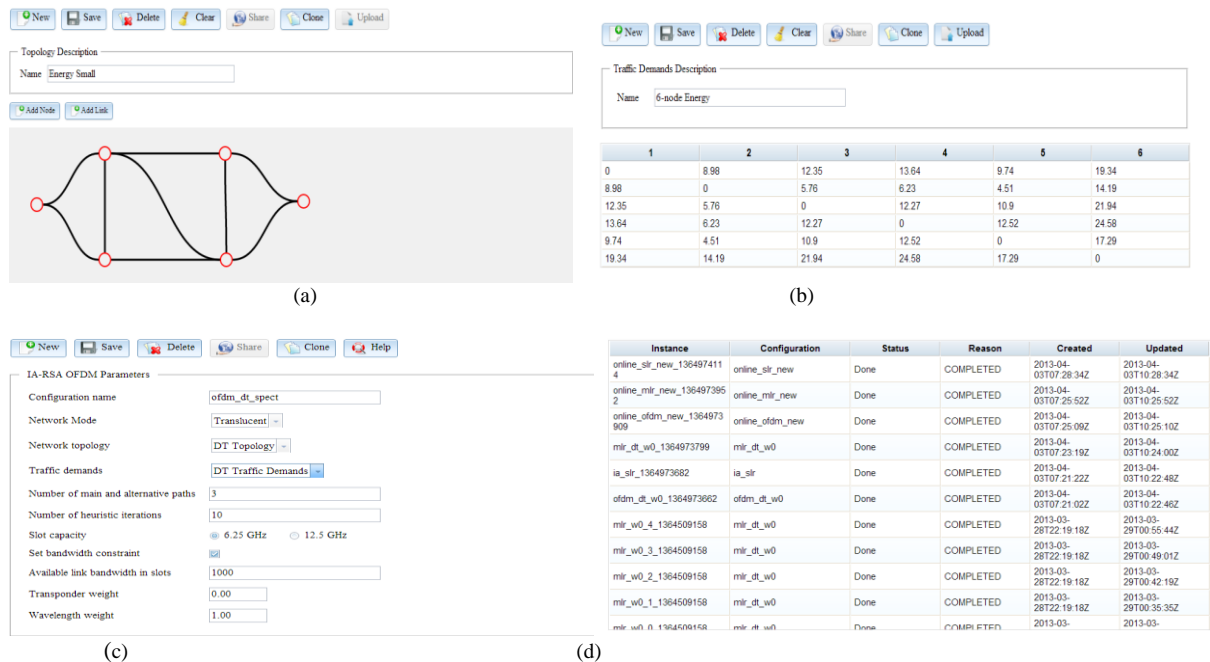


Figure 2. (a) create network topology (b) define traffic requests (c) create configuration (d) information for all instances

Mantis user interface enables users to easily and graphically design network topologies by defining network nodes and links between them, save, edit, and use them at any time. Similarly, users can define their own traffic

demands, which are expressed as square matrices, either graphically or import them from comma-separated values (CSV) files. Figures 2a and 2b show the available interface for network topologies and traffic demands respectively. New configurations can be created for each algorithm by selecting a network topology, traffic demands and specifying all the other required parameters (Fig 2c). Mantis automatically checks that all provided parameters are indeed a valid combination and informs user for possible mistakes before store a configuration. For each configuration, users can request either its execution or the creation of a new projection by selecting the parameter in which the resulting instances will differ and by defining a range of values for that parameter.

Users can always check the status of the running instances and have access to useful details for all the instances including the instance name, configuration name, execution status, creation date, execution date (Fig 2d). For every successfully executed instance the user can view analytic results and export them for further analysis. Furthermore charts can be created by combining the results of completed instances. Additionally, network topologies, traffic demands, configurations and charts can be shared with other users who have only read access to these items. However, they can create their own copies with edit permissions by them. In [9] there are available more details regarding Mantis and how someone can use it.

#### 4. FUTURE WORK

We have planned a number of extensions in all MANTIS layers for the next months. These include adding new functionality both in the web-based user interface and in the exposed RESTful API, and developing a python API and a command-line interface (CLI) for users to interact with the tool. New algorithms will also be integrated in the library for spectrum allocation and defragmentation, restoration, and energy efficiency. Furthermore, in order to speed-up the execution of various computationally intensive optical network design and operation algorithms we are developing appropriate techniques that will enable the decomposition of algorithms operation in smaller subtasks so as to but executed in parallel in the cloud. Finally, we plan to provide appropriate interfaces in order to integrate Mantis's online algorithms with network management tools for optical networks providing path computation element (PCE) functionalities.

#### 5. CONCLUSIONS

Mantis is a network planning and operation tool for designing the next generation optical networks. Mantis can be deployed either as desktop application or as cloud service, with the latter being the primary implementation of interest so as to be available over the internet to the users. Users (researchers, operators, vendors) can use Mantis to perform studies as desired but also implement their own algorithms and compare them against the ones already incorporated in the tool. Through the web-based interface users can select different topologies, traffic demands, CAPEX/OPEX and energy models and algorithms. Using the cloud services Mantis can scale based on the user demands, executing fast and efficiently the scenarios requested. In this way users will be able to use the services of an advanced network planning and operation tool without the need of installing anything.

#### ACKNOWLEDGEMENTS

This work was partially funded by the European Community's Seventh Framework Programme FP7/2007-2013 through the Integrated Project (IP) IDEALIST under grant agreement n° 317999.

#### REFERENCES

- [1] Cisco Visual Networking Index -Forecast and Methodology, 2010–2015.
- [2] T. Stern, G. Ellinas, K. Bala: Multiwavelength Optical Networks: Architectures, Design and Control, *Cambridge University Press*, 2nd Edition, 2008.
- [3] O. Gerstel, M. Jinno, A. Lord, S. J. Ben Yoo: Elastic Optical Networking: A New Dawn for the Optical Layer?, *IEEE Communications Magazine*, Feb, 2012
- [4] K. Christodoulopoulos, P. Soumplis, E. Varvarigos: Trading off Transponders for Spectrum in Flexgrid Networks, *Proc. Optical Fiber Communication Conference 2013*, Anaheim, USA, 2013, paper OTu2A.3.
- [5] K. Manousakis, A. Angeletou, E. Varvarigos: Energy Efficient RWA Strategies for WDM Optical Networks, *Journal of Optical Communications and Networking* Vol. 5, Issue. 4, pp. 338–348, 2013
- [6] Amazon Web Services: <http://aws.amazon.com/>
- [7] OKEANOS – GRNET cloud infrastructure: <https://oceanos.grnet.gr>
- [8] Cython: <http://www.cython.org/>
- [9] <http://www.mantis-tool.net/>