Commentary on: Multi-Objective Scheduling for the Cluster II Constellation

Mark D. Johnston and Mark Giuliano

by **Roman Barták**
Charles University in Prague, Faculty of Mathematics and Physics
Malostranské nám. 2/25, 118 00 Praha 1, Czech Republic
bartak@ktiml.mff.cuni.cz

Paper Summary

The paper describes a system for scheduling observations for a cluster of four satellites studying plasma environment of the Earth and its magnetosphere. There are two important aspects in the problem. First, there is no on-board memory for storing the measurements so data need to be streamed directly to Earth via NASA Deep Space Network (DSN) during the observation. This requires co-ordination of the schedule with other schedules submitted to DNS. Second, as many real-life problems, there are several evaluation criteria that need to be taken in account during generating the schedule. The paper focuses on the second aspect and covers only briefly the first aspect.

The paper starts with a description of MUSE (Multi-User Scheduling Environment) architecture that was designed to support a wide range of scheduling problems by including both a generic part and an application-specific part. To support a multi-user environment, the architecture is also split into a server part running the computationally intensive calculations (I guess) and a client part running only GUI. After that, the scheduling problem is informally described with the focus on schedule quality criteria and problem input. Then the paper briefly describes a core part of the scheduling model, namely the variables representing the solution. At the end the authors show the GUI of the system that displays the schedules resulting from multi-criteria optimization using genetic algorithms and allows users to select a set of schedules based on certain criteria.

Solving Approach

The paper does not say a lot about how the problem is being solved. Genetic algorithms (generalized differential evolution) are used to solve the problem because they can keep a population of non-dominated schedules from which the user can select. The key component is representation of the problem using decision variables where each variable describes a combination of observation opportunities. This is an interesting though ad-hoc approach and I would like to see more about how the opportunities are grouped together to form a decision variable. What criteria are used to put the opportunities together? The second step is mapping the discrete domain of the variable to real interval (0,1) due to used solving method. Again, it would be valuable to see the general principle how to map the particular values to intervals in (0,1). Isn't it better to use a different solving approach that works directly with discrete variables? Is there a particular advantage of using GDE? One aspect of the solving approach is also not clear. The generated schedules take in account existing schedules for DSN with the goal to avoid already busy areas. Isn´t it possible that other users of DSN use a similar approach so at the end parts of the schedules are in conflict and cannot be realized due to exceeding the capacity of DSN? How is the observation schedule integrated to a DSN schedule and how are the possible conflicts resolved?

Visualization

The graphical user interface allows users to visually compare generated schedules and to show a particular schedule. I like visual clarity of the interface consisting of three parts: the histograms of individual criteria, the X-Y plot for comparing pairs of criteria, and the Gantt-view of the schedule. The user can highlight a particular interval in a single histogram and the corresponding schedules are highlighted in the other views. I only think that it would be better to use red-blue colors from histograms also in the X-Y plot view which would simplify understanding the selection. It is not fully clear to me whether the user can also select an area in X-Y plot and whether it is possible to combine selections, namely to select a set of schedules in one view and then to restrict this selections in other views until a single final schedule is obtained. The paper does not describe how the user selects the schedule to be executed and I think that a process of subsequent pruning of schedules could be useful.