Chapter 2
Scheduling Process

In this chapter we put the scheduling process in the context of the various missions that the problem arises in. This chapter is intended for readers that are approaching SRS, or for professionals from other industries, to show where all the constraints of this problem come from. As we mentioned in the previous chapter, SRS is applicable in satellite communications, Earth observation, and sensor scheduling. We will describe the scheduling process applied to satellite communication scenarios, without loss of generality for other cases.

Although the experienced professional may skip this chapter, reading of §2.2 and §2.3 is encouraged as these subsections present the parameters used in this book to classify scheduling problems.

2.1 Scheduling Process

For our scheduling model we consider a set of *ground stations* which move with the surface of the Earth, a set of *mission control centers* which can be assumed to be continuously connected to the ground stations, and *satellites* traveling through different kinds of orbits generating *visibility windows* when line of sight (LOS) to ground stations exist [1]. A possible scenario is displayed in Fig. 2.1.

The satellite *operators* aim to establish communications between their mission control center and their associated satellite, but this can only be done through the *ground station network*. Based on the dynamics of the scenario and on the requirements of the mission, the operators generate an operations plan. Then, from these visibility time windows and the operations plan, the operators generate a set of *requests* characterized by constraints associated to these time windows. The

A.J. Vázquez Álvarez, R.S. Erwin, *An Introduction to Optimal Satellite Range Scheduling*, Springer Optimization and Its Applications 106,
DOI 10.1007/978-3-319-25409-8_2

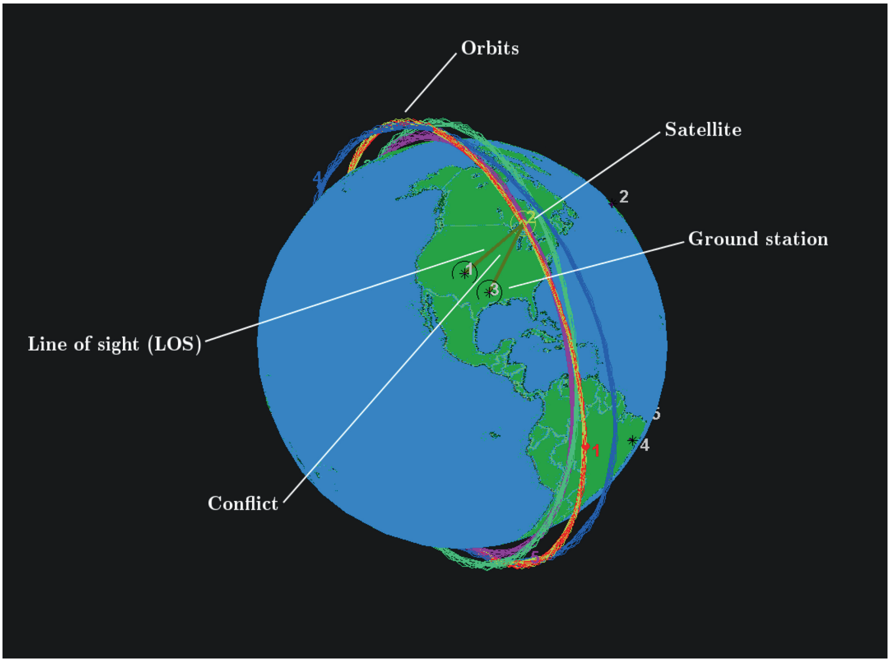
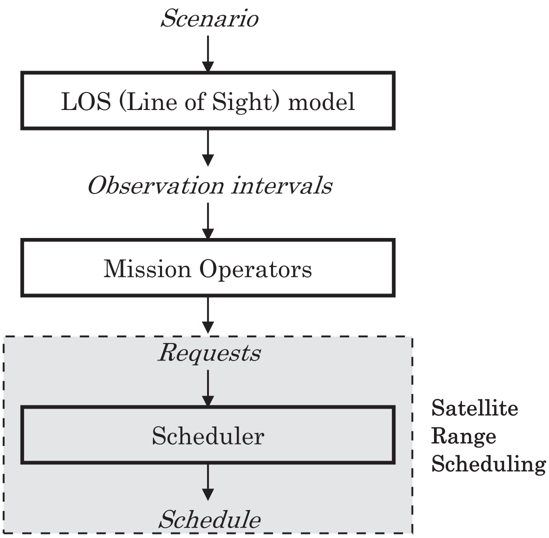


Fig. 2.1 Scheduling scenario

Fig. 2.2 Scheduling process



objective of the scheduler is to generate, from this set of requests, a *schedule*, which is a subset of these requests selected for execution. This process is illustrated in Fig. 2.2.

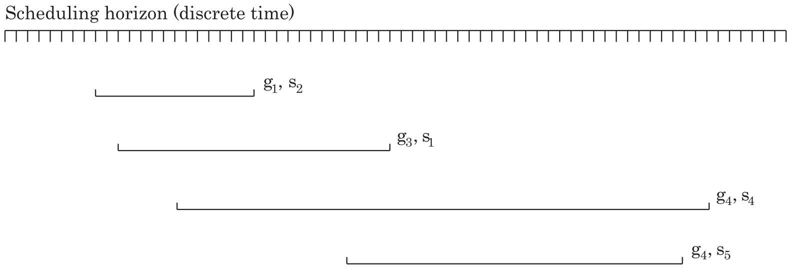


Fig. 2.3 Visibility windows

The duration of the visibility windows will depend on the dynamics of the problem (orbits of the satellites and positions of the ground stations). For a low Earth orbit (LEO) satellite these durations may be of around 10 min at most. Given that this time is relatively small, the whole visibility window is generally requested. For higher orbits however, these durations may be longer than one hour, up to the case of geostationary orbits (GEO), where there may be continuous visibility. For this last case the operators may only need a small section of the visibility window. This section will be delimited by a *release time* and a *due time* (terms widely used in general machine scheduling referring to the availability and deadline times of the tasks), inside which the operators will request a *minimum duration* and a *maximum duration* of communication with the satellite.

We will consider problems that require scheduling over a finite time duration, termed the *scheduling horizon*, that is, finite time limits for enclosing the set of requests to be scheduled, e.g., seven days [2, 3]. We furthermore consider time to be discretized with a certain *discretization step*, e.g., 1 min [2, 3]. We show a set of visibility windows in Fig. 2.3.

The constraints on the requests are not only time-related. The operators may prefer to communicate with the satellite under lighting conditions, or more critically, the operators may need to communicate as soon as possible. Therefore these requests will have associated *priorities* to model these preferences. Reference [4] introduces the use of a suitability function for calculating these priorities.

Additionally, the operations plan may require one of the communications to be performed before another one. This constraint is known as *precedence* (imagine for example a relay network). Another constraint is that the communication may be required to be performed without interruptions, that is, with no *preemption*. And in some cases, two or more ground stations may communicate at the same time with the same satellite, which is known as *redundancy* or multiple (*m*-ary) capacity. Otherwise the problem has unitary capacity, and time-overlaying requests associated to either the same satellite or ground station are considered a *conflict*. For this case with unitary capacity a schedule with conflicts will not be *feasible*, and thus neither it will be valid as a possible solution to the problem. Generally no-preemption and no-redundancy are considered [2, 3], although some references do consider redundancy [5]. We will provide formal definitions for these terms in Chap. 3 (§3.1.2 and §3.1.3), but they are briefly presented now to illustrate the problem.

The Satellite Range Scheduling (SRS) problem requires us to find a feasible schedule that maximizes the sum of the priorities of the requests included in this schedule, given the requests and the associated constraints. We will refer to this sum of priorities as the performance or *metric* of the schedule. The problem will be even more complicated as we have several satellite missions to be served in a ground station network [6–8].

Additional constraints are introduced in the previous description which define different variants of this problem. In some systems, the mission operators may schedule in a *distributed* fashion rather than following a centralized schedule (consider for example different missions with different objectives computing their associated schedules without a central coordinating entity). In other cases, there could be *uncertainty* in the duration of the communication (e.g., considering uncertainty on the total amount of data to be sent). And other scenarios may be *dynamic*, requiring to be able to quickly react to changes in the model of the requests (consider the case where a satellite enters safe mode and requires immediate communication). These three variants are explained in more detail in Chaps. 5, 6 and 7, respectively.

We show in Fig. 2.4 a set of requests, indicating the start and end times of the visibility windows, the minimum and maximum duration requested by the operators along with their associated priorities, and the relations of no-preemption, no-redundancy, and precedence. In Fig. 2.5 we show a feasible schedule, complying with the constraints specified in Fig. 2.4.

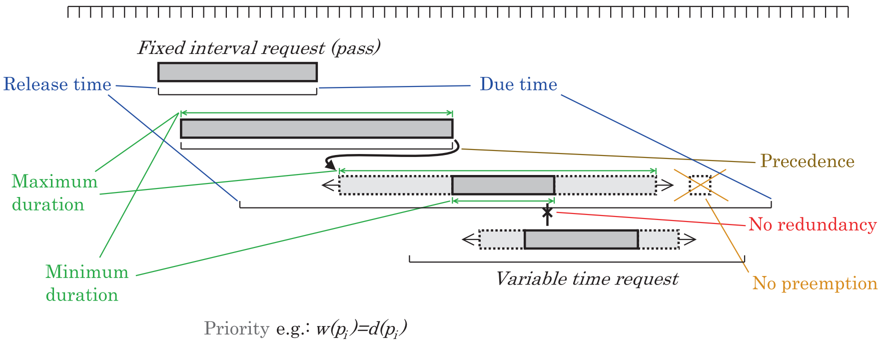


Fig. 2.4 Scheduling requests (based on visibility windows from Fig. 2.3)

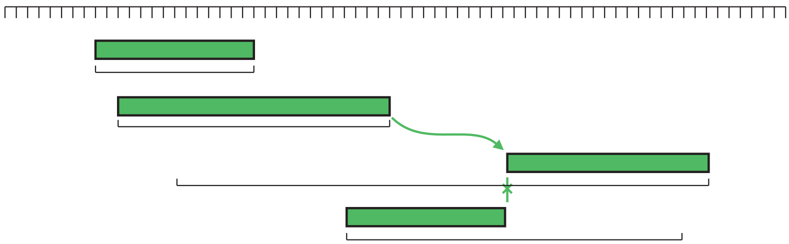


Fig. 2.5 Final schedule (based on requests from Fig. 2.4)

2.2 Scheduler Characteristics

We describe the different kinds of schedulers that can be used to solve the problem, and which depend on additional constraints on the missions. Our classification involves three parameters generally used in existing literature:

- *Topology*: the calculation of the schedule may be performed in a *centralized* fashion (that is, a single schedule is calculated based on the information from all the entities), or in a *distributed* fashion (different entities compute their associated schedules independently, without a coordinating entity).
- *Uncertainty*: we say that the problem is *deterministic* if the entities are 100% reliable, or that it is *stochastic* if the reliability is less than 100% (e.g., scheduled requests have a probability of not being executed).
- *Changes*: we say the requests are *static* if they do not change from the start of the scheduling horizon to its end, and that requests are *dynamic* if they may change before the completion of the scheduling horizon.

2.3 Satellite Range Scheduling Problems

According to the classification presented in §2.2, we present the problems that we will tackle in this book:

- *Satellite Range Scheduling*: classified as centralized, deterministic, and static, in this case the objective is to find the optimal schedule. We present the formulation of the problem in Chap. 3, and provide an algorithm for finding the optimal schedule in Chap. 4.
- *Noncooperative Satellite Range Scheduling*: classified as distributed, deterministic, and static, in this case the scheduling entities (either the ground stations or the satellites) aim at maximizing their own performance. We tackle this problem through a game-theoretic approach in Chap. 5.
- *Robust Satellite Range Scheduling*: classified as centralized, stochastic, and static, in this case the communication intervals are subject to fail, so that an approach to provide the maximum expected performance will be presented in Chap. 6.
- *Reactive Satellite Range Scheduling*: classified as centralized, deterministic, and dynamic, in this case requests change along the execution of the window, so the computation of the optimal schedule needs to be speed up. An approach based on the solution of the basic SRS problem is presented for this problem in Chap. 7.

A graphical representation for this classification is shown in Fig. 2.6.

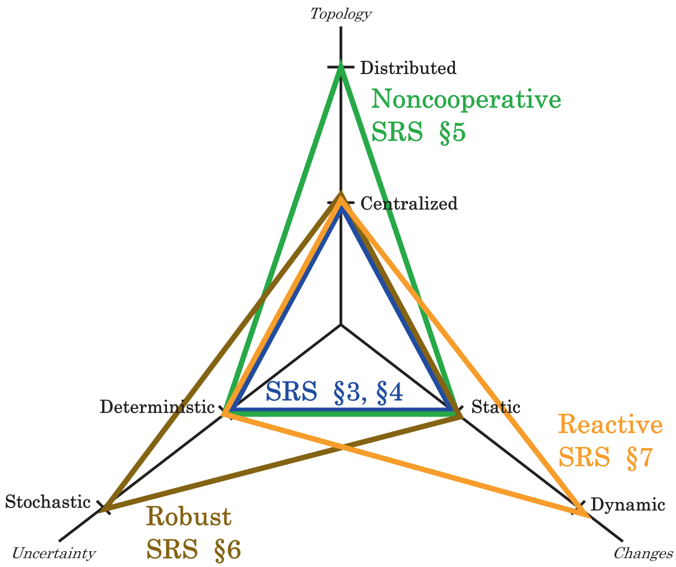


Fig. 2.6 Satellite Range Scheduling problems

2.4 Issues Beyond the Scope of this Text

As we have stated previously, this book is focused on reference (optimal) solutions. Therefore it is out of the scope of this work to reference lists of available *suboptimal solution algorithms* (see, for example, [2, 9, 10] for enumeration and performance comparison).

Neither we consider *satellite-specific models* including power, memory, and channel capacity (for suboptimal solutions on these problems see for example [10] for the deterministic problem with a single resource, [11] for the stochastic problem with failure probabilities, and [12] for the dynamic problem).

It is also out of the scope of this book to detail the *scheduling process in professional networks* (see [7, 13–15] for scheduling in NASA's Deep Space Network (DSN), [6, 16–18] for the ESA Tracking Station Network (ESTRACK), and [8, 19] for the Air Force Satellite Control Network (AFSCN)).

Acknowledgements This research was performed while the author held a National Research Council Research Associateship Award at the Air Force Research Laboratory (AFRL).

References

1. D.A. Vallado, *Fundamentals of Astrodynamics and Applications* (Space Technology Library, Microcosm Press, Portland, OR, 2001)
2. L. Barbulescu, J.P. Watson, L.D. Whitley, A.E. Howe, Scheduling space-ground communications for the air force satellite control network. J. Sched. **7**(1), 7–34 (2004)
3. F. Marinelli, F. Rossi, S. Nocella, S. Smriglio, A Lagrangian heuristic for satellite range scheduling with resource constraints. Comput. Oper. Res. **38**(11), 1572–1583 (2005)
4. W.J. Wolfe, S.E. Sorensen, Three scheduling algorithms applied to the Earth observing systems domain. Manag. Sci. **46**(1), 148–168 (2000)
5. M. Schmidt, Ground station networks for efficient operation of distributed small satellite systems. Ph.D. Thesis, University of Wurzburg, 2011
6. W. Heinen, M. Unal, Scheduling tool for ESTRACK ground station management, in *SpaceOps 2010* (American Institute of Aeronautics and Astronautics, Reston, VA, 2010)
7. M.D. Johnston, B.J. Clement, Automating Deep Space Network scheduling and conflict resolution. *Autonomous Agent and Multiagent Systems, AAMAS'06*, ACM (2006)
8. L. Barbulescu, A.E. Howe, L.D. Whitley, M. Roberts, Understanding algorithm performance on an oversubscribed scheduling application. J. Artif. Intell. Res. **27**, 577–615 (2006)
9. A. Globus, J. Crawford, J. Lohn, A. Pryor, A comparison of techniques for scheduling Earth observing satellites, in *Proceedings of the Sixteenth Innovative Applications of Artificial Intelligence Conference. IAAI* (2004)
10. S. Spangelo, Modeling and optimizing space networks for improved communication capacity. Ph.D. Thesis. University of Michigan, 2013
11. J. Wang, E. Demeulemeester, D. Qiu, A pure proactive scheduling algorithm for multiple Earth observation satellites under uncertainty of clouds. SSRN (2014). doi:10.2139/ssrn.2495339
12. J. Wang, X. Zhu, D. Qiu, L.T. Yang, Dynamic scheduling for emergency tasks on distributed imaging satellites with task merging. IEEE Trans. Parallel Distrib. Syst. **25**(9), 2275–2285 (2014)
13. B.J. Clement, M.D. Johnston, The Deep Space Network scheduling problem, in *IAAI'05 Proceedings of the 17th Conference in Innovative Applications of Artificial Intelligence*, **3** (AAAI, Palo Alto, CA, 2005), pp. 1514–1520
14. M.D. Johnston, D. Tran, B. Arroyo, S. Sorensen, P. Tay, B. Carruth, A. Coffman, M. Wallace, Automating mid- and long-range scheduling for NASA's Deep Space Network, in *SpaceOps 2012* (American Institute of Aeronautics and Astronautics, Reston, VA, 2012)
15. M.D. Johnston, D. Tran, Automated scheduling for NASA's Deep Space Network, in *6th International Workshop on Planning and Scheduling in Space* (Space Telescope Science Institute, Baltimore, MD, 2011)
16. S. Damiani, H. Dreihahn, J. Noll, M. Niezette, G.P. Calzolari, Automated allocation of ESA ground station network services, in *International Workshop on Planning and Scheduling for Space* (American Association for Artificial Intelligence, Palo Alto, CA, 2006)
17. S. Damiani, H. Dreihahn, J. Noll, M. Niezette, G.P. Calzolari, A planning and scheduling system to allocate ESA ground station network services, in *The International Conference on Automated Planning and Scheduling, ICAPS 2007* (AAAI, Palo Alto, CA, 2007)
18. H. Dreihahn, M. Niezette, M. Gotzelmann, Centralized schedule and SLE service configuration file generation with the ESTRACK scheduling system. *European Ground System Architecture Workshop* (ESA, Paris, 2007)
19. B.R. Hays, A.M. Carlile, T.S. Mitchel, Visualizing and integrating AFSCN utilization into a common operational picture. *Advanced Maui Optical and Space Surveillance Technologies Conference, AMOS 2006* (Maui Economic Development Board, Kihei, HI, 2006)

http://www.springer.com/978-3-319-25407-4

An Introduction to Optimal Satellite Range Scheduling
Vázquez Álvarez, A.J.; Erwin, R.S.
2015, XXVII, 162 p. 69 illus., 20 illus. in color., Hardcover
ISBN: 978-3-319-25407-4