ACTIVE LEARNING IN SATELLITE MISSION DESIGN

Siti Fariza Mohd Dahlan
School of Aerospace Engineering
Universiti Sains Malaysia
Penang, Malaysia
fariza@eng.usm.my

Illyia Mohd Yusof
School of Aerospace Engineering
Universiti Sains Malaysia
Penang, Malaysia
illyia@eng.usm.my

ABSTRACT

Designing a satellite mission is a not an easy task as it requires thorough consideration, complex calculations and repeated analysis in determining the best design parameters. In designing a satellite mission, we need to have a structured or systematic approach in tackling the issue. For beginners in satellite mission design, the process can be an exhaustive and a complex activity. Without careful attention, we may miss some important aspects of the design procedures. Such experience would also contribute to the loss of interest among students. Therefore, a tool that can facilitate the design activity, and minimize the complexity of the activity, is needed. Such tool would also promote active learning amongst students in learning not only the theories in designing a satellite mission, but also the analysis part of the design process.

The purpose of this paper is to present a research conducted in developing satellite mission design tool that is aimed to promote better understanding in satellite mission design and analysis techniques to students who specialized in Astronautics. The software, with different design module, is targeted to facilitate analytical aspect of the design process in the form of computer aided interaction (CAI) and computer based learning (CBL). The outcome of this research is a graphical satellite mission design and analysis software package for educational purposes either for teaching aid or for self-learning.

INTRODUCTION

Among the important topics learnt by students in Astronautic program are the theories of space mechanics, space vehicles, space applications and space environment. These topics sum up to be the rudimentary elements needed in designing a satellite mission. Mission design involves planning for the right flight characteristics in order to satisfy the mission requirements. The determination process of the right flight characteristic involves not only to determine the best orbital location, but also to consider the space environment condition at that location and the influence of perturbation overtime[5]. Spacecraft sizing is another crucial element in a mission design as it involves dependencies of several subsystems together on-board to successfully perform the mission during its lifetime. These are among the prerequisite steps to be taken before moving to higher hierarchy of satellite design phases.

The time taken for a thorough mission planning and analysis can be greatly reduced with the use of good computer software. Nowadays, there exist many freeware, shareware and commercial satellite mission design and analysis software in the market. The best software could cost dearly to an organization. Developing equivalent software could be painstaking and time consuming. Nevertheless, the process is seen worthwhile as the development process itself is a learning process. An educator whom assimilates knowledge could better transmit his/her knowledge through software development, while students could benefit from the hands-on experience gained during the process.

Effective learning is an active rather than a passive process. This would require students to be placed in settings in which they are challenged to think critically and to articulate their thoughts and experiences through continual engagement with their peers, their lecturers, and their surrounding community. In achieving this, the use of proper methodologies and technologies should be provided to promote self-learning while having greater student and lecturer interaction. The development of Satellite Mission Design software is hoped to be able to deliver an active learning endeavor required.

Experiences gained in early 2001 from the School of Aerospace Engineering, Universiti Sains Malaysia in taught courses ESA111 – Introduction to Aerospace Engineering (Astronautics section), ESA264 – Flight Mechanics and design electives courses indicate that students encounter difficulties in understanding rudimentary theories associated to space mechanics[5]. Lecturers often have to spend more time and energy to prepare visual aided pictorial and models depicting the solar system and satellite orbits to get the message through. Although limited in every aspect (especially in describing derived terms and of course limitation of the material used) this method has transpired certain extend of interest that leads to self-learning of space elements. Nevertheless, students cannot fully benefit from these techniques due to the complexity of the subject as lecturers tend to simplify things up and examples are limited on special cases only.

A more comprehensive and robust technique has to be employed based on one-to-one or step-by-step approach in order to promote self-learning. The physical models and motions described can be depicted in the computer without limitation. Any motions can be simulated in the computer as well as help items for beginners.

The primary objective of this project is then to develop satellite mission design software as a tool to promote better understanding and self-learning of space mechanics and to facilitate analysis aspects of the design process for undergraduate students. Besides that, the project also aims to develop skilled programming capabilities among undergraduate students, improve and strengthen space mechanics related knowledge, provide an avenue for enthusiastic students to further improve the software, and nurture primary and secondary students in the understanding of space mechanics and design process.

REQUIREMENTS BASELINE

The basic requirements for this project are to develop a preliminary design tool of an earth observation satellite mission. This includes step-by-step procedures in designing a satellite for an earth observation mission and the tool should be in Graphical-User-Interface (GUI) format to facilitate usage.

Though many other similar preliminary mission analysis tool available in the market offer broader spectrum of functions, for example J-Track, Home Planet, Planetarium V7, Orbwin and Satellite Tool Kit, this project scope is limited to two-body mechanics due to limited time and resources without compromising the aspects required by the objectives of the project. The end product of the project will offer functions adequate to the needs of the School of Aerospace Engineering whilst being able to provide rooms for improvements and other customizations.

Functional Requirements

Referring to the problems faced by the group of students, the actual need is to develop a more systematic model to effectively promote better understanding in space mechanics. Students would learn better through graphical representation and best through step-by-step approach. The design of the software should be in a way such that answers can be immediately retrieved after the execution of a set of data. Besides preparing some help files on each step to be carried out, students should also be furnished with a set of default data for comparison with user input data. The final result should be in the form of computer aided interaction (CAI) and computer based learning (CBL).

System Requirements

This project is carried out under the MATLAB environment. Adopting Graphical User Interface based design concept, the project relies heavily on the Graphical User Interface Development Environment and a few modules shall require the use of MATLAB Mapping Toolbox besides other mathematical utilities toolbox. The end product shall consist of only an executable file and some help itineraries. With the current feature of MATLAB 7 of Release 14, the software shall be easily converted to a stand alone application later on. This software should be able to run on the latest computer technology using Windows operating system.

SOFTWARE REQUIREMENTS ANALYSIS

An analysis of the software requirements was conducted based on the functional requirements presented earlier.

Software Functionalities

The first phase of the project focused on the fundamentals of space mission design. As the software is targeted to be a teaching tool and to aid students in self-learning of space mission design, the software should be in such a way that the theories learned in class can easily be visualized and understood. The software should also increase time efficiency in understanding complex theories, as well as mission design activities. Apart from that, the software shall be user-friendly, such that it should be simple enough to be used by entry level students. In addition, tool tips and help files will be made available to facilitate software usage.

Software Architecture

To be able to demonstrate a step-by-step procedure in designing a satellite mission, the software is designed as a module based application, to enable more modules to be added for future needs. As mentioned by Larson and Wertz[1], modules planned for preliminary requirements are based on spacecraft mission elements which are:

a) Subject or mission to accomplish
b) Orbital elements
c) Space segment: payload and subsystem
d) Launch Segment
e) Ground Segment
f) Communication Structure
g) Mission Operation

The modules are interdependent between one and another in order to acquire a complete analysis on the preliminary requirements. However, in the first phase of the project, all mission elements listed above are analyzed and programmed as a general overview. Detail analysis of each module, such as the Orbital Elements, Space Segment and Communication Structure, are done in the second phase of the project. As one of the strategies is to get students involve in the project and as part of an active learning, a few students are involved in the development of the modules.

Technical Specification

Some software constraints were identified and shall be rectified in future phase of the project. Some orbital parameters, such as the type of coordinate system, graphical images or some simulation data were made constant due to these constraints. Greater flexibility is hoped to be achieved in future phases. Since the software runs on modular basis, default values are made available for general calculations.

Table I. Default Values for General Orbital Calculations

Parameter	Type	Value
Propagator	Constant	Two Body
Coordinate Type	Constant	Classical
Coordinate System	Constant	J2000
Animation Start Time	Variable	
Animation Stop Time	Variable	
Animation Step Size	Variable	
Orbit Epoch	Variable	

Default values and range of relevant values are also made available for most of the required parameters to further facilitate the usage of the software. This is especially useful for beginners in space mission design, as it gives them an idea of the relevant values to be entered.

DESIGN ENGINEERING PROCESS

The design process is dependent on the software requirements as discussed above. The design process life cycle include designing of software items, coding, testing and integration of software units[3].

However, the software design should be quite similar to the real mission design process itself. Before a satellite is placed in its orbit and performs its mission, the satellite mission needs to be thoroughly designed. This design process involves determining the mission to be performed, size of satellite, the best payload and subsystem equipments, spacecraft sizing, orbital elements and so on. This process actually goes through a thorough satellite project life cycle. There are 5 phases in this cycle[1]. Namely:

- Pre-Phase A: Advanced Studies
- Phase A: Preliminary Analysis
- Phase B: Definition
- Phase C: Design
- Phase D: Development
- Phase E: Operations

Nevertheless, this project shall only concern on Phase A. Phase A is normally considered as the most important phase in a satellite project life cycle. Phase A concerns on the determination of the feasibility of a suggested mission, preparation of mission need statement, requirements, constraints and initial considerations[1]. Briefly, Phase A involves Mission Analysis of a satellite project.

Mission Analysis is the most important phase in the development of a satellite as it ties together two important elements, which are, the satellite which will be performing the mission and the users, who will use the satellite outputs for their applications. In Mission Analysis, it is important to relate the spacecraft orbit and attitude to the ground and to ensure that operational aspects can be achieved, such as spacecraft ground communications and spacecraft tracking. Consequently, a careful analysis of the mission objectives, payload operations and orbital aspects is required which places design requirements upon the spacecraft subsystem elements. Mission design then becomes an iterative process and can be time consuming in order to get the best parameters.

In the first phase of this project, all modules are developed under the MATLAB common environment, without using GUIDE. Mission elements are programmed as general overview. A thorough top-level architectural design is written in the form of seven inter-dependent main modules as mentioned earlier. Users are asked to input a set of variables in the first module which will then output another set of variables to the second module and so on. Though the sequence of operations is unapparent to users, the order of each module should be noted in order to obtain a detail analysis. In return, the user shall be able to experience the step-by-step process in designing certain preliminary mission. However, each module can still be executed independently, in which case default values will be used where necessary. The program is then executed from MATLAB command window as shown in the Figure 1 below.

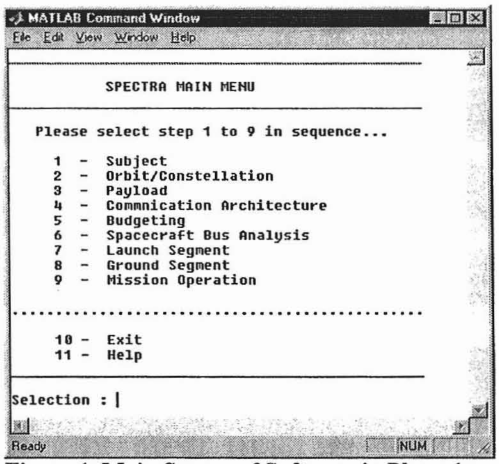


Figure 1. Main Screen of Software in Phase 1

Module 1: Subject

In module 1, the software requires user to specify the subject to be detected or mission to be performed by the satellite. Users need to define the criteria for the subject to be viewed such as the coverage area and spatial resolution. Fig. 2 shows an excerpt from the Subject report generated at the end of Module 1.

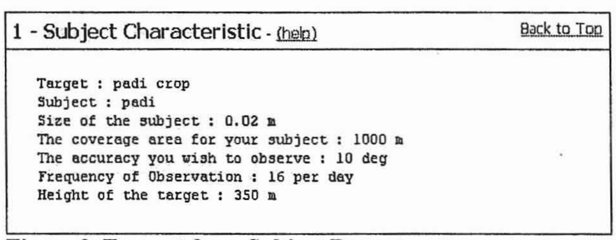


Figure 2. Excerpt from Subject Report

Module 2: Orbital Design

The second module allows user to define important initial orbital parameters of a satellite orbit. The expected output of the process is a 2-D graphical simulation of the orbit based on the information gathered from user. User has an option to choose one of the following predefined orbits:

a) Critically inclined
b) Critically inclined sun synchronous
c) Geostationary
d) Molniya
e) Repeating ground trace

f) Repeating sun synchronous
g) Sun synchronous

or, a custom defined orbit.

Only two input parameters are required to create a custom orbit. As such, the first parameter selected determines the input parameter needed in the second.

Table II. Available Pairs of Parameters

First Parameter	Second Parameter
Semimajor axis	Eccentricity
Apogee Radius	Perigee Radius
Apogee Altitude	Perigee Altitude
Period	Eccentricity
Mean Motion	Eccentricity

Orbit plane changes can be viewed by choosing the kind of transfer orbit required, for example, Hohmann Transfer or Bielliptical Transfer.

In order to view the orientation of the orbit plane in space, two Keplerian elements, represented by the following parameters, are required:

- Right ascension of the ascending node (RAAN): The angle in the Earth's equatorial plane measured eastward from the vernal equinox to the ascending node of the orbit[2]

Or,

- Longitude of the ascending node: The angle between the vernal equinox vector and the ascending node, measured in the reference plane in a counterclockwise direction as viewed from the northern hemisphere[2].
- Argument of perigee: The angle, in the plane of the satellite's orbit, between the ascending node and the perigee of the orbit, measured in the direction of the satellite's motion[2].

A report on the result of orbital calculations can be viewed at the end of the simulation, as shown in Figure 3.

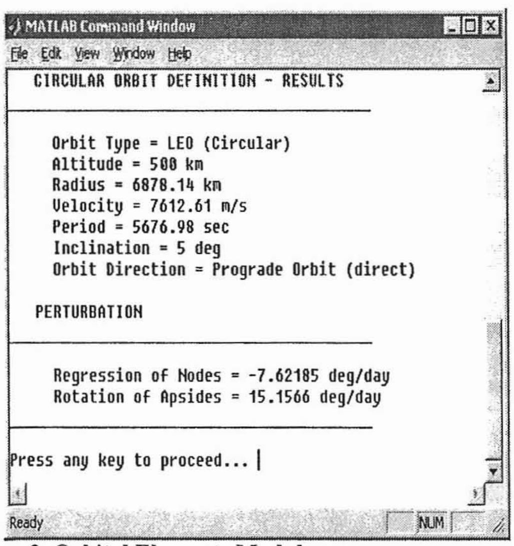


Figure 3. Orbital Elements Module

A range of parameters will be used to determine the satellite location above the earth. The orbital parameters used for such case are depicted in Figure 4 and Figure 5.

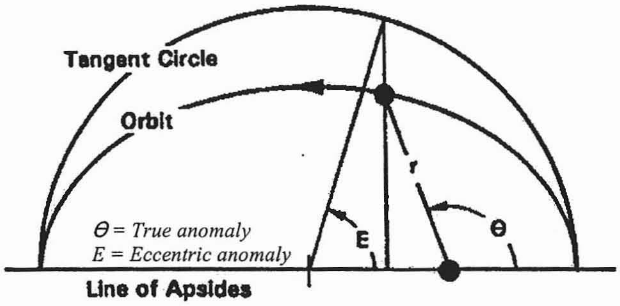


Figure 4. Parameters Determining Satellite Position[2]

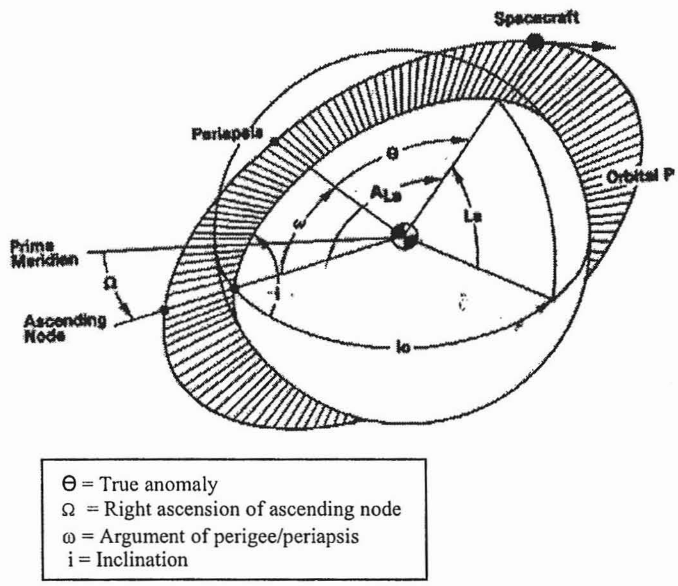


Figure 5. Parameters Determining Orbit Orientation[2]

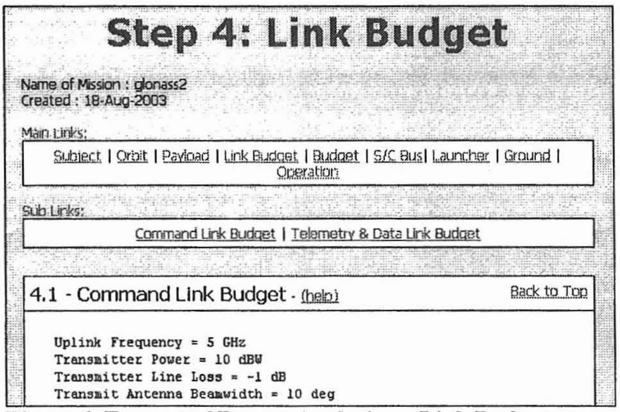
Other parameters used in determining satellite position are Mean anomaly, Argument of Time past ascending node, Time past perigee.

Module 3: Space Segment

Space segment module lets user to select suitable payload for the spacecraft mission and then to estimate payload's size. The software will calculate the aperture size and aperture ratio of the payload based on the wavelength and resolution required.

Module 4: Communication Architecture

In this module, the software will perform the link budget analysis for the specified mission, which is the command link budget and telemetry and data link budget.



Step 4: Link Budget

Name of Mission : glonass2
Created : 18-Aug-2003

Main Links:

Subject | Orbit | Payload | Link Budget | Budget | S/C Bus | Launcher | Ground | Operation

Sub Links:

Command Link Budget | Telemetry & Data Link Budget

4.1 - Command Link Budget - (help) Back to Top

Uplink Frequency = 5 GHz
Transmitter Power = 10 dBW
Transmitter Line Loss = -1 dB
Transmit Antenna Beamwidth = 10 deg

Figure 6. Excerpt of Report Analysis on Link Budget

Module 5: Mass Budget

Two analyses will be done in this module, which are to calculate the power budget and the mass budget for the whole spacecraft. For this purpose, the payload power estimation calculated in the earlier module will be used. However, basic assumptions need to be made to calculate the approximations.

Module 6: Spacecraft Bus

In this module, users will be guided through the subsystems analysis: attitude determination and control, communication, command and data handling, power, thermal, structure, guidance and propulsion. In each subsystem analysis, the elements to be used or specified parameters are to be determined and calculated.

Final Result of Phase 1

All results from each module are stored in HTML documents during the process. Each HTML file contains the results from each module and all these files are stored in a folder created during runtime. As such, users can compare the results obtained from one mission to another for further analysis. However, the first phase of the software only focuses on the programming engine not on the user friendly aspect of the design.

SECOND PHASE OF SOFTWARE DEVELOPMENT

The next step of the development process is then to migrate the software to be developed under Graphical User Interface Development Environment (GUIDE) in MATLAB. More detail analysis is done for each module and a few new stand-alone modules are designed based on the existing software codes. Other than that, more help files are included to provide a more user-friendly environment.

As shown in Figure 7, the interface to the software is in GUI based, compared to the earlier version as in Figure 1. In Figure 8, we can see that the subject Module is discussed in greater length in its help file and more default values are given besides the range of relevant parameters to be selected.

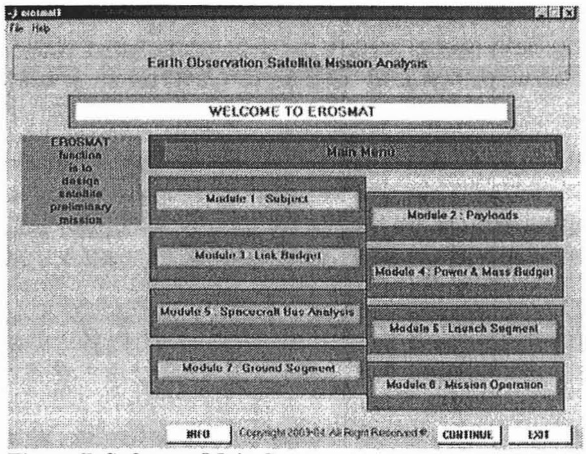


Figure 7. Software Main Screen

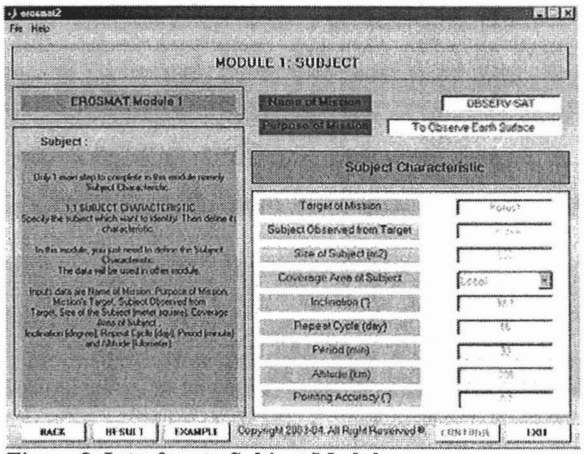


Figure 8. Interface to Subject Module

One of the new stand-alone modules developed concerned on the Orbital Elements Module (refer Figure 9). This module is targeted to aid in learning the rudimentary theories in Orbital Mechanics. The module is further divided into five sub-modules which enables user to:

- Locate the location of satellite with parameters from user input
- Locate commercial satellites with pre-set parameters
- Observe satellite crossing earth shadow
- Calculate orbital elements from ground station parameters
- Analyze orbit perturbations

Figure 9. Stand-alone Orbital Elements Module

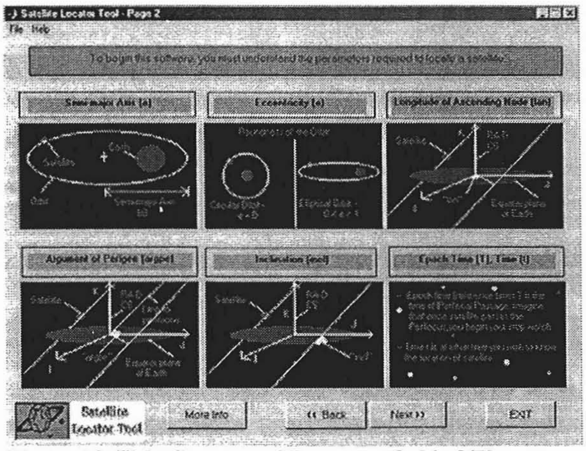


Figure 10. Help Screen on Theory on Orbital Elements

In simulating the satellite ground track, the software adopts Universal time, Julian Date, and Sidereal time[4]. User needs to enter the simulation time but are advised not to exceed more than 1 day simulation. Figure 11 shows the interface for user input parameters to simulate a satellite ground track.

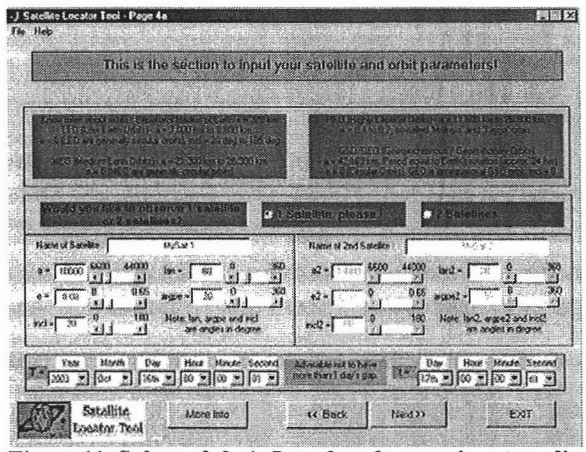


Figure 11. Sub-module 1: Interface for user input to display a satellite ground track.

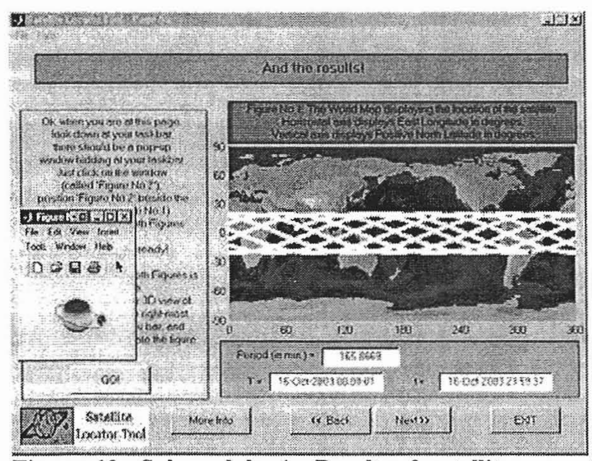


Figure 12. Sub-module 1: Result of satellite ground track displayed based on input parameters and simulation time selected.

This module heavily used the Mapping Toolbox available in MATLAB in mapping the satellite ground track and in observing the satellite crossing the earth shadow. A small 3-D view of the earth in Fig. 12 shows the movement of the satellite around the earth while the ground track is drawn on the earth map. As the software is targeted for beginners, some common assumptions were incorporated in this module as follows:

- □ the satellite is only attracted by Earth,
- □ the mass of the satellite is ignored,
- □ earth is spherical and concentration of mass is at the center of the earth,
- □ and no other forces are involved except gravitational force and centrifugal force.[4]

VALIDATION AND VERIFICATION PROCESS

The software validation and verification is trivial in confirming that all requirements have been met and all design constraints discussed earlier are respected. In this process, each software module is checked for validity of output, and consistency. Some calculations made in the first phase were verified by comparing the results with other existing software and some were verified manually. Results showed some variations on the calculations due to the different default values used and the different models adopted, for example in terms of the coordinate system and the perturbation model.

For the Orbital Elements stand-alone module, verification was done using Satellite Tool Kit. In the verification process, a same set of orbital elements is used for both before the ground tracks were simulated. The results shown to be close where same ground track patterns were obtained and same maximum and minimum latitude were achieved. However, there are some slight variations found on the coordinates of both starting positions of the ground track: 9.48% of longitude difference, and 6.5% of latitude difference[4]. These are based on a couple of assumptions which were incorporated into the module and the different kinds of model being utilized. However, the software shall be optimized from time to time to minimize the errors and the assumptions made.

SOFTWARE APPLICATION

The modules shall be further enhanced and optimized from time to time to include more functionality and to be more user-friendly. Since the software is design for beginners in satellite mission design, a lot of default parameters will be incorporated where suitable. Enhancements need to be done in order to minimize the number of assumptions made to get better results.

All of the advanced modules developed separately will be reintegrated in the main software in the final phase in order to achieve the objective of this project. One of the advantages in this method is that these individual advance modules can be used separately in courses such as Orbital Mechanics and Satellite Communication. The software as a whole shall be used in Spacecraft Design courses.

CONCLUSIONS

The development of this software is carried out with respect to the needs of Astronautic Engineering students in learning the process of earth observation satellite mission design. Each software module developed has the potential to be made stand-alone to provide more functionality and to be used by more advanced users. Apart from being a teaching tool for lecturers in conducting their space mechanics related courses, the software is targeted to aid students in self-learning of the fundamental concepts of orbital mechanics as well as to enhance the programming skills of the students involve during the development process.

While facilitating their learning process, involvement of students in the project has transpired a great length of interest in them in space mission design. The project is aimed to provide an active learning environment to Space Mission Design students which shall in return help them in learning more effectively.

ACKNOWLEDGMENT

This research is part of the USM Short Term Grant "Preliminary Mission Analysis Tool: A Software for Satellite Mission Design", 2001-2003.

REFERENCES

[1] W. J. Larson and J. R. Werzt (1995), *Space Mission Analysis and Design*, 2nd ed., Kluwer Academic, London.
[2] C. D. Brown (1992), *Spacecraft Mission Design*, American Institute Aeronautics and Astronautics, Inc. Washington.
[3] European Cooperation For Space Standardization (1999), *Space Engineering Software*. ESA Publications Division. ESTEC. Netherlands.
[4] P.M. Foong (2004), *Pembangunan Perisian Penentuan Lokasi Satelit*, Final Year Project, Universiti Sains Malaysia.
[5] S.F. M. Dahlan and I. M. Yusof (2003), "Preliminary Mission Analysis Tool (PMAT): Educational Software for Satellite Mission Design", *Proceedings of the Seventh Triennial AEESEAP Conference, Enhancing Engineering Education and Training: A University-Industry-Government Partnership*, pp. 149-154.