

Scheduling with Slew/readout Time Considered

July 30, 2018

This document describes how the scheduling works with slew and readout time considered. Scheduling is the process of arranging the order of observations given the time allocated on each tile. Thus, it takes input from time allocation and outputs an list of observations: a list of tiles and the times to watch them. The purpose of the scheduling optimization is to arrange these observations in order to maximize the probability of detecting an EM counterpart.

Slewing is the movement of the telescope between the tiles and readout is the time required to process the image after an exposure. After each exposure, we have to move the telescope to the location of the next tile and process the previous exposure. Thus the time gap between each observation is a function of the slew time and readout time. The slew rate is the speed of the telescope movement, measured by the angle the telescope can move per second. We treat the slew rate and the readout time as constants. The slew rate is constant so that the slew time will be the angular distance between the tiles divided by the slew rate. We also assume that the slewing and readout can take place at the same time. Thus, for a tile at angular distance d from the current location, the total time T between observations would be

$$T = \max(\frac{d}{s}, r), \quad (1)$$

where s is the slew rate and r is the readout time.

Previously, we have three scheduling options: greedy, sear and weighted. They ignore the slew time and readout time. Now we are going to upgrade them into "greedy_slew", "sear_slew" and "weighted_slew" by bringing slew time and readout time into the algorithms. The approach of taking account of slew and readout time is to bring them as new elements into the algorithm and try to output the best possible schedule. Sometimes we introduce structural changes in order to incorporate the new elements; however, we keep the basic concepts of the algorithms intact.

1 Greedy

The greedy algorithm without considering the slew and readout is simple; we just select a tile that is both available and covers the highest probability. However, when we make decision with the time wasted between observations considered, we need to introduce a score S for every tile that we need time T to reach from the current location:

$$S = \frac{pt_a}{1 + T} \quad (2)$$

where p is the probability covered by the tile; t_a is the remaining allocated time for that tile. Thus, the longer it takes to slew to that tile, the worse the score is. We also introduce the remaining allocated time t_a to this equation since we prefer to observe a tile that we can sit on for a longer duration.

The procedures of the algorithm is shown in Figure 1. We start at the first available observation time of the telescope. Then, we move on checking if the tiles were available for observation now. If the tile is available, we calculate the score. If there is any tile available, we choose the one with the highest score and stay on it until it becomes unavailable or its allocated time runs out. After we update the time and go back calculating the scores again. Otherwise, if there is no tile available currently, we need to check if there is any tile available in the future. If there is, we choose the first one becoming available for observation. However, if there is not any available tile in the future, we should end the program and output the schedule.

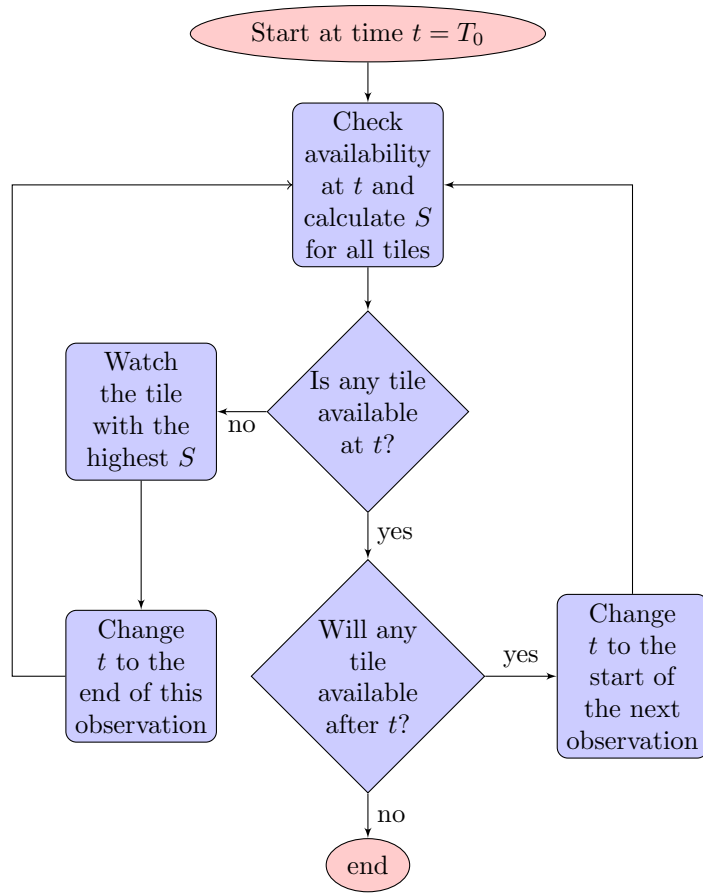


Figure 1: Greedy scheduling considering slew and readout time.

2 SEAR

3 Weighted